MQ-200

LCD/CRT 2D Graphics Subsystem Data Book

Version 0.03

May 7, 1999

Preliminary Product Information

Copyright Notice

Restricted Rights Legend

Trademark Acknowledgment

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

Table 0-1: Revision History

Revision	Date	By	Comment
0.01	11/23/98	SRV	Preliminary
0.02	12/9/98	SV,MB,BD	Electricals, AC Timings
0.03	5/7/99	SV,MB	Corrected LCD Interface Specification and added Interrupt Controller Registers.

Contents

Chapter 1 MQ-200 LCD/CRT 2D Graphics Subsystem 1-1

Preliminary Product Information 1-1
MQ-200 Key Features 1-1
MQ-200 Overview 1-2
MQ-200 Features 1-2
Superior Graphics Performance 1-2
Two Independent Graphics Controllers 1-3
Dual-Display System Implementation with QView™ Technology 1-3
Direct Interface to a Broad Range of Liquid Crystal Displays 1-3
Programmable Pulse Width Modulated (PWM) Outputs
Control the Brightness and Contrast of the Liquid Crystal Display 1-3
Direct Interface to CRT Monitor Displays 1-4
Integrated Programmable Frequency Synthesizers Provide
Independent Frequencies for Display and Memory Operation 1-4
High Performance System Solution 1-4
Embedded 2-MByte SDRAM for Display Memory 1-4
Advanced Power Management Features 1-4
Fully Static Design fabricated in CMOS Process 1-4
Packaged in 180 pin Flex BGA package (12x12 mm)
with 0.8 mm ball pitch and 0.5 mm ball size 1-4

Chapter 2 Pin Description 2-1

MQ-200 LCD/CRT 2D Graphics Subsystem 2-1
LCD Interface 2-2
CRT Interface 2-4
CPU Interface 2-4
Power Management and Miscellaneous Pins 2-10
Clocks 2-11

Chapter 3 Functional Description 3-1

CPU Interface 3-3
Overview 3-4
SH-7750 Interface 3-4
SH-7709 Interface 3-5
Intel SA-1110 Interface 3-7
NEC VR-4121 Interface 3-8
Toshiba Tx-3922 Interface 3-9
Interrupt Controller 3-9
Clocks and Oscillator 3-10

Graphics Engine 3-11
Theory of Operation 3-11
Raster-Ops 3-12
Clipping and Transparency 3-12
Bresenham Line Draw 3-12
Monochrome to Color Expansion 3-13
Graphics Engine Register Set 3-13
Memory Subsystem 3-13
Display Section 3-13
Theory of Operation 3-16
Display Clocks 3-17
Pixel Generation 3-18
Look-Up Table Expansion 3-18
Hardware Cursor 3-18
Hardware Cursor Image Format 3-19
Flat Panel Interface 3-19
Flat Panel Interface External Signals 3-20
Signal-To-Pin Mapping for Different Display Types 3-21
Interface Signal Mapping for Color Panels 3-21
Interface Signal Mapping for Mono Panels 3-23
Timing Diagrams for S-STN Panels 3-24
4-Bit Color S-STN 3-24
8-Bit Color S-STN 3-24
12-Bit Color S-STN 3-25
16-Bit Color S-STN 3-25
24-Bit Color S-STN 3-26
Timing Diagrams for Color D-STN panels 3-27
8-Bit Color D-STN 3-27
16-Bit Color D-STN 3-27
24-Bit Color D-STN 3-28
Timing Diagrams for Mono S-STN Panels 3-29
4-Bit Mono S-STN 3-29
8-Bit Mono S-STN 3-29
16-Bit Mono S-STN 3-30
Timing Diagrams for Mono D-STN Panels 3-30
8-Bit Mono D-STN 3-30
16-Bit Mono D-STN 3-31
Pulse-Width Modulator 3-31
Controlling Backlight and Contrast 3-32
General Purpose I/O Port 3-32
General Purpose Output Port 3-32
CRT Interface 3-32
CRT Control Signals 3-32
Digital-to-Analog Conversion 3-33
Blanking and SYNC Pedastal 3-33

Cyclic Redundancy Check (CRC) 3-33
Graphics Controller Register Set 3-33

Chapter 4 Power Management 4-1

Theory of Operation 4-1
D0 State 4-1
D1 State 4-1
D2 State 4-1
D3 State 4-1
D4 State 4-2
Software-Initiated Mode Transitions 4-2
Hardware Mode Transitions 4-2
Role of the Operating System in Managing Power 4-2
Role of the Device Driver in Managing Power 4-2
Power Management Unit 4-2
Input Clocks 4-3
Power Sequencing 4-3
Power Management Register Set 4-4

Chapter 5 Programming Information 5-1

MQ-200 Address Map 5-1
Register Locator Table 5-4
Device Initialization 5-7
Recommended Initialization Procedure 5-7
Memory Configuration 5-8
Graphics Controller Programming Information 5-9
CRT Interface 5-12
Flat Panel Interface 5-12
Single-Display Configuration 5-13
Simultaneous Display Configuration 5-13
QView™ Single Image 5-13
QView™ Dual Image 5-13
C100R Through C1FFR: Color Palette 5-43
GC20R: Graphics Controller 2 Control Register 5-44
GC21R: Graphics Controller CRC Result Register 5-47
GC22R: Horizontal Display Timing Control Register 5-48
GC23R: Vertical Display Timing Control Register 5-49
GC24R: Horizontal SYNC Control Register 5-50
GC25R: Vertical SYNC Control Register 5-51
GC26R and GC27R: Reserved 5-51
GC28R: Graphics Controller 2 Main Window Horizontal Control Register 5-52
GC29R: Graphics Controller 2 Main Window Vertical Control Register 5-53
GC2AR: Graphics Controller 2 Alternate Window Horizontal Control Register . 5-54

GC2BR: Graphics Controller 2 Alternate Window Vertical Control Register 5-55
GC2CR: Main Window 2 Start Address 5-56
GC2DR: Alternate Window 2 Start Address 5-57
GC2ER: Window 2 Stride Register 5-58
GC2FR: Window 2 Line Size Register 5-59
GC30R: Hardware Cursor 2 Position 5-60
GC31R: Hardware Cursor 2 Start Address and Offset Register 5-61
GC32R: Hardware Cursor 2 Foreground Color Register 5-62
GC33R: Hardware Cursor 2 Background Color Register 5-63
GC34R Through GC3FR: Reserved 5-63
Flat Panel Interface Programming Information 5-64
FP00R: Flat Panel Control Register 5-65
FP01R: Flat Panel Output Pin Control Register 5-68
FP02R: Flat Panel General Purpose Output Control Register 5-72
FP03R: General Purpose I/O Port Control Register 5-74
FP04R: STN Panel Control Register 5-76
FP05R: D-STN Half-Frame Buffer Control Register 5-77
FP0FR: Pulse Width Modulation Control Register 5-78
FP10R to FP2FR: Frame-Rate Control Pattern Registers 5-80
FP30R to FP37R: Frame-Rate Control Weight Registers 5-81
Graphics Engine Programming Information 5-82
Rectangle Fill (Solid Fill) 5-83
Regular BitBLT – Screen to Screen 5-84
Overlapping Rectangles 5-84
Example 5-84
Regular BitBLT – Memory to Screen 5-85
Monochrome-to-Color Expansion BitBLT 5-87
Transparent BLT 5-88
Pattern BitBLT 5-90
Regular BitBLT - Screen to Memory 5-91
Line Drawing 5-91
Graphics Engine Register Definition 5-92
GE00R: Primary Drawing Command Register 5-93
GE01R: Primary Width and Height Register 5-96
GE02R: Primary Destination Address Register 5-97
GE03R: Primary Source XY Register 5-98
GE04R: Primary Color Compare Register 5-100
GE05R: Primary Clip Left/Top Register 5-101
GE06R: Primary Clip Right/Bottom Register 5-102
GE07R: Primary Source and Pattern Offset Register 5-103
GE08: Primary Foreground Color Register / Rectangle Fill Color 5-104
CPU Interface Programming Information 5-113
CPU Register Definition 5-113
Memory Interface Unit Programming Information 5-116

Memory Interface Unit Register 5-116
Power Management Unit Programming Information 5-122
Power Management Unit Register Definition 5-122
PMR: Power Management Unit Configuration Register 5-125

Chapter 6 Electrical Specifications 6-1

Chapter 1

MQ-200 LCD/CRT 2D Graphics Subsystem

Preliminary Product Information

The MQ-200 is an integrated 2D-accelerated graphics and display sub-system on a single chip designed for high performance mobile, connected platforms such as Smart Appliances and PC Companion devices. The MQ-200 is the first device of its kind to provide high-performance interface to multiple CPUs including the Hitachi SH-7750, the Intel StrongARM SA-1110, the NEC VR4121 and Toshiba Tx-3922 processors. The seamless interface of the MQ-200 to a rich mix of RISC CPUs is optimized for performance with each CPU architecture minimizing cycle counts and maximizing system throughput. In addition, a PCI bus interface provides a standardized, high performance interface to the MQ-200 for any system implementation.

MQ-200 Key Features

- 128-Bit 2D graphics acceleration engine
- Embedded 2-MByte SDRAM for display memory
- High Performance bus supports multiple CPUs: Hitachi SH-7750, Intel SA-1110, NEC VR-4121, Toshiba Tx-3922
- 33 MHz PCI V 2.1 standardized bus interface
- Direct interface to TFT, D-STN and S-STN LCDs
- Display resolutions up to 1280x1024; pixel depths from 1 to 32 bits per pixel
- Direct interface to CRT monitors with on-chip true color DAC
- QView^{TM} 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
- DynamiQ^{TM} advanced power management and Microsoft Windows CE power management support
- Fully static design fabricated in CMOS Process

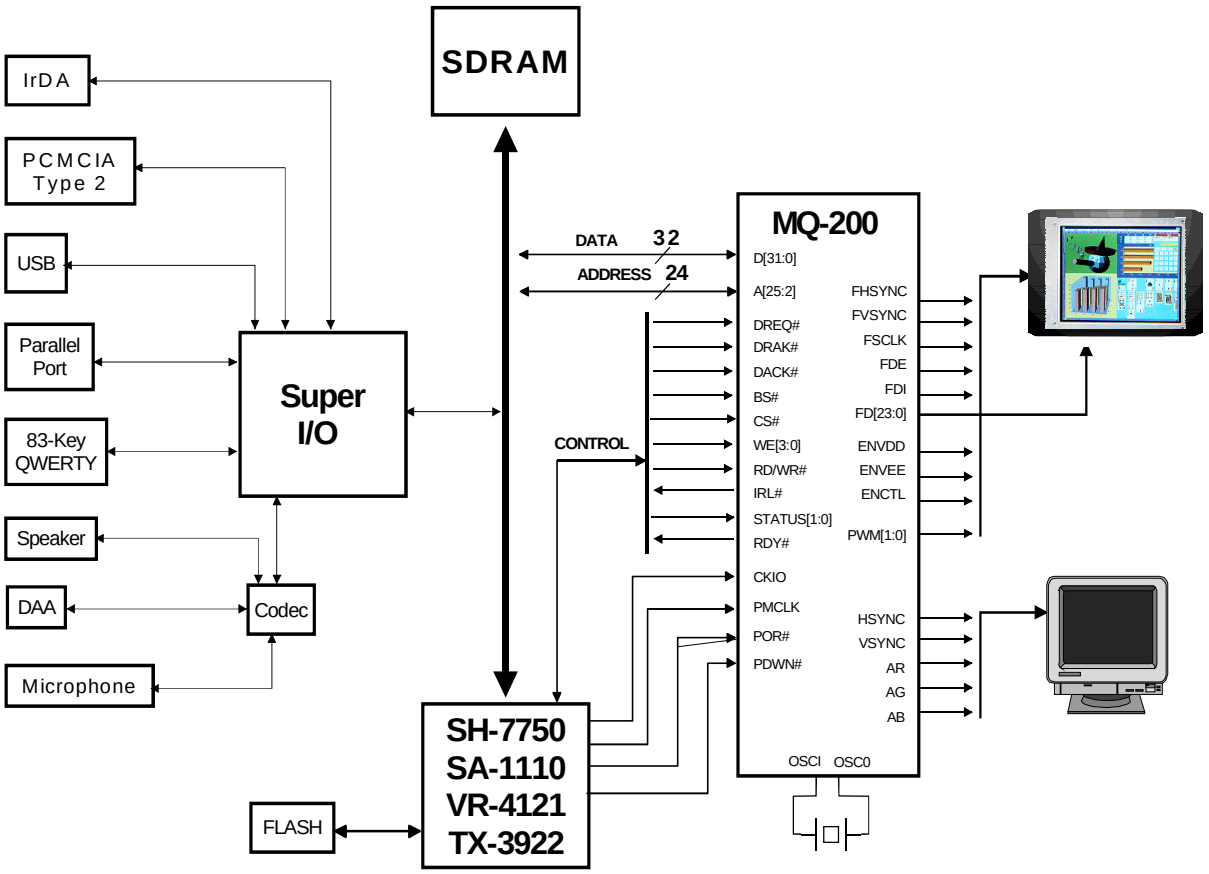


Figure 1-1: MQ-200 based system implementation diagram

MQ-200 Overview

The MQ-200 is an integrated 2D-accelerated graphics and display sub-system with 2 MBytes of Embedded DRAM on a single chip to provide high performance mobile solutions. The MQ-200 graphics engine and two on-board graphics controllers share the unique high-bandwidth on-chip memory bus ensuring that graphics operations are not adversely affected by display refresh requirements. The MQ-200 offers unique simultaneous display capability called *QView™* which lets the user configure the primary LCD display and the optional CRT display at different resolutions, refresh rates and color depths. The MQ-200 employs advanced low-power design techniques including *DynamiQ™* to implement dynamic power management, making the MQ-200 the ideal choice for mobile applications with demanding display requirements.

MQ-200 Features

Superior Graphics Performance

- Integrated high-performance 128-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, font caching and image caching
- High-speed interface to the internal frame buffer memory

- Graphics acceleration support for 8, 16 and 32 bits per pixel modes
- 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)
  - Bresenham Line Draw Acceleration

Two Independent Graphics Controllers

- Full-featured CRT and LCD graphics controllers
- Support for 1, 2, 4, 8, 16, 24 and 32-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 16, 24 and 32-bpp modes
- Each graphics controller contains:
  - FIFO for data from display memory
  - Completely programmable display timing generator
  - 64x64 hardware cursor

Dual-Display System Implementation with QView™ Technology

QView™ technology enhances the quality of the images in dual-display applications by providing separate refresh rates, resolutions and color depths for each display.

- The primary and secondary display devices can show the same image or independent images with programmable resolutions, color depths and refresh rates
- Allows “panning” of an image on LCD and CRT display windows when the same image is displayed; when different images are 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 supports monochrome TFT, S-STN, and D-STN panels
- Advanced dithering and Frame rate control (FRC) algorithms 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 1280x1024 (SXGA) resolutions on CRT monitors with multiple refresh rates
- Built-in High Performance triple TrueColor DAC with monitor sense circuitry, blank pedestal and sync pedestal
- Cyclic Redundancy Check logic to ensure 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 to provide three internal clocks.

High Performance System Solution

- Direct, “glueless” 32-bit interface to Hitachi SH-7750, Intel SA-1110, NEC VR-4121 and Toshiba Tx-3922
- 128-bit Graphics Engine operates independent of the CPU, performing BitBLTs, 3 operand rasterops (ROPs) and Bresenham Line Draw Acceleration
- 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 2-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 form factor by eliminating external display memory

Advanced Power Management Features

- Compliant with the Microsoft OnNow Architecture Initiative: D0 (normal), D1, D2, D3 and D4 (Suspend) power states are provided; transitions between states is accomplished via software
- D1, D2 and D3 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-200 are automatically shut down without any software intervention, optimizing power consumption in all operating modes

Fully Static Design fabricated in CMOS Process

Packaged in 180 pin Flex BGA package (12x12 mm) with 0.8 mm ball pitch and 0.5 mm ball size

Chapter 2

Pin Description

MQ-200 LCD/CRT 2D Graphics Subsystem

The following are valid pin types:

Table 2-1: Valid Pin Types

Pin Type	Pin Type Description
I	Input pin.
O	Output pin.
I/O	Bidirectional input/output pin.
P	Power (VDD) pin.
S I/O	Sustained I/O 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: '#' indicates active low signal

LCD Interface

Table 2-2: LCD Interface

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
ENVDD	O	B9	8 / 16	Enable VDD In panel power-up sequence, this signal is asserted first; while in panel power-down sequence, this signal is driven last (minimum 32 ms after ENCTL). This signal is driven low during reset.
ENCTL (GPO0)	O	A9	8 / 16	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	C9	8 / 16	Enable VEE In panel power-up sequence, this signal is asserted last (minimum 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	B11, E10, D11, B12, C12, A13, A14, B13, B14, D12, C13, E11, D13, D14, E12, F11, E13, E14, F12, F13, G11, G12, G14, G13	8 / 16	Flat Panel Data (programmable polarity) 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, FD2 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 also when the flat panel is disabled regardless of set polarity.
FDE FMOD	O O	A11	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	D10	8 / 16	Flat Panel Data Inversion If flat panel data inversion is enabled, this signal indicates if 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	C11	8 8	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.

Table 2-2: LCD Interface

FHSYNC LP	O O	B10	8 8	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	C10	8 / 16	Flat Panel Shift Clock Rising edge of this clock is used to output flat panel data and control signals. Falling edge of this clock is normally used to externally latch flat panel data and control signals.
PWM0 (GPO2)	O	C1	8/16	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 backlog. If not used as PWM0, this signal can be used as General Purpose Output Port Bit 2.
PWM1 (GPO3)	O	D4	8/16	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	C2, C3, G5	8/16	General Purpose I/Os. These pins can be programmed to be either inputs or outputs. These pins are programmed as inputs during reset.
FVDD [2:0]	P	G10, F10, E8	-	Flat Panel Interface (3.3V) Power Supply pins
FGND [2:0]	G	F14, C14, A12	-	Flat Panel Interface Ground Pins

CRT Interface

Table 2-3: CRT Interface

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
AR	A	C6	16	Analog Red Output to the CRT monitor
AG	A	B6	16	Analog Green Output to the CRT monitor
AB	A	A7	16	Analog Blue Output to the CRT monitor
VSYNC	O	A1	8	Vertical Sync Output to the CRT monitor. In NAND Chain test mode, this pin is the output of the NAND chain.
HSYNC	O	B1	8	Horizontal Sync Output to the CRT monitor
VREFIO	A	B8	-	Voltage Reference Internal voltage reference output, connect this pin to analog ground pin through a 0.1 uF capacitor.
FSADJ	A	D8	-	Full-Scale Adjust This pin must be tied GND through a 47 ohm external resistor to set the full scale current value for the DAC outputs.
COMPC	A	C8	-	Compensation This pin is tied to one of the analog VDD pins through a 0.1 uF capacitor, such as AVDD1
AVDDC [3:1]	AP	C7, D7, E7		Analog Power for CRT DAC (2.5V)
AGNDC [3:1]	AG	A6, B7, A5		Analog GND for CRT DAC
AVDDD	P	D9	-	Digital power pins for CRT DAC REFERENCE (2.5 V)
AGNDD	G	A8	-	Digital ground pins for CRT DAC REFERENCE

CPU Interface

Table 2-4: CPU Interface

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
Hitachi SH-7750				
CKIO	I	M8	-	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.

Table 2-4: CPU Interface

BS#	I	L8	-	Bus Cycle Start CPU asserts this signal to request a bus cycle.
A[25:24]	I	H11, H12	-	CPU Address Bus bits [25:24] This address bus is driven by the CPU for all read/write accesses. During test mode operation, these bits determine the processor interface being supported.
A[23:2]	I	H14, H13, J11, J12, J13, K12, K14, K11, H3, H4, J2, J1, J3, K1, K2, J4, K3, L1, L2, J5, K4, M2	-	CPU Address Bus This address bus is driven by the CPU for all read/write accesses.
D[31:0]	I/O	L3, N1, N2, P2, M3, N3, P3, L4, L5, N5, P5, M5, N6, M6, L6, N7, N9, P9, M9, N10, L9, M10, P11, N11, P12, N12, M11, P13, N13, N14, M12, M13	8/16	CPU Data Bus This data bus is driven by the CPU during write accesses and driven by MQ-200 during read accesses. These pins are tri-stated during reset and when the CPU is disabled (powered down).
CS#	I	M7	-	Chip Select Chip Select signal from the CPU to the MQ-200. MQ-200 decodes the address presented on the CPU Address bus only when this signal is asserted.
WE[3:0]#	I	N4, P7, K9, M14	-	Write Enables[3:0] These are the byte enables [3:0] used for writes from the CPU to the MQ-200. Read operations are always 32-bit wide reads.
RD/WR#	I	L7	-	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.
IREQ#	O	L13	24	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 the MQ-200. Active Low. Connected to one of the IRL lines on the CPU.
RDY#	O	N8	24	Ready (active low) This signal is deasserted by the MQ-200 to request additional hardware wait states when it is not ready to complete the requested bus cycle
DREQ#	O	L10	24	Data Transfer Request Data Transfer Request Input from the MQ-200 to Channel 0 or 1 of the DMA Controller in the CPU.
DRAK	I	P8	-	DMA Transfer Request Acknowledge DMA Transfer Request Acknowledge from the Channel 0 or 1 of the DMA Controller in the CPU to the MQ-200. During normal/test modes, this pin along with the DACK pin is used to select the processor interface.

Table 2-4: CPU Interface

DACK	I	M4		Strobe from Channel 0 Strobe output to MQ-200 from Channel 0 or 1 of the DMA Controller in the CPU. During normal/test modes, this pin along with DRAK pin is used to select the processor interface.
NEC VR-4121/4111				
A[25]	I	H11		No Connect. This input pin should be tied to GND to save current drain.
A[24]	I	H12	-	CPU Address Bus bit 24 During normal mode of operation, this pin provides the strapping option to decide the 8M of the 16M bank that MQ-200 resides in. If A[24] = 0, the MQ-200 occupies the lower 8M of the 16M address space. If A[24]=1, the MQ-200 occupies the upper 8M of the 16M address space. For test mode operations, these bits decide the processor interface being supported.
A[23:2]	I	H14, H13, J11, J12, J13, K12, K14, K11, H3, H4, J2, J1, J3, K1, K2, J4, K3, L1, L2, J5, K4, M2		CPU Address Bus This address bus driven by CPU for all read/write accesses.
D[31:0]	I/O	L3, N1, N2, P2, M3, N3, P3, L4, L5, N5, P5, M5, N6, M6, L6, N7, N9, P9, M9, N10, L9, M10, P11, N11, P12, N12, M11, P13, N13, N14, M12, M13	8/16	CPU Data Bus This data bus is driven by CPU during write accesses and driven by MQ-200 during read accesses. These pins are tri-stated during reset and when CPU is disabled (powered down).
BE[3:0]#	I	N4, P7, K9, M14		Byte Enables [3:0] (Active Low) These are the byte enables [3:0] used for writes from CPU to the MQ-200. Reads are always 32-bit reads. These signals are connected to the processor signals UUCAS#, ULCAS# and LCAS# respectively.
CS#	I	M7		Chip Select (Active Low) Chip select signal from the CPU to the MQ-200. MQ-200 decodes the address presented on the CPU address bus only when this signal is asserted. This signal is connected to LCDCS# of the processor.
RD#	I	L7		Read Signal (Active Low) This signal becomes active when data is read from the MQ-200. This signal is connected to the RD# pin of the processor.
WR#	I	L10		Write Signal (Active Low) This signal becomes active when data is written to the MQ-200. This signal is connected to the WR# pin of the processor.
RDY	O	N8	24	Ready (Active High) This signal is generated by the MQ-200. It is de-asserted by the MQ-200 at the appropriate time to request additional Hardware Wait States. This signal is connected to the LCDRDY signal of the processor.

Table 2-4: CPU Interface

IREQ#	O	L13	24	Interrupt Request Line This is an interrupt request output. This is connected to the appropriate CPU interrupt input pin.
MD[4:3]	I	P8, M4		Mode Select. These pins should be tied to GND at Power-On-Reset, to choose the VR4121 mode of operation.
CKIO	I	M8		No Connect. This input pin should be tied to GND to save current drain.
BS#	I	L8		No Connect. This input pin should be tied to GND to save current drain.
Toshiba Tx-3922				
CKIO	I	M8		Bus Interface Clock This is the clock used by the CPU for generating the bus interface signals. The maximum frequency of this clock is 83 MHz. This input to the MQ-200 is connected to the DCLKOUT pin of the Tx-3922.
ALE	I	L8		Address Latch Enable This pin is used as the address latch enable to latch address bits A[12:0] to generate address bits A[25:13].
A[25:13]	I	H11,H12,H14,H13, J11,J12,K12,K14, K11,H3		No Connect. These input pins should be tied to GND to save current drain.
A[12:0]	I	H4, J2, J1, J3. K1, K2, J4, K3, L1, L2, J5, K4, M2		CPU Address Bus This address is driven by the CPU and for all read/write accesses. The address lines are multiplexed. While ALE is active, these lines represent address bits A[25:13].
D[31:0]	I/O	L3, N1, N2, P2, M3, N3, P3, L4, L5, N5, P5, M5, N6, M6, L6, N7, N9, P9, M9, N10, L9, M10, P11, N11, P12, N12, M11, P13, N13, N14, M12, M13	8/16	CPU Data Bus This data bus is driven by the Tx-3922 during write accesses and driven by the MQ-200 during read accesses. These pins are tri-stated during reset and when the CPU is disabled (powered down).
CS#	I	M7		Chip Select (Active Low) Chip Select signal from the CPU to the MQ-200. MQ-200 decodes the address presented on the CPU address bus only when this signal is asserted. This signal connects to either MCS0 or MCS1 pins of the Tx-3922. The MQ-200 occupies the upper 32M address space realized by each chip select.
WE[3:0]#	I	N4, P7, K9, M14		Write Enables [3:0] (Active Low) These are the byte enables [3:0] used for writes from the CPU to the MQ-200. Reads are always 32-bit wide reads. These signals are connected to WE[3:0]# signals of the Tx-3922.
RD#	I	L7		Read Signal (Active Low) This is the read signal from the CPU.

Table 2-4: CPU Interface

WR#	I	L10		Write Signal (Active Low) This is the write signal form the CPU.
WAIT#	O	N8	24	Wait (Active Low) This is the signal generated by MQ-200. It is de-asserted by the MQ-200 at the appropriate time to request additional Hardware Wait States. This signal is connected to MCS0WAIT or MCS1WAIT signal of the Tx-3922 depending on which MCS pin the CS# is connected to on the CPU.
IREQ#	O	L13	24	Interrupt Request Line (Active Low) This is an interrupt request output. This is connected to the appropriate interrupt input pin of the Tx-3922.
MD[4:3]	I	P8, M4		Mode Select. These pins should be tied to MD4=GND and MD3=VDD at Power-On-Reset, to choose the Tx-3922 mode of operation
PCI Bus				
CLK	I	M8		Bus Interface Clock This is the clock by the PCI controller for generating the bus interface signals. The maximum frequency of the clock is 33 MHz.
RST#	I	M1		Reset Signal (Active Low) This reset signal is used to bring PCI specific registers, sequencers and signals to a consistent state.
A[25:2}	I	H11,H12,H14, H13, J11, J12, J13, K12, K14, K11, H3, H4, J2, J1, J3, K1, K2, J4, K3, L1, L2, J5, K4, M2		No connect. These input pins should be tied to GND to save current drain.
AD[31:0]	I/O	L3, N1, N2, P2, M3, N3, P3, L4, L5, N5, P5, M5, N6, M6, L6, N7, N9, P9, M9, N10, L9, M10, P11, N11, P12, N12, M11, P13, N13, N14, M12, M13	8/16	Address/Data Bus Address and Data are multiplexed on the same PCI pins.
C/BE[3:0]#	I	N4, P7, K9, M14		Command/Byte Enable (Active Low) Signals These signals provide command during Address phase and Byte Enable information during the data phase of the bus cycle.
FRAME#	I	M7		FRAME Signal (Active Low) This signal is driven by the PCI host to indicate the beginning and duration of an access
IRDY#	I	L7		Initiator Ready (Active Low) Initiator Ready indicates the PCI Host's ability to complete the current data phase.
TRDY#	O	N8	24	Target Ready (Active Low) Target Ready indicates MQ-200's ability to complete the current data phase.

Table 2-4: CPU Interface

STOP#	O	L10	24	STOP Signal (Active Low) STOP indicates that MQ-200 is requesting the PCI Host to stop the current transaction.
DEVSEL#	O	L8	24	Device Select (Active Low) This signal when actively driven, indicates that the MQ-200 has actively decoded its address as the target of the current access.
IDSEL	I	L2		Initialization Device Select (Active High) This signal is used as a chip select during configuration read and write transactions
IREQ#	O	L13	24	Interrupt Request Line (Active Low) This is an interrupt request output. This is connected to the PCI Host signal INTA pin.
PAR	I/O	L1		Parity Signal This is even parity across AD[31:0] and C/BE[3:0]. During write transactions, PCI host drives this signal. During read transactions, MQ-200 drives this signal.
PERR#	O	M2	24	Parity Error-Data Cycle This signal, when active, indicates that parity error occurred during the data transfer cycle.
CLKRUN	I/O	J5	24	Clock Run This signal is from the PCI bus controller indicating that the bus clock will be shut down.
SERR#	O	K4	24	Parity Error-Data/Command Cycle This signal indicates parity error during address/command transfer cycles.
MD[4:3]	I	P8, M4		Mode Select. These pins should be tied as MD4= VDD and MD3=GND at Power-On-Reset, to choose the PCI mode of operation
Intel SA-1110				
nCS	I	M7		Chip Select Active low chip select output from SA-1110
ADDR [25:2]	I	H14, H13, J11, J12, J13, K12, K14, K11, H3, H4, J2, J1, J3, K1, K2, J4, K3, L1, L2, J5, K4, M2		CPU Address Bus This address bus is driven by the CPU during all read/write accesses.
DATA [31:0]	I/O	L3, N1, N2, P2, M3, N3, P3, L4, L5, N5, P5, M5, N6, M6, L6, N7, N9, P9, M9, N10, L9, M10, P11, N11, P12, N12, M11, P13, N13, N14, M12, M13		CPU Data Bus This data bus is driven by the SA-1110 during write accesses and driven by the MQ-200 during read accesses. These pins are tri-stated during reset and when CPU is disabled (powered down).
nOE	I	L7		Output Enable Active Low Output Enable

Table 2-4: CPU Interface

nWR	I	L10		Write Enable Active low Write Enable
RDY	O	N8		Data Ready Active high data ready signal
MD[4:3]	I	P8, M4		Mode Select. These pins should be tied to VDD at Power-On-Reset, to choose the SA-1110 mode of operation.
WE[3:0]	I	N4,P7,K9,M14		No connect. These input pins should be tied to GND to save current drain.
CKIO	I	M8		No Connect. This input pin should be tied to GND to save current drain.
BS#	I	L8		No Connect. This input pin should be tied to GND to save current drain.
CPU Interface Power and Ground Pins				
BVDD [2:0]	P	H10, K5, K8,		VDD for IO pins for Bus Interface (3.3V, 3 pairs) These pins supply power for bus interface signals.
BGND [2:0]	G	P14, P10, P4		Ground for IO pins for Bus Interface These pins supply ground for bus interface signals
CORE Power and Ground Pins				
CVDD [3:0]	P	K10, E5, J10, E9		VDD for Core (2.5 V) These pins supply power for the MQ-200 Internal Core
CGND [3:0]	G	L14, D10, J14, A10		Ground for Core These pins supply ground for the MQ-200 Internal Core

Power Management and Miscellaneous Pins

Table 2-5: Miscellaneous Pins

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
PDWN#	I	K13		Power Down This is an active low input to the MQ-200. This places the MQ-200 in the lowest power state.
PMCLK	I	L11		Power Mode Clock This is the 16.384KHz clock from the CPU, used to control the power management state machine in the MQ-200.
EVDD[1:0]	P	H5, K6		Power Supply for EDRAM (2.5V)
EGND[1:0]	G	G1,P6		Ground for EDRAM supply

Table 2-5: Miscellaneous Pins

EVDD3.3	P	K7		Power supply for EDRAM (3.3V)
EGND3.3	G	P1		Ground for EDRAM supply
VDDOSC 3.3	P	A3		Power supply for on-board Oscillator (3.3 V)
VGNDOSC 3.3	G	A2		Ground for on-board Oscillator supply
BATT-FAULT#	I	G3		Battery Fault condition input. Active low input.
NC[12:1]		H1,H2,G4,G2,F4,F3, F1,,F2,F5,E3,D3,D2		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.

Clocks

Table 2-6: Clocks

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
OSCO	A	C4		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 12.288MHz.
OSCI L1CLK	A	B3		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 the MQ-200 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	D5		PLL 2 Bypass Clock When the MQ-200 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	B2		PLL 3 Bypass Clock When the MQ-200 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.
AVDD1	AP	D6		Analog VDD for the System Clock PLL (2.5 V). This pin provides power to the memory clock (MCLK) PLL.
AGND1	AG	B5		Analog GND for the System Clock PLL. This pin provides ground to the memory Clock (MCLK) PLL.
AVDD2	AP	C5		Analog VDD for the Dot Clock PLL (2.5 V). This pin provides power to the dot clock (DCLK) PLL.
AGND2	AG	A4		Analog GND for the Dot Clock PLL. This pin provides ground to the dot clock (DCLK) PLL.
AVDD3	AP	E6		Analog VDD for the Dot Clock PLL (for CRT) (2.5 V). This pin provides power to internal dot clock (DCLK) PLL.
AGND3	AG	B4		Analog GND for the Dot Clock PLL (for CRT). This pin provides ground to the dot clock (DCLK) PLL.

Table 2-6: Clocks

POR#	I	M1		Power-On Reset (active low) This signal resets the whole system. It clears all the registers including RTC and its registers. It also puts the system in STANDBY mode.
SHIF	I	E4		SH Interface Pin If MD[1:0] pins are set to 00, this pin selects between the SH processor interfaces. A ’0’ on this pin select the SH-7750 and a ’1’ selects the SH-7709 processor interface.
MD[2:0]	I	L12, E2, E1		Mode Select These pins should be tied to VDD or GND, at Power-On-Reset, to choose the mode of operation. These pins together with the strapping options on L2CLK, DRAK(MD4), DACK(MD3) and A[25:24] decide the mode of operation as shown in the table below:

MD[2:0]	SHIF	A[25:24]	DRAK (MD4)	DACK (MD3)	Operating Mode
000	0	XX	X	X	Hitachi SH-7750
000	1	XX	X	X	Hitachi SH-7709
001	X	XX	0	0	NEC VR-4121
001	X	XX	0	1	Toshiba Tx-3922
001	X	XX	1	0	PCI Bus
001	X	XX	1	1	Intel SA-1110
010	0	XX	0	0	Reserved: Test mode
010	1	XX	0	0	Reserved: Test mode
010	X	XX	0	1	Reserved: Test mode
010	X	00	1	0	Reserved: Test mode
010	X	01	1	0	Reserved: Test mode
010	X	10	1	0	Reserved: Test mode
011	0	11	X	X	Reserved: Test mode
011	1	11	X	X	Reserved: Test mode
011	X	10	0	0	Reserved: Test mode
011	X	10	0	1	Reserved: Test mode
011	X	10	1	0	Reserved: Test mode
011	X	10	1	1	Reserved: Test mode
011	X	01	X	X	Reserved: Test mode
011	X	00	0	X	Nand chain test mode
011	X	00	1	X	Reserved: Test mode
100	X	XX	X	X	Reserved: Test mode

Table 2-7: Pins Sorted by Ball Number

Pin	BGA Ball	SH-4 IF	NEC-4121 IF	Toshiba-3922 IF	PCI IF	SA-1110 IF
1	A1	VSYNC (O)	VSYNC (O)	VSYNC (O)	VSYNC (O)	VSYNC (O)
2	B1	HSYNC (O)	HSYNC (O)	HSYNC (O)	HSYNC (O)	HSYNC (O)
3	G5	GPIO0 (IO)	GPIO0 (IO)	GPIO0 (IO)	GPIO0 (IO)	GPIO0 (IO)
4	C3	GPIO1 (IO)	GPIO1 (IO)	GPIO1 (IO)	GPIO1 (IO)	GPIO1 (IO)
5	C2	GPIO2 (IO)	GPIO2 (IO)	GPIO2 (IO)	GPIO2 (IO)	GPIO2 (IO)
6	C1	PWM0 (O)	PWM0 (O)	PWM0 (O)	PWM0 (O)	PWM0 (O)
7	D4	PWM1 (O)	PWM1 (O)	PWM1 (O)	PWM1 (O)	PWM1 (O)
8	E5	CVDD	CVDD	CVDD	CVDD	CVDD
9	D2	NC1	NC1	NC1	NC1	NC1
10	D1	CGND	CGND	CGND	CGND	CGND
11	D3	NC2	NC2	NC2	NC2	NC2
12	E4	SHIF (I)	SHIF (I)	SHIF (I)	SHIF (I)	SHIF (I)
13	E2	MD[1] (I)	MD[1] (I)	MD[1] (I)	MD[1] (I)	MD[1] (I)
14	E1	MD[0] (I)	MD[0] (I)	MD[0] (I)	MD[0] (I)	MD[0] (I)
15	E3	NC3	NC3	NC3	NC3	NC3
16	F5	NC4	NC4	NC4	NC4	NC4
17	F2	NC5	NC5	NC5	NC5	NC5
18	F1	NC6	NC6	NC6	NC6	NC6
19	F3	NC7	NC7	NC7	NC7	NC7
20	F4	NC8	NC8	NC8	NC8	NC8
21	G2	NC9	NC9	NC9	NC9	NC9
22	G1	EGND	EGND	EGND	EGND	EGND
23	G3	BATT-FAULT#	BATT-FAULT#	BATT-FAULT#	BATT-FAULT#	BATT-FAULT#
24	G4	NC10	NC10	NC10	NC10	NC10
25	H2	NC11	NC11	NC11	NC11	NC11
26	H1	NC12	NC12	NC12	NC12	NC12
27	H3	A[15] (I)	A[15] (I)	BANK	-	A[15] (I)
28	H4	A[14] (I)	A[14] (I)	A[12] (I)	-	A[14] (I)
29	J2	A[13] (I)	A[13] (I)	A[11] (I)	-	A[13] (I)
30	J1	A[12] (I)	A[12] (I)	A[10] (I)	-	A[12] (I)
31	J3	A[11] (I)	A[11] (I)	A[9] (I)	-	A[11] (I)
32	H5	EVDD	EVDD	EVDD	EVDD	EVDD

Table 2-7: Pins Sorted by Ball Number

Pin	BGA Ball	SH-4 IF	NEC-4121 IF	Toshiba-3922 IF	PCI IF	SA-1110 IF
33	K1	A[10] (I)	A[10] (I)	A[8] (I)	-	A[10] (I)
34	K2	A[9] (I)	A[9] (I)	A[7] (I)	-	A[9] (I)
35	J4	A[8] (I)	A[8] (I)	A[6] (I)	-	A[8] (I)
36	K3	A[7] (I)	A[7] (I)	A[5] (I)	-	A[7] (I)
37	L1	A[6] (I)	A[6] (I)	A[4] (I)	PAR (O)	A[6] (I)
38	L2	A[5] (I)	A[5] (I)	A[3] (I)	IDSEL (I)	A[5] (I)
39	J5	A[4] (I)	A[4] (I)	A[2] (I)	CLKRUN(IO)	A[4] (I)
40	K4	A[3] (I)	A[3] (I)	A[1] (I)	SERR	A[3] (I)
41	M1	POR# (I)	POR# (I)	POR# (I)	POR# (I)	POR# (I)
42	M2	A[2] (I)	A[2] (I)	A[0] (I)	PERR	A[2] (I)
43	L3	D[31] (IO)	D[31] (IO)	D[31] (IO)	D[31] (IO)	D[31] (IO)
44	N1	D[30] (IO)	D[30] (IO)	D[30] (IO)	D[30] (IO)	D[30] (IO)
45	N2	D[29] (IO)	D[29] (IO)	D[29] (IO)	D[29] (IO)	D[29] (IO)
46	P1	EGND33	EGND33	EGND33	EGND33	EGND33
47	P2	D[28] (IO)	D[28] (IO)	D[28] (IO)	D[28] (IO)	D[28] (IO)
48	K7	EVDD33	EVDD33	EVDD33	EVDD33	EVDD33
49	M3	D[27] (IO)	D[27] (IO)	D[27] (IO)	D[27] (IO)	D[27] (IO)
50	N3	D[26] (IO)	D[26] (IO)	D[26] (IO)	D[26] (IO)	D[26] (IO)
51	P3	D[25] (IO)	D[25] (IO)	D[25] (IO)	D[25] (IO)	D[25] (IO)
52	L4	D[24] (IO)	D[24] (IO)	D[24] (IO)	D[24] (IO)	D[24] (IO)
53	K5	BVDD	BVDD	BVDD	BVDD	BVDD
54	N4	WE[3] (I)	BE[3] (I)	WE[3] (I)	C/BE[3] (I)	-
55	P4	BGND	BGND	BGND	BGND	BGND
56	M4	DACK (I)	MD4	MD4	MD4	MD4
57	L5	D[23] (IO)	D[23] (IO)	D[23] (IO)	D[23] (IO)	D[23] (IO)
58	N5	D[22] (IO)	D[22] (IO)	D[22] (IO)	D[22] (IO)	D[22] (IO)
59	P5	D[21] (IO)	D[21] (IO)	D[21] (IO)	D[21] (IO)	D[21] (IO)
60	M5	D[20] (IO)	D[20] (IO)	D[20] (IO)	D[20] (IO)	D[20] (IO)
61	K6	EVDD	EVDD	EVDD	EVDD	EVDD
62	N6	D[19] (IO)	D[19] (IO)	D[19] (IO)	D[19] (IO)	D[19] (IO)
63	P6	EGND	EGND	EGND	EGND	EGND
64	M6	D[18] (IO)	D[18] (IO)	D[18] (IO)	D[18] (IO)	D[18] (IO)
65	L6	D[17] (IO)	D[17] (IO)	D[17] (IO)	D[17] (IO)	D[17] (IO)

Table 2-7: Pins Sorted by Ball Number

Pin	BGA Ball	SH-4 IF	NEC-4121 IF	Toshiba-3922 IF	PCI IF	SA-1110 IF
66	N7	D[16] (IO)	D[16] (IO)	D[16] (IO)	D[16] (IO)	D[16] (IO)
67	P7	WE[2] (I)	BE[2] (I)	WE[2] (I)	C/BE[2] (I)	-
68	M7	CS# (I)	CS# (I)	CS# (I)	FRAME# (I)	nCS (I)
69	L7	RD/WR# (I)	RD# (I)	RD# (I)	IRDY# (I)	nOE (I)
70	N8	RDY/WAIT(O)	RDY# (O)	WAIT# (O)	TRDY# (O)	RDY(O)
71	P8	DRAK (I)	MD3	MD3	MD3	MD3
72	M8	CKIO (I)	-	CKIO (I)	CLK (I)	-
73	L8	BS# (I)		ALE (I)	DEVSEL#(O)	-
74	N9	D[15] (IO)	D[15] (IO)	D[15] (IO)	D[15] (IO)	D[15] (IO)
75	P9	D[14] (IO)	D[14] (IO)	D[14] (IO)	D[14] (IO)	D[14] (IO)
76	M9	D[13] (IO)	D[13] (IO)	D[13] (IO)	D[13] (IO)	D[13] (IO)
77	K8	BVDD	BVDD	BVDD	BVDD	BVDD
78	P10	BGND	BGND	BGND	BGND	BGND
79	N10	D[12] (IO)	D[12] (IO)	D[12] (IO)	D[12] (IO)	D[12] (IO)
80	L9	D[11] (IO)	D[11] (IO)	D[11] (IO)	D[11] (IO)	D[11] (IO)
81	M10	D[10] (IO)	D[10] (IO)	D[10] (IO)	D[10] (IO)	D[10] (IO)
82	P11	D[9] (IO)	D[9] (IO)	D[9] (IO)	D[9] (IO)	D[9] (IO)
83	N11	D[8] (IO)	D[8] (IO)	D[8] (IO)	D[8] (IO)	D[8] (IO)
84	K9	WE[1] (I)	BE[1] (I)	WE[1] (I)	C/BE[1] (I)	-
85	L10	DREQ# (O)	WR# (I)	WR# (I)	STOP# (O)	nWR (I)
86	P12	D[7] (IO)	D[7] (IO)	D[7] (IO)	D[7] (IO)	D[7] (IO)
87	N12	D[6] (IO)	D[6] (IO)	D[6] (IO)	D[6] (IO)	D[6] (IO)
88	M11	D[5] (IO)	D[5] (IO)	D[5] (IO)	D[5] (IO)	D[5] (IO)
89	P13	D[4] (IO)	D[4] (IO)	D[4] (IO)	D[4] (IO)	D[4] (IO)
90	N13	D[3] (IO)	D[3] (IO)	D[3] (IO)	D[3] (IO)	D[3] (IO)
91	P14	BGND	BGND	BGND	BGND	BGND
92	N14	D[2] (IO)	D[2] (IO)	D[2] (IO)	D[2] (IO)	D[2] (IO)
93	H10	BVDD	BVDD	BVDD	BVDD	BVDD
94	M12	D[1] (IO)	D[1] (IO)	D[1] (IO)	D[1] (IO)	D[1] (IO)
95	M13	D[0] (IO)	D[0] (IO)	D[0] (IO)	D[0] (IO)	D[0] (IO)
96	M14	WE[0] (I)	BE[0] (I)	WE[0] (I)	C/BE[0] (I)	-
97	L11	PMCLK (I)	PMCLK (I)	PMCLK (I)	PMCLK (I)	PMCLK (I)
98	K10	CVDD	CVDD	CVDD	CVDD	CVDD

Table 2-7: Pins Sorted by Ball Number

Pin	BGA Ball	SH-4 IF	NEC-4121 IF	Toshiba-3922 IF	PCI IF	SA-1110 IF
99	L13	IREQ# (O)	IREQ# (O)	IREQ# (O)	IREQ# (O)	IREQ# (O)
100	L14	CGND	CGND	CGND	CGND	CGND
101	L12	MD[2] (I)	MD[2] (I)	MD[2] (I)	MD[2] (I)	MD[2] (I)
102	K11	A[16] (I)	A[16] (I)	-	-	A[16] (I)
103	K13	PDWN# (I)	PDWN# (I)	PDWN# (I)	PDWN# (I)	PDWN# (I)
104	K14	A[17] (I)	A[17] (I)	-	-	A[17] (I)
105	K12	A[18] (I)	A[18] (I)	-	-	A[18] (I)
106	J10	CVDD	CVDD	CVDD	CVDD	CVDD
107	J13	A[19] (I)	A[19] (I)	-	-	A[19] (I)
108	J14	CGND	CGND	CGND	CGND	CGND
109	J12	A[20] (I)	A[20] (I)	-	-	A[20] (I)
110	J11	A[21] (I)	A[21] (I)	-	-	A[21] (I)
111	H13	A[22] (I)	A[22] (I)	-	-	A[22] (I)
112	H14	A[23] (I)	A[23] (I)	-	-	A[23] (I)
113	H12	A[24] (I)	BANK[1] (I)	-	-	A[24] (I)
114	H11	A[25] (I)	BANK[0] (I)	-	-	A[25] (I)
115	G13	FD[23] (O)	FD[23] (O)	FD[23] (O)	FD[23] (O)	FD[23] (O)
116	G14	FD[22] (O)	FD[22] (O)	FD[22] (O)	FD[22] (O)	FD[22] (O)
117	G12	FD[21] (O)	FD[21] (O)	FD[21] (O)	FD[21] (O)	FD[21] (O)
118	G11	FD[20] (O)	FD[20] (O)	FD[20] (O)	FD[20] (O)	FD[20] (O)
119	F13	FD[19] (O)	FD[19] (O)	FD[19] (O)	FD[19] (O)	FD[19] (O)
120	F14	FGND	FGND	FGND	FGND	FGND
121	F12	FD[18] (O)	FD[18] (O)	FD[18] (O)	FD[18] (O)	FD[18] (O)
122	G10	FVDD	FVDD	FVDD	FVDD	FVDD
123	E14	FD[17] (O)	FD[17] (O)	FD[17] (O)	FD[17] (O)	FD[17] (O)
124	E13	FD[16] (O)	FD[16] (O)	FD[16] (O)	FD[16] (O)	FD[16] (O)
125	F11	FD[15] (O)	FD[15] (O)	FD[15] (O)	FD[15] (O)	FD[15] (O)
126	E12	FD[14] (O)	FD[14] (O)	FD[14] (O)	FD[14] (O)	FD[14] (O)
127	D14	FD[13] (O)	FD[13] (O)	FD[13] (O)	FD[13] (O)	FD[13] (O)
128	D13	FD[12] (O)	FD[12] (O)	FD[12] (O)	FD[12] (O)	FD[12] (O)
129	F10	FVDD	FVDD	FVDD	FVDD	FVDD
130	E11	FD[11] (O)	FD[11] (O)	FD[11] (O)	FD[11] (O)	FD[11] (O)
131	C14	FGND	FGND	FGND	FGND	FGND

Table 2-7: Pins Sorted by Ball Number

Pin	BGA Ball	SH-4 IF	NEC-4121 IF	Toshiba-3922 IF	PCI IF	SA-1110 IF
132	C13	FD[10] (O)	FD[10] (O)	FD[10] (O)	FD[10] (O)	FD[10] (O)
133	D12	FD[9] (O)	FD[9] (O)	FD[9] (O)	FD[9] (O)	FD[9] (O)
134	B14	FD[8] (O)	FD[8] (O)	FD[8] (O)	FD[8] (O)	FD[8] (O)
135	B13	FD[7] (O)	FD[7] (O)	FD[7] (O)	FD[7] (O)	FD[7] (O)
136	A14	FD[6] (O)	FD[6] (O)	FD[6] (O)	FD[6] (O)	FD[6] (O)
137	A13	FD[5] (O)	FD[5] (O)	FD[5] (O)	FD[5] (O)	FD[5] (O)
138	E8	FVDD	FVDD	FVDD	FVDD	FVDD
139	C12	FD[4] (O)	FD[4] (O)	FD[4] (O)	FD[4] (O)	FD[4] (O)
140	B12	FD[3] (O)	FD[3] (O)	FD[3] (O)	FD[3] (O)	FD[3] (O)
141	A12	FGND	FGND	FGND	FGND	FGND
142	D11	FD[2] (O)	FD[2] (O)	FD[2] (O)	FD[2] (O)	FD[2] (O)
143	E10	FD[1] (O)	FD[1] (O)	FD[1] (O)	FD[1] (O)	FD[1] (O)
144	B11	FD[0] (O)	FD[0] (O)	FD[0] (O)	FD[0] (O)	FD[0] (O)
145	A11	FDE (O)	FDE (O)	FDE (O)	FDE (O)	FDE (O)
146	C11	FVSYNC(O)	FVSYNC (O)	FVSYNC (O)	FVSYNC (O)	FVSYNC (O)
147	D10	FDI (O)	FDI (O)	FDI (O)	FDI (O)	FDI (O)
148	B10	FHSYNC(O)	FHSYNC(O)	FHSYNC(O)	FHSYNC(O)	FHSYNC(O)
149	A10	CGND	CGND	CGND	CGND	CGND
150	C10	FSCLK (O)	FSCLK (O)	FSCLK (O)	FSCLK (O)	FSCLK (O)
151	E9	CVDD	CVDD	CVDD	CVDD	CVDD
152	B9	ENVDD (O)	ENVDD (O)	ENVDD (O)	ENVDD (O)	ENVDD (O)
153	A9	ENCTL (O)	ENCTL (O)	ENCTL (O)	ENCTL (O)	ENCTL (O)
154	C9	ENVEE (O)	ENVEE (O)	ENVEE (O)	ENVEE (O)	ENVEE (O)
155	D9	AVDDD	AVDDD	AVDDD	AVDDD	AVDDD
156	B8	VREFIO	VREFIO	VREFIO	VREFIO	VREFIO
157	A8	AGNDD	AGNDD	AGNDD	AGNDD	AGNDD
158	C8	COMPC	COMPC	COMPC	COMPC	COMPC
159	D8	FSADJ (I)	FSADJ (I)	FSADJ (I)	FSADJ (I)	FSADJ (I)
160	B7	AGNDC	AGNDC	AGNDC	AGNDC	AGNDC
161	A7	AB (O)	AB (O)	AB (O)	AB (O)	AB (O)
162	C7	AVDDC	AVDDC	AVDDC	AVDDC	AVDDC
163	D7	AVDDC	AVDDC	AVDDC	AVDDC	AVDDC
164	B6	AG (O)	AG (O)	AG (O)	AG (O)	AG (O)

Table 2-7: Pins Sorted by Ball Number

Pin	BGA Ball	SH-4 IF	NEC-4121 IF	Toshiba-3922 IF	PCI IF	SA-1110 IF
165	A6	AGNDC	AGNDC	AGNDC	AGNDC	AGNDC
166	C6	AR (O)	AR (O)	AR (O)	AR (O)	AR (O)
167	E7	AVDDC	AVDDC	AVDDC	AVDDC	AVDDC
168	A5	AGNDC	AGNDC	AGNDC	AGNDC	AGNDC
169	B5	AGND1	AGND1	AGND1	AGND1	AGND1
170	D6	AVDD1	AVDD1	AVDD1	AVDD1	AVDD1
171	C5	AVDD2	AVDD2	AVDD2	AVDD2	AVDD2
172	A4	AGND2	AGND2	AGND2	AGND2	AGND2
173	B4	AGND3	AGND3	AGND3	AGND3	AGND3
174	E6	AVDD3	AVDD3	AVDD3	AVDD3	AVDD3
175	D5	L2CLK (I)	L2CLK (I)	L2CLK (I)	L2CLK (I)	L2CLK (I)
176	A3	VDDOSC3.3	VDDOSC3.3	VDDOSC3.3	VDDOSC3.3	VDDOSC3.3
177	B3	OSCI (I)	OSCI (I)	OSCI (I)	OSCI (I)	OSCI (I)
178	C4	OSCO (O)	OSCO (O)	OSCO (O)	OSCO (O)	OSCO (O)
179	A2	VGNDOSC	VGNDOSC	VGNDOSC	VGNDOSC	VGNDOSC
180	B2	L3CLK (I)	L3CLK (I)	L3CLK (I)	L3CLK (I)	L3CLK (I)

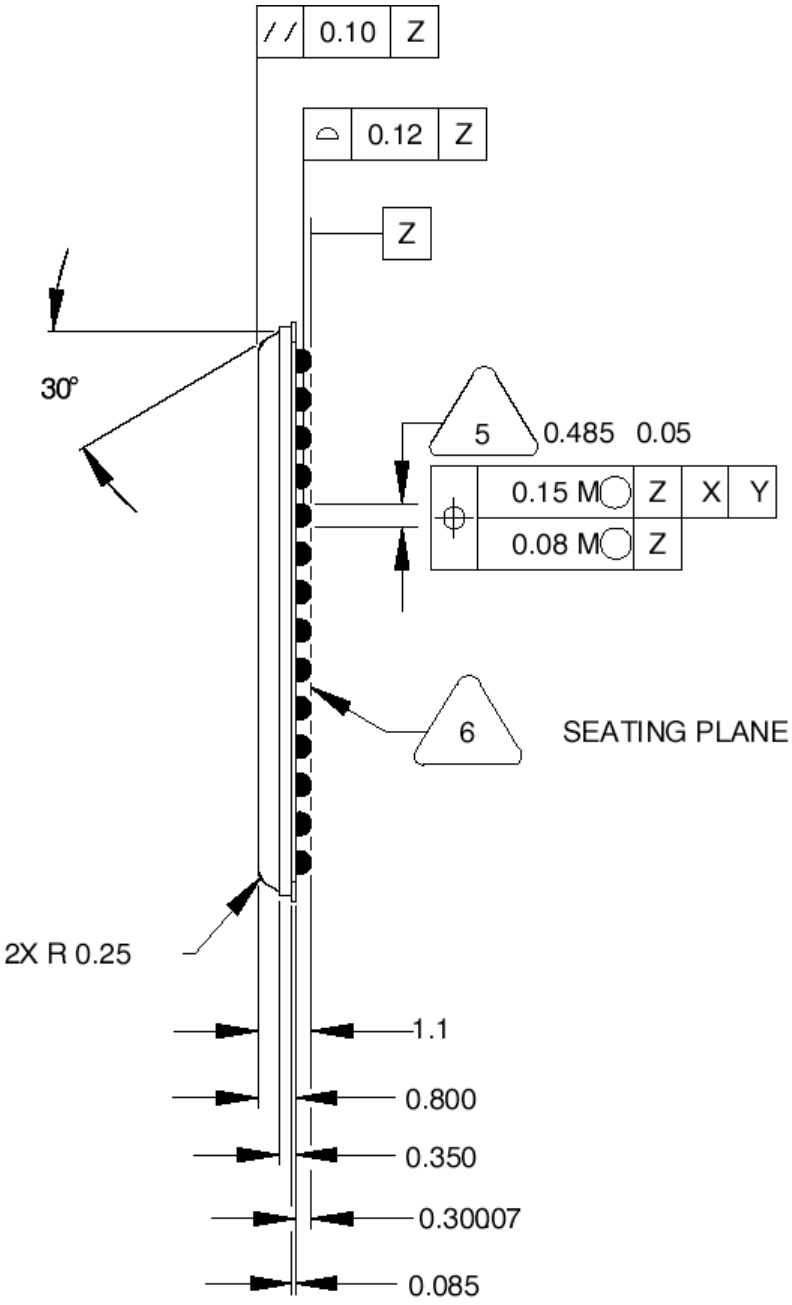


Figure 2-8: MQ-200 Micro BGA Package Side View

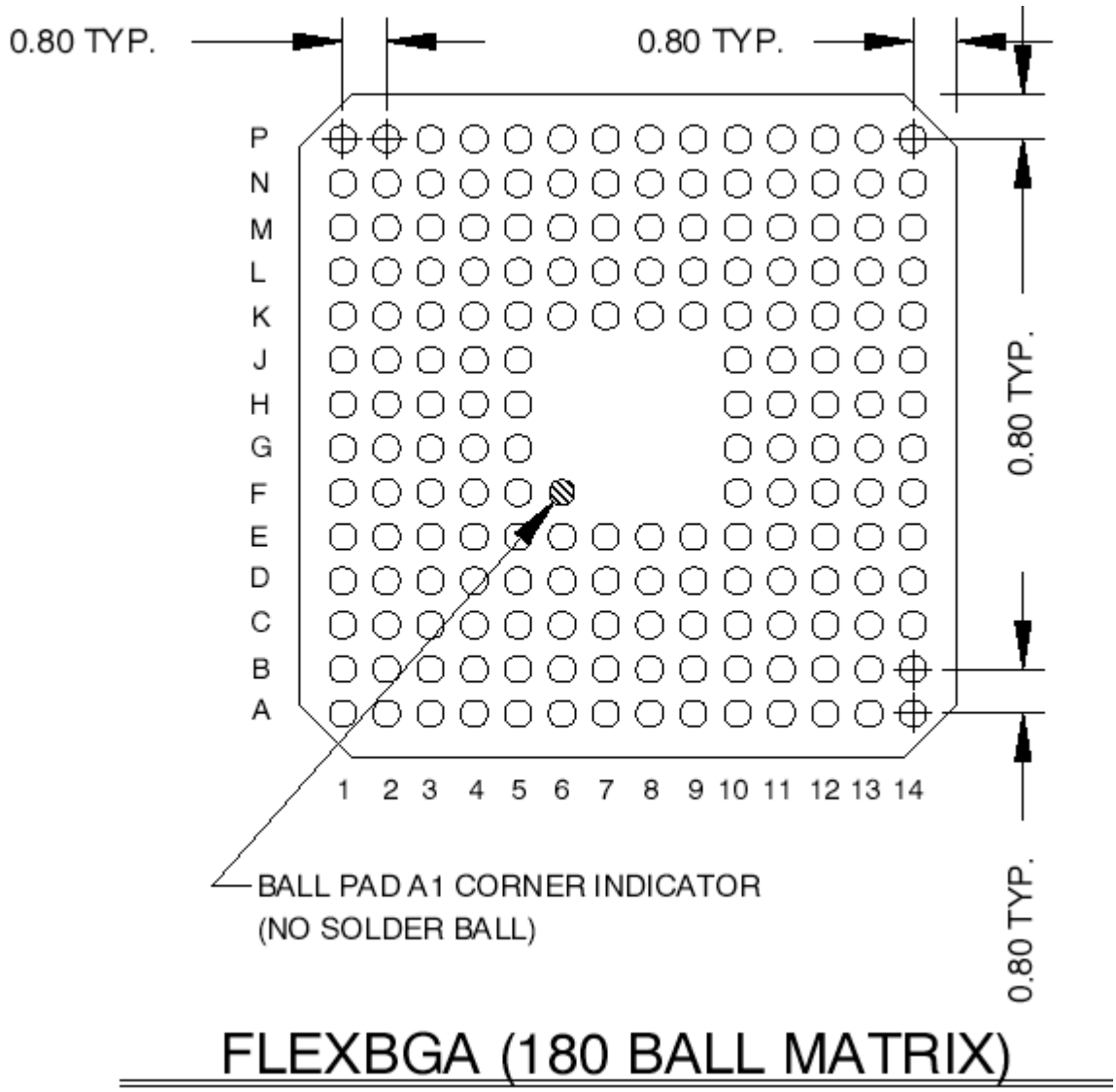


Figure 2-9: MQ-200 Micro BGA Package Bottom View

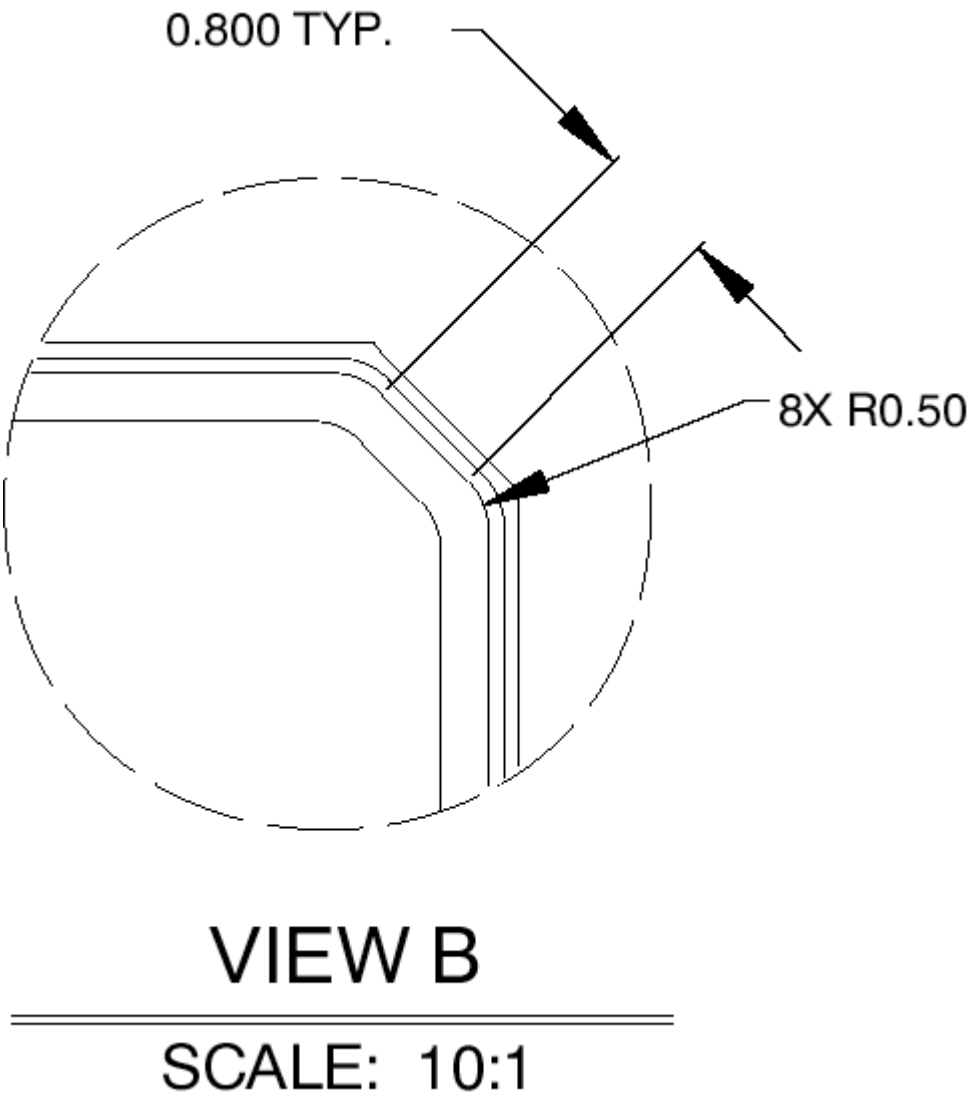


Figure 2-10: MQ-200 Micro BGA Package Package Corner View A

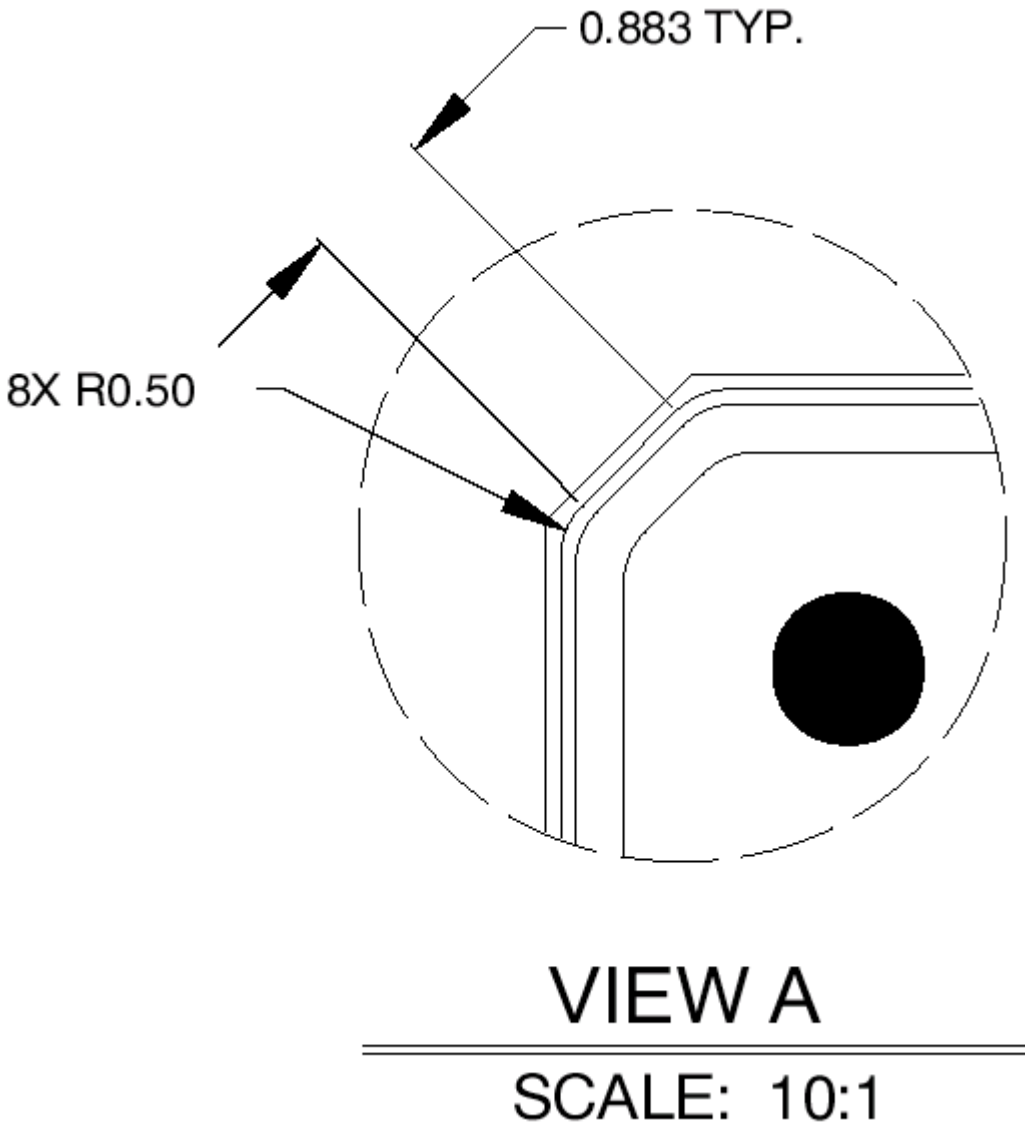


Figure 2-11: MQ-200 Micro BGA Package Package Corner View A (refernce ball 1).

Chapter 3

Functional Description

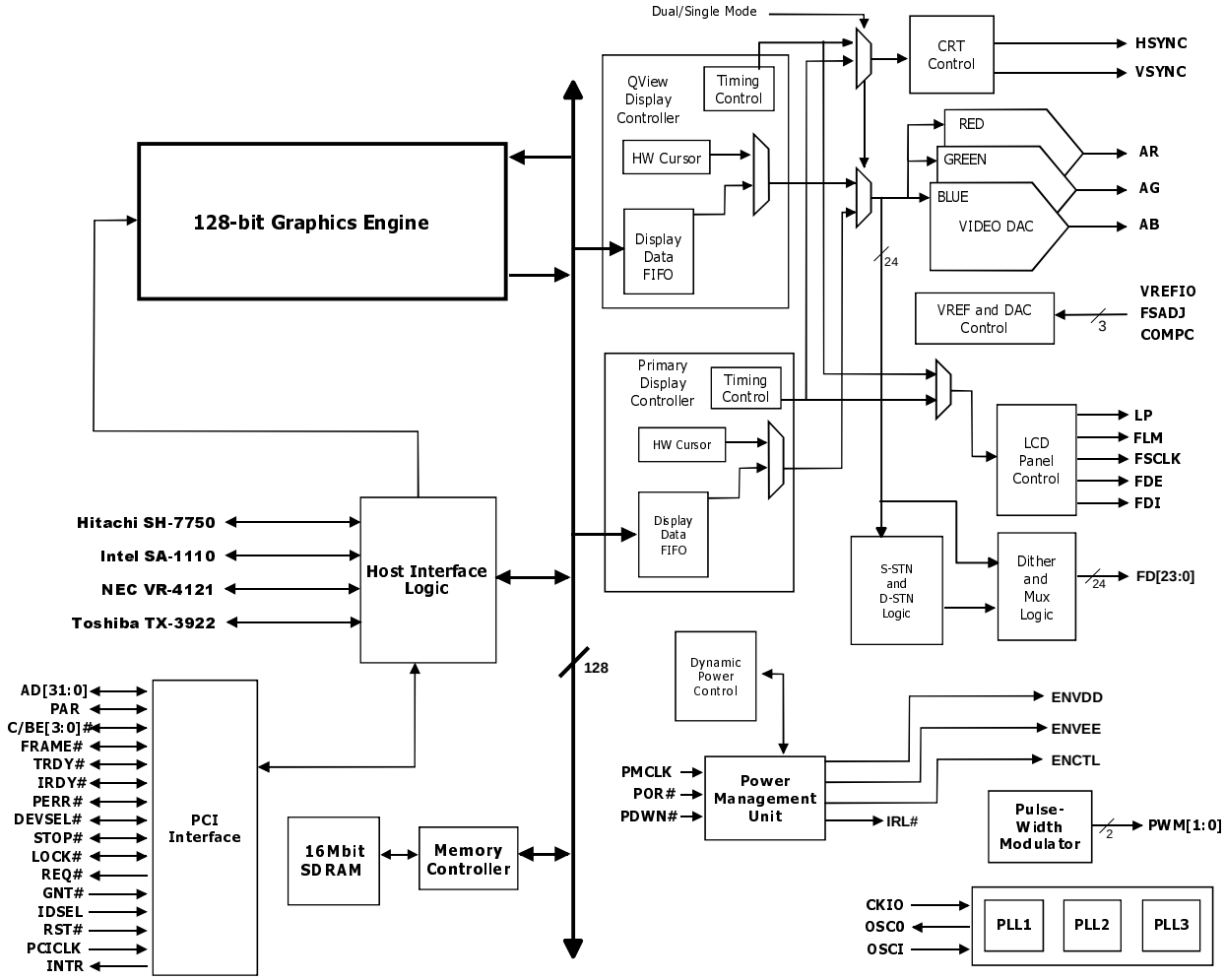


Figure 3-1: MQ-200 Block Diagram

The main functional components of the MQ-200 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.

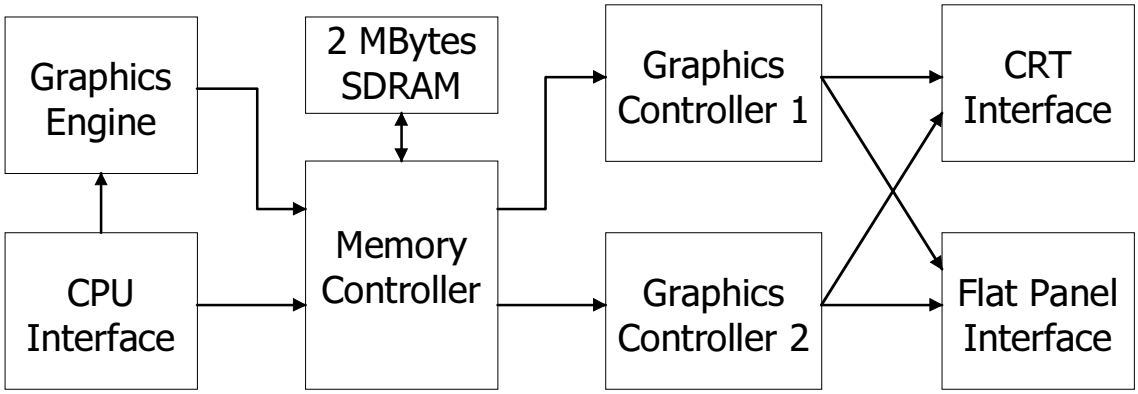


Figure 3-2: MQ-200 Functional Modules

The Power Management Unit (PMU) takes control of the MQ-200 following reset and manages the power-up and power-down sequencing. It is must be active in order for the MQ-200 to operate, and has it's own clock signal, allowing it to operate even when the rest of the MQ-200 is shut down. The PMU provides the means for software to selectively shut down portions of the MQ-200 to optimize power consumption.

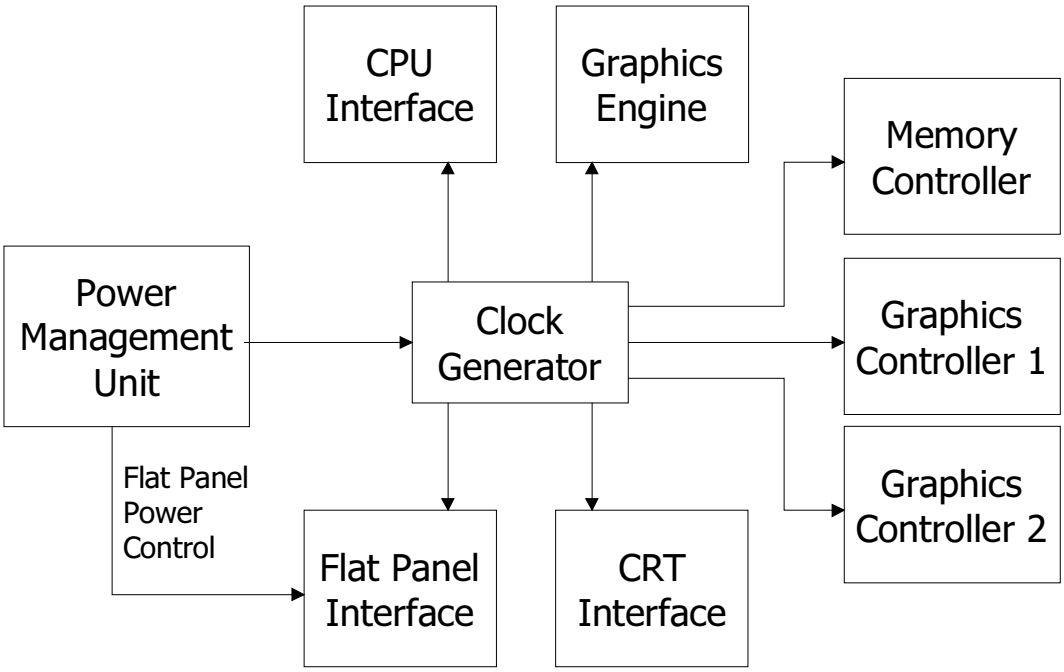


Figure 3-3: MQ-200 Clocking Scheme

CPU Interface

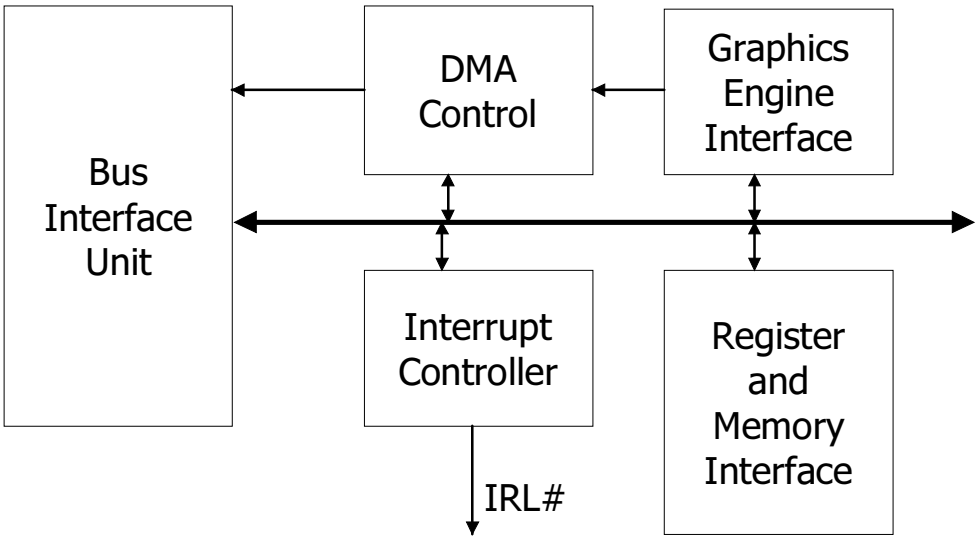


Figure 3-4: MQ-200 CPU Interface

Overview

The MQ-200 connects directly to the Hitachi SH-7750, Intel SA-1110, NEC VR-4121 and Toshiba Tx-3922 processors. In addition, the MQ-200 offers PCI V 2.1 compliant bus interface. The bus interface is configured upon power-on reset (POR# is low) for operation with one of the four processors listed above or the PCI bus depending on the state of the SHIF, MD[0:2], MD3 and MD4 pins.

Figures 3E and 3F show the pin-to-pin connections for the MQ-200 to the SH-7750 and to the SH-7709 configurations.

SH-7750 to MQ-200 Interface

Interfacing the MQ-200 to the SH-7750 CPU, the system designer must choose between the following options:

- CS1# or CS4#
- MQ-200 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

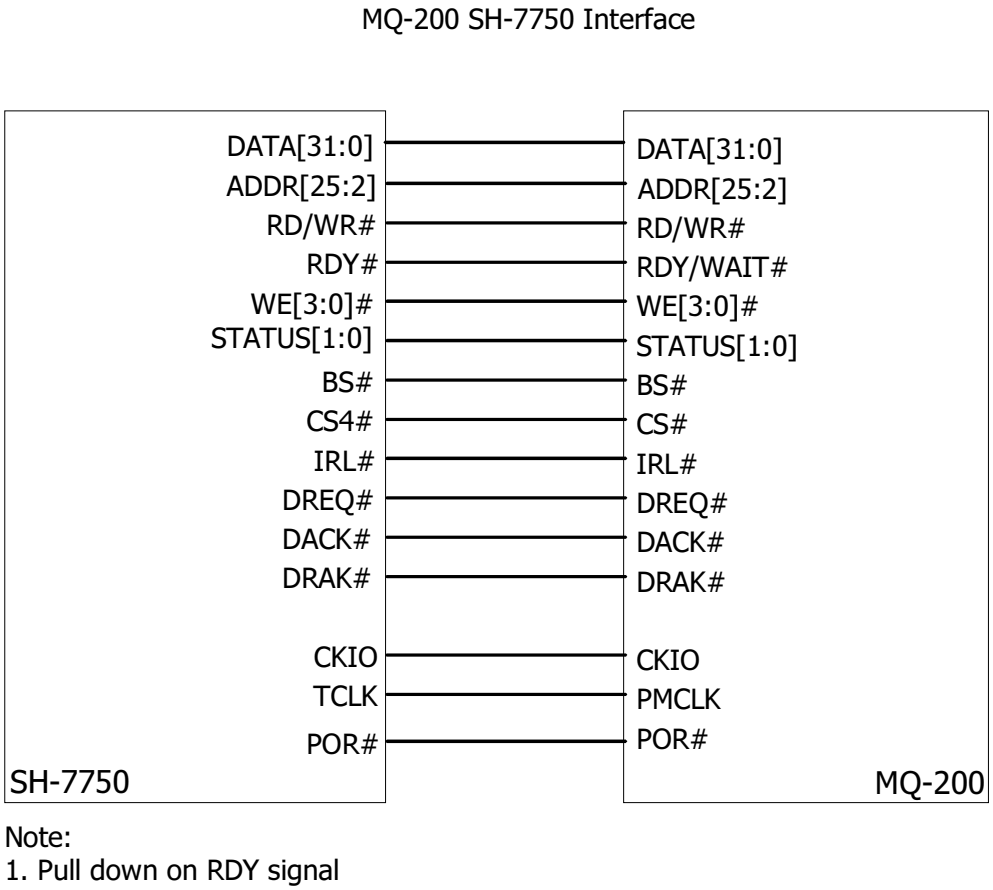


Figure 3-5: SH-7750 to MQ-200 Interface

Intel SA-1110 to MQ-200 Interface

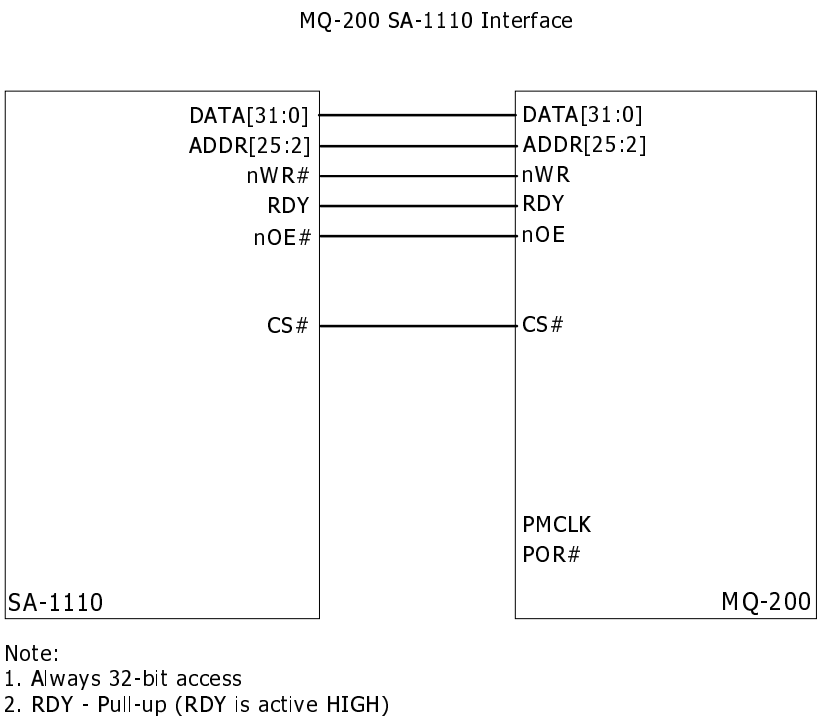


Figure 3-6: SA-1110 to MQ-200 Interface

NEC VR-4121 Interface

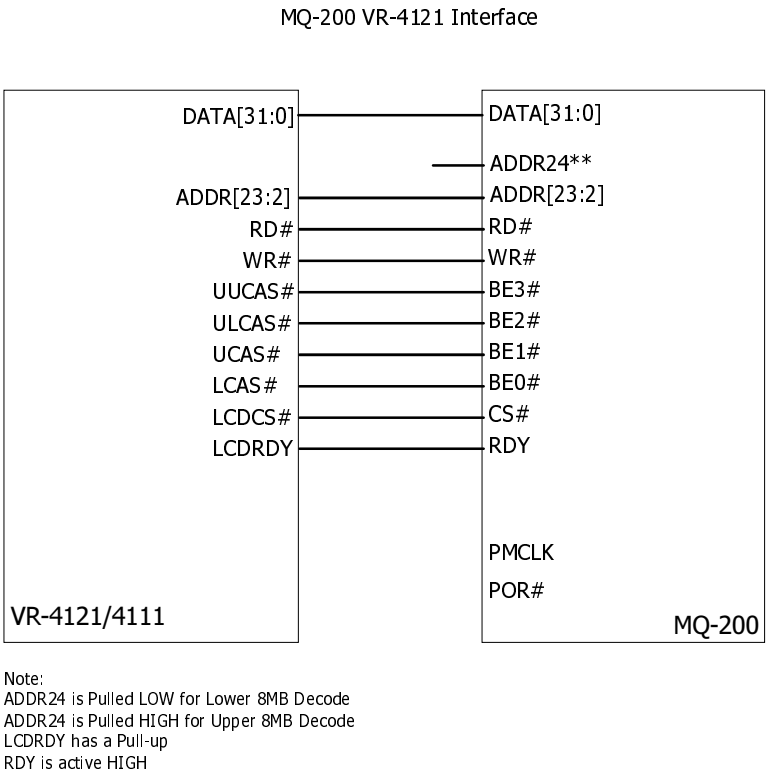


Figure 3-7: VR-4121 to MQ-200 Interface

Toshiba Tx-3922 Interface

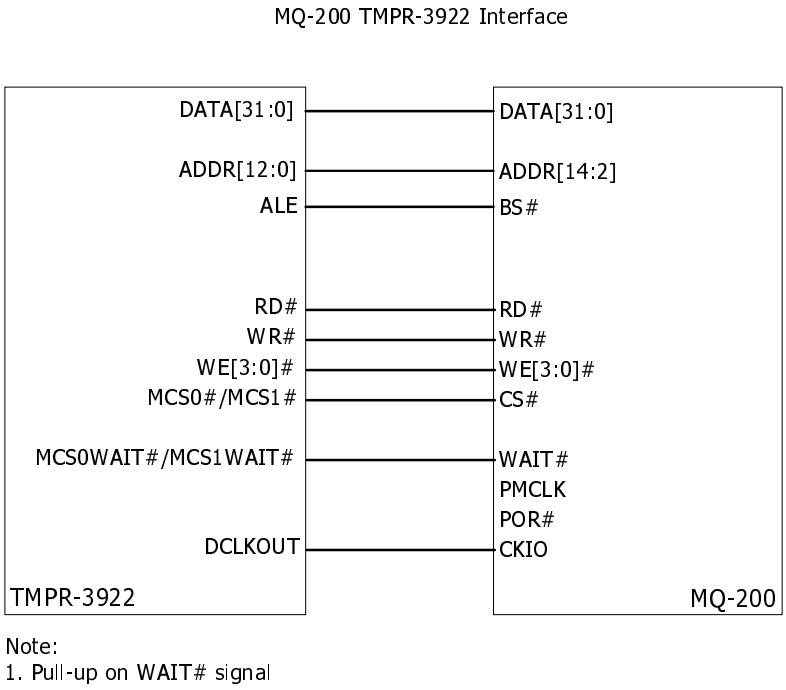


Figure 3-8: Tx-3922 to MQ-200 Interface

Interrupt Controller

There are 16 possible interrupt sources in the MQ-200:

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.

Clocks and Oscillator

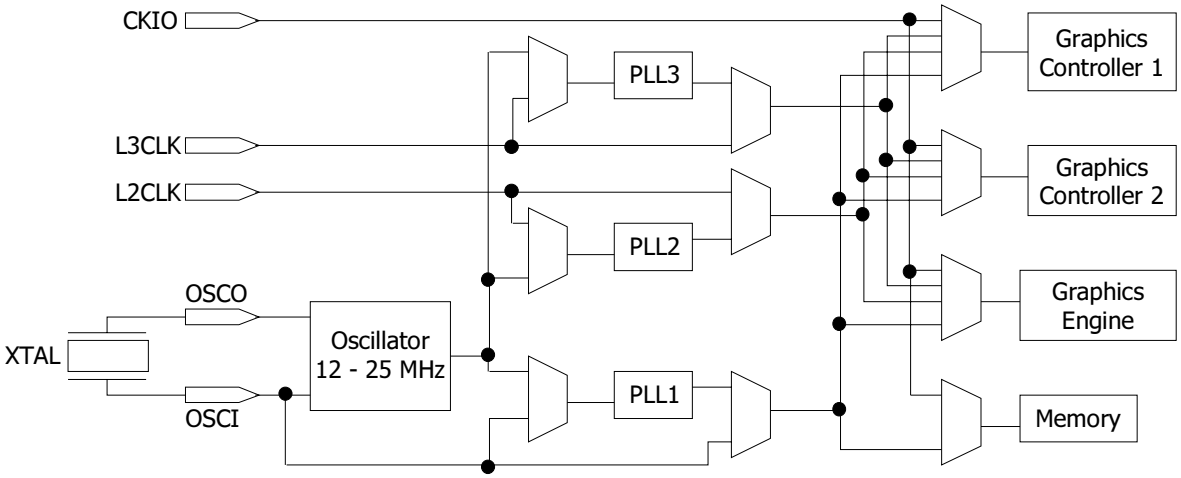


Figure 3-9: Clocks and Oscillators

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 selection 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) in the SH-7750 mode of operation.

The CPU interface uses the CPU bus clock, CKIO in the Hitachi SH-7750, Intel SA-1110, NEC VR-4121 and Toshiba Tx-3922 modes of operation.

The Power Management Unit's clock is PMCLK. The frequency of PMCLK is expected to be 16.384KHz

Graphics Engine

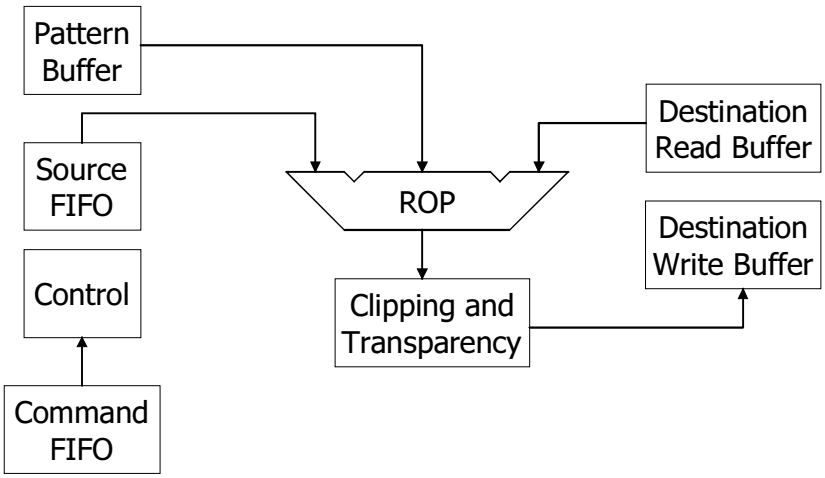


Figure 3-10: Graphics Engine

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, 16 and 32 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.

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.

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.

Bresenham Line Draw

Full Bresenham line drawing algorithm including arbitrary line draw functionality is implemented in the MQ-200. This hardware implementation is designed with consideration to Windows CE driver architecture to achieve the most optimized performance. The MQ-200 is capable of drawing any non-

patterned line of a single pixel width between any 2 points on scrren and the line drawn is fully compatible with internal Windows CE line drawing implementation. The MQ-200 line draw implementation also supports ROP functions which means that when a line draw command is issued, it can be combined with a destination ROP.

Monochrome to Color Expansion

The Graphics Engine supports pixel resolutions of 8, 16 and 32-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.

Graphics Engine Register Set

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

Memory Subsystem

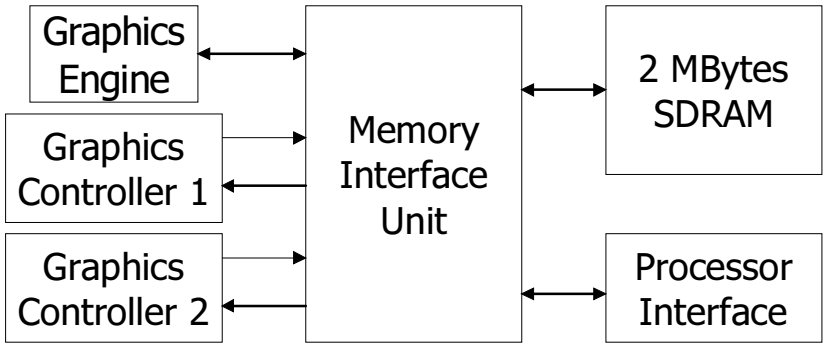


Figure 3-11: Memory Subsystem

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.

Display Section

The MQ-200 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.

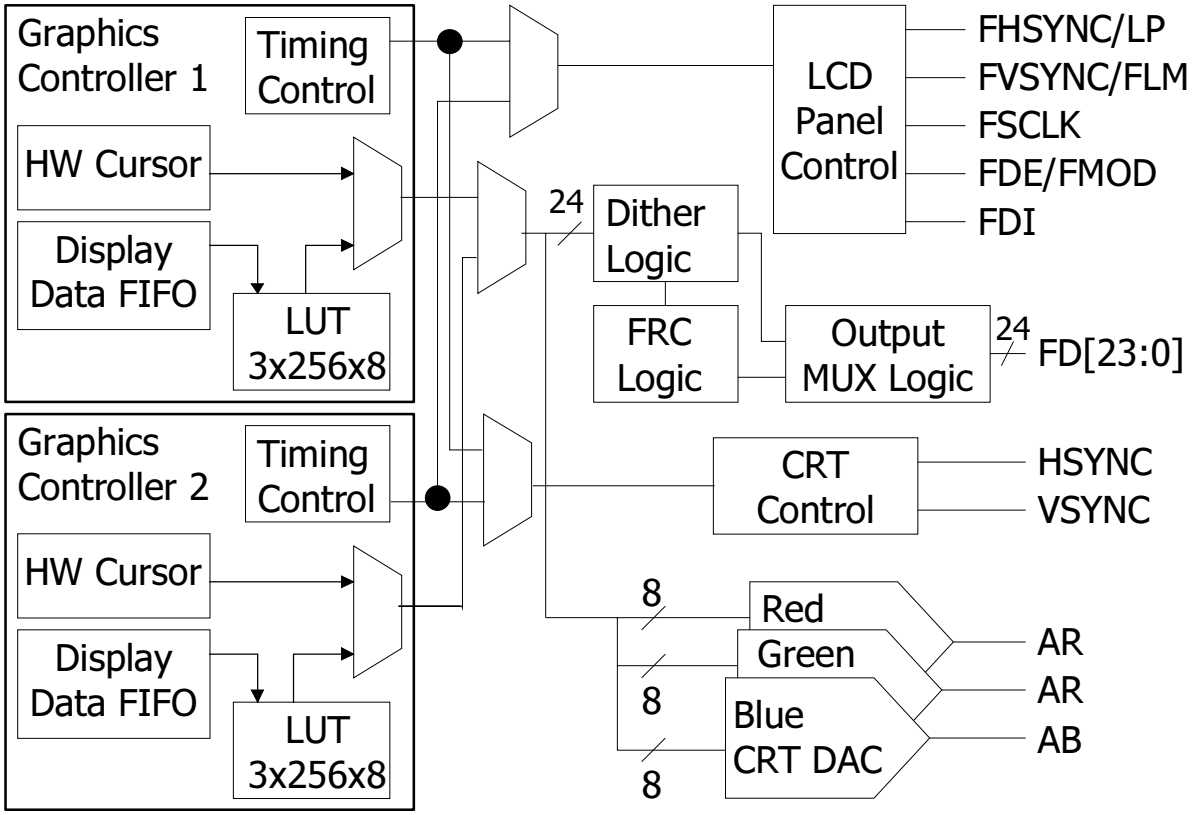


Figure 3-12: Display Diagram

The MQ-200 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 3-13: MQ-200 display configuration options for LCD and CRT display utilizing QView

Display Configuration	CRT Size	CRT Color Depth (bpp)	LCD Size	LCD Color Depth (bpp)
CRT + TFT/S-STN, Simultaneous, Same Image	XGA	16	XGA	16
CRT + TFT/S-STN, Simultaneous, Same Image	XGA	8	XGA	8
CRT + TFT/S-STN, Simultaneous, Same Image	SVGA	32	SVGA	32
CRT + TFT/S-STN, Simultaneous, Same Image	SVGA	24	SVGA	24
CRT + TFT/S-STN, Simultaneous, Same Image	SVGA	16	SVGA	16
CRT + TFT/S-STN, Simultaneous, Same Image	SVGA	8	SVGA	8
CRT + TFT/S-STN, Simultaneous, Same Image	VGA	32	VGA	32
CRT + TFT/S-STN, Simultaneous, Same Image	VGA	24	VGA	24
CRT + TFT/S-STN, Simultaneous, Same Image	VGA	16	VGA	16
CRT + TFT/S-STN, Simultaneous, Same Image	VGA	8	VGA	8
CRT + D-STN, Simultaneous, Same Image	XGA	16	XGA	16
CRT + D-STN, Simultaneous, Same Image	XGA	8	XGA	8
CRT + D-STN, Simultaneous, Same Image	SVGA	32	SVGA	32
CRT + D-STN, Simultaneous, Same Image	SVGA	24	SVGA	24
CRT + D-STN, Simultaneous, Same Image	SVGA	16	SVGA	16
CRT + D-STN, Simultaneous, Same Image	SVGA	8	SVGA	8
CRT + D-STN, Simultaneous, Same Image	VGA	32	VGA	32
CRT + D-STN, Simultaneous, Same Image	VGA	24	VGA	24
CRT + D-STN, Simultaneous, Same Image	VGA	16	VGA	16
CRT + D-STN, Simultaneous, Same Image	VGA	8	VGA	8
CRT + TFT/S-STN, Qview - Dual Image	XGA	16	VGA	8
CRT + TFT/S-STN, Qview - Dual Image	XGA	8	SVGA	16
CRT + TFT/S-STN, Qview - Dual Image	SVGA	24	VGA	8
CRT + D-STN, Qview - Dual Image	XGA	16	VGA	8
CRT + D-STN, Qview - Dual Image	XGA	8	SVGA	16
CRT + D-STN, Qview - Dual Image	SVGA	24	VGA	8
CRT + TFT/S-STN, Qview - Dual Image	SVGA	16	SVGA	16
CRT + D-STN, Qview - Dual Image	SVGA	16	SVGA	16
CRT + TFT/S-STN, Qview - Dual Image	SVGA	8	SVGA	8

Table 3-13: MQ-200 display configuration options for LCD and CRT display utilizing QView

CRT + D-STN, Qview - Dual Image	SVGA	8	SVGA	8
CRT + TFT/S-STN, Qview - Dual Image	VGA	32	VGA	16
CRT + D-STN, Qview - Dual Image	VGA	32	VGA	16
CRT + TFT/S-STN, Qview - Dual Image	VGA	24	VGA	24
CRT + D-STN, Qview - Dual Image	VGA	24	VGA	24

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.

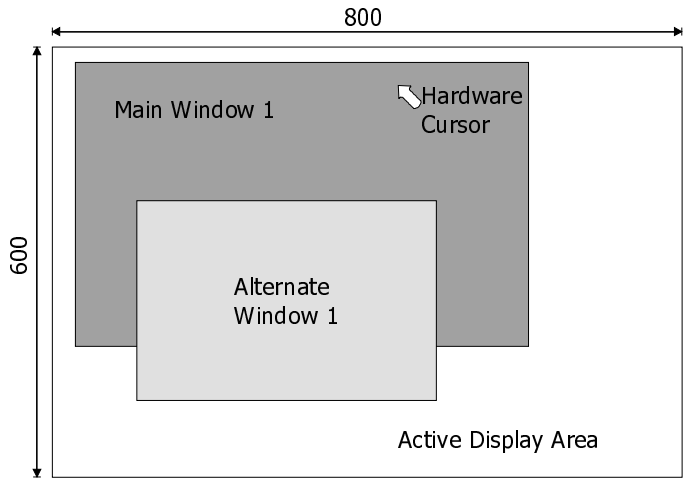


Figure 3-14: Active Display Area

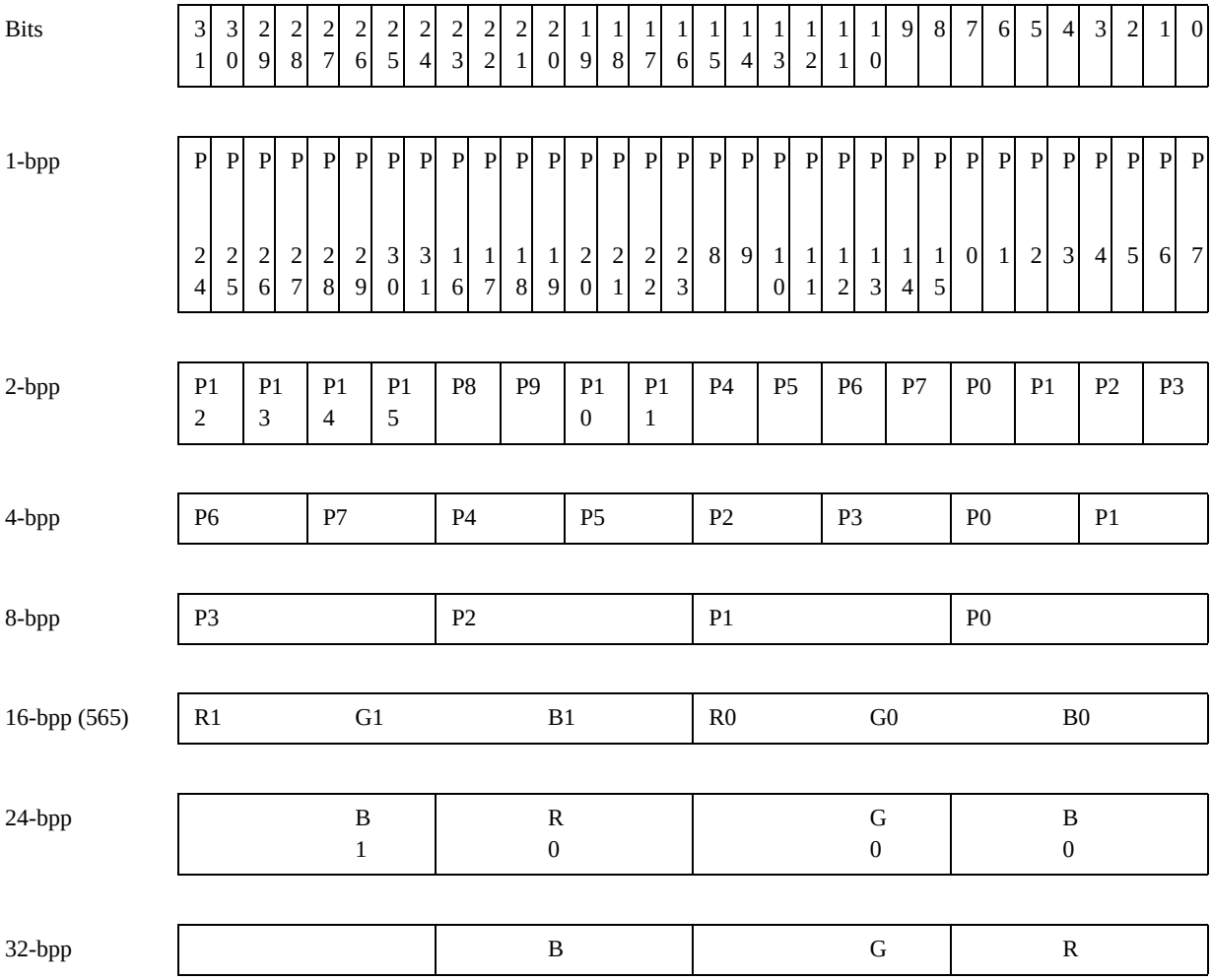
Theory of Operation

The two graphics controllers in the MQ-200 (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-200 supports pixel depths of 1, 2, 4, 8, 16, 24 and 32 bits per pixel (bpp). These pixel depths correspond to 2, 4, 16, 256, 65536 and 16.7 million color 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 16, 24 and 32 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:



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-200 logic. FIFOs in each graphics controller decouple the display control unit from the internal display memory.

Pixel Generation

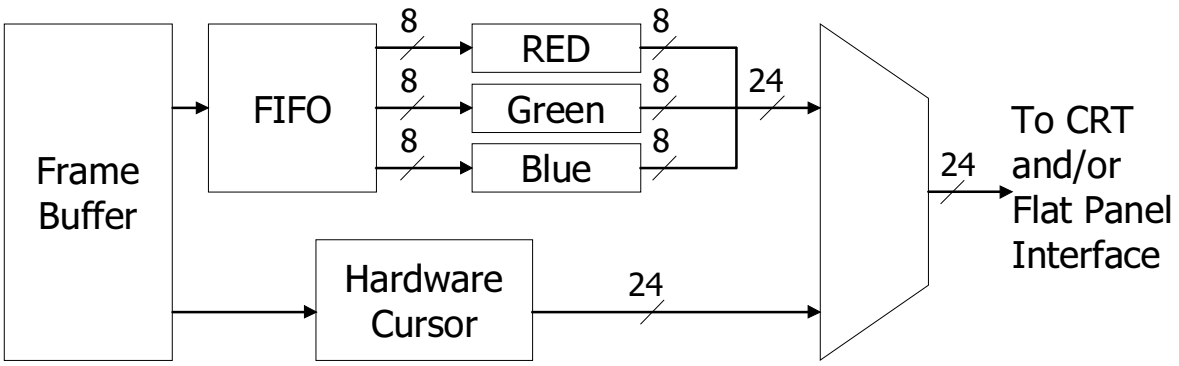


Figure 3-15: Pixel Generation

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.

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 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.

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.

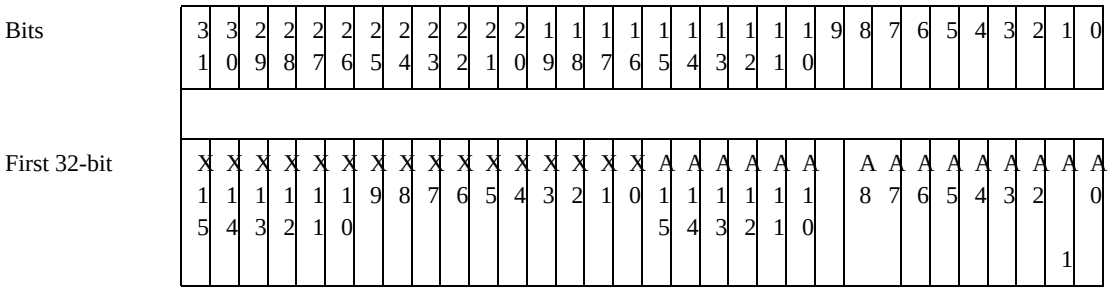
Hardware Cursor Image Format

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:

Table 3-16: AND/XOR Bit Functions

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:



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.

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-200 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-200 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.

Flat Panel Interface External Signals

The MQ-200 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:

Table 3-17: Signals for TFT Panels

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-200 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:

Table 3-18: Signals for STN Panels

FLM	First Line Mark
LP	Line Pulse
FMOD	Modulation Clock signal
FSCLK	**Shift Clock:** data and timing signals are output by the MQ-200 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:

Table 3-19: Signals for all Panels

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.

Table 3-19: Signals for all Panels

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

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.

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 3-20: Interface 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	Color S-STN 24-bit
FD23	R7	R5	R3	UR0	UR0	UR0	UR0	UR0
FD22	R6	R4	R2	UR1	UR1	UR1	UR1	UR1
FD21	R5	R3	R1	UR2	UR2	UR2		UR2
FD20	R4	R2	R0	UR3	UR3	LR0	LR0	UR3
FD19	R3	R1		UR4	LR0	LR1	LR1	UR4
FD18	R2	R0		UR5	LR1	LR2		UR5

Table 3-20: Interface 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	Color S-STN 24-bit
FD17	R1				LR2			UR6
FD16	R0				LR3			UR7
FD15	G7	G5	G3	UG0	UG0	UG0	UG0	UG0
FD14	G6	G4	G2	UG1	UG1	UG1		UG1
FD13	G5	G3	G1	UG2	UG2	UG2		UG2
FD12	G4	G2	G0	UG3	UG3	LG0	LG0	UG3
FD11	G3	G1		UG4	LG0	LG1		UG4
FD10	G2	G0			LG1	LG2		UG5
FD9	G1				LG2			UG6
FD8	G0				LG3			UG7
FD7	B7	B5	B3	UB0	UB0	UB0	UB0	UB0
FD6	B6	B4	B2	UB1	UB1	UB1		UB1
FD5	B5	B3	B1	UB2	UB2			UB2
FD4	B4	B2	B0	UB3	UB3	LB0	LB0	UB3
FD3	B3	B1		UB4	LB0	LB1		UB4
FD2	B2	B0		FSCLK= FPCLK*3 /16	LB1	FSCLK= FPCLK*3 /8	FSCLK= FPCLK*3 /4	UB5
FD1	B1				LB2			UB6
FD0	B0				LB3			UB7
FVSYNC	FVSYNC	FVSYNC	FVSYNC	FLM	FLM	FLM	FLM	FLM
FHSYNC	FHSYNC	FHSYNC	FHSYNC	LP	LP	LP	LP	LP
FDE	FDE	FDE	FDE	FMOD	FMOD	FMOD	FMOD	FMOD

Table 3-20: Interface 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	Color S-STN 24-bit
FDI	FDI	FDI	-	-	-	-	-	-
FSCLK	FSCLK= FPCLK	FSCLK= FPCLK	FSCLK= FPCLK	FSCLK or FPCLK	FSCLK= FPCLK/4	FSCLK or FPCLK	FSCLK or FPCLK	FSCLK= FPCLK

Note that interface for 12-bit/8-bit/4-bit color S-STN panel is the same as for 24-bit/16-bit/8-bit color D-STN panel without the lower panel data. With some CPU, only 18 out of 24 flat panel data bits are available. In this case, only FD[23:18], FD[15:10], and FD[7:2] are used to output data to flat panel, therefore panels that require more than 18 bits of data cannot be supported.

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 3-21: interface mapping for mono panels.

Pin Name	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				UD6	UD6	
FD22				UD7	UD7	
FD21						
FD20				UD14	LD6	
FD19				UD15	LD7	
FD18						
FD17						
FD16						
FD15	P1			UD0	UD0	UD0
FD14	P0			UD1	UD1	UD1
FD13				UD2	UD2	UD2
FD12				UD3	UD3	UD3
FD11				UD4	UD4	
FD10				UD5	UD5	
FD9						
FD8						

Table 3-21: interface mapping for mono panels.

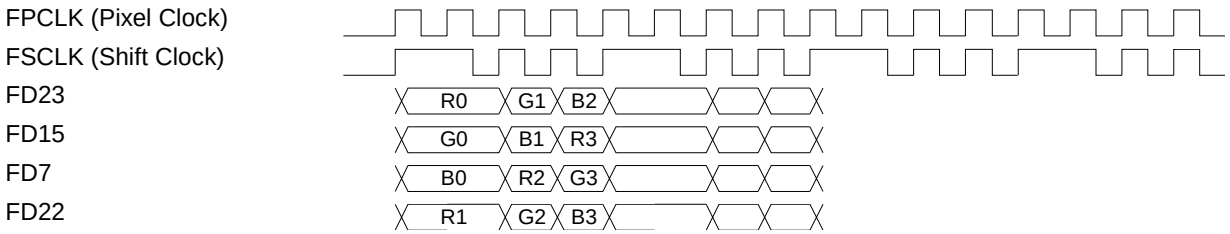
Pin Name	Mono TFT 8-bit	Mono TFT 6-bit	Mono TFT 4-bit	Mono S-STN 16-bit	Mono D-STN 16-bit	Mono D-STN 8-bit
FD7	P7	P5	P3	UD8	LD0	LD0
FD6	P6	P4	P2	UD9	LD1	LD1
FD5	P5	P3	P1	UD10	LD2	LD2
FD4	P4	P2	P0	UD11	LD3	LD3
FD3	P3	P1		UD12	LD4	
FD2	P2	P0		UD13	LD5	
FD1						
FD0						
FVSYNC	FVSYNC	FVSYNC	FVSYNC	FLM	FLM	FLM
FHSYNC	FHSYNC	FHSYNC	FHSYNC	LP	LP	LP
FDE	FDE	FDE	FDE	FMOD	FMOD	FMOD
FDI	-	-	-	-	-	-
FSCLK	FSCLK=FPCLK	FSCLK=FPCLK	FSCLK=FPCLK	FSCLK=FPCLK/16	FSCLK=FPCLK/8	FSCLK=FPCLK/4

Note that interface for 8-bit/4-bit mono S-STN panel is the same as for 16-bit/8-bit mono D-STN panel without the lower panel data. With some CPU, only 18 out of 24 flat panel data bits are available. In this case, only FD[23:18], FD[15:10], and FD[7:2] are used to output data to flat panel, therefore panels that require more than 18 bits of data cannot be supported.

Timing Diagrams for S-STN Panels

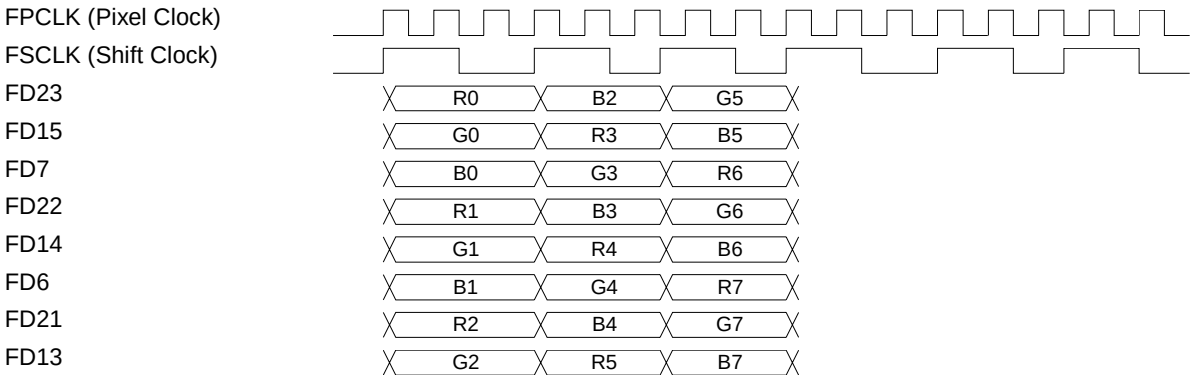
4-Bit Color S-STN

4-bit Color S-STN



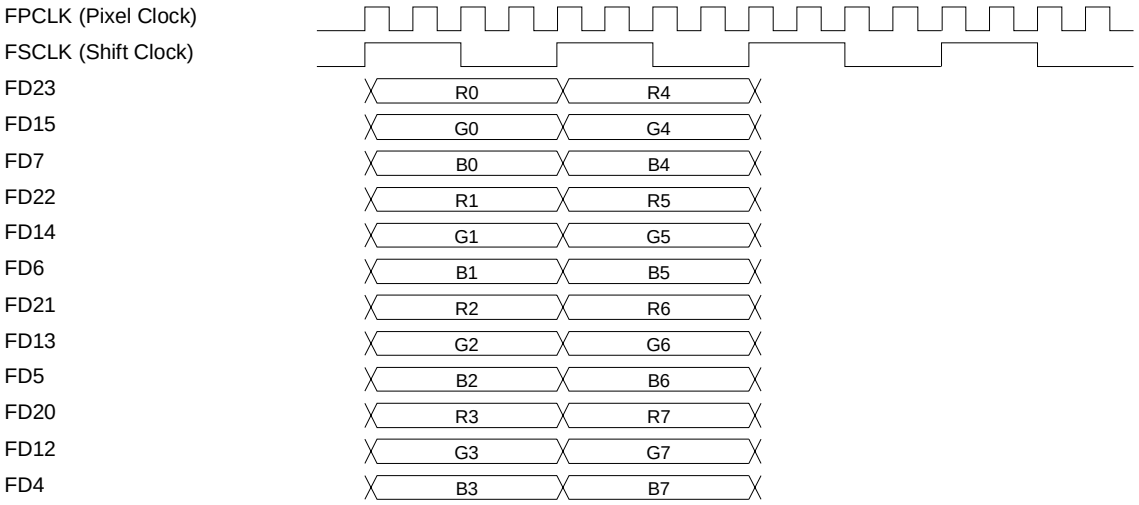
8-Bit Color S-STN

8-bit Color S-STN



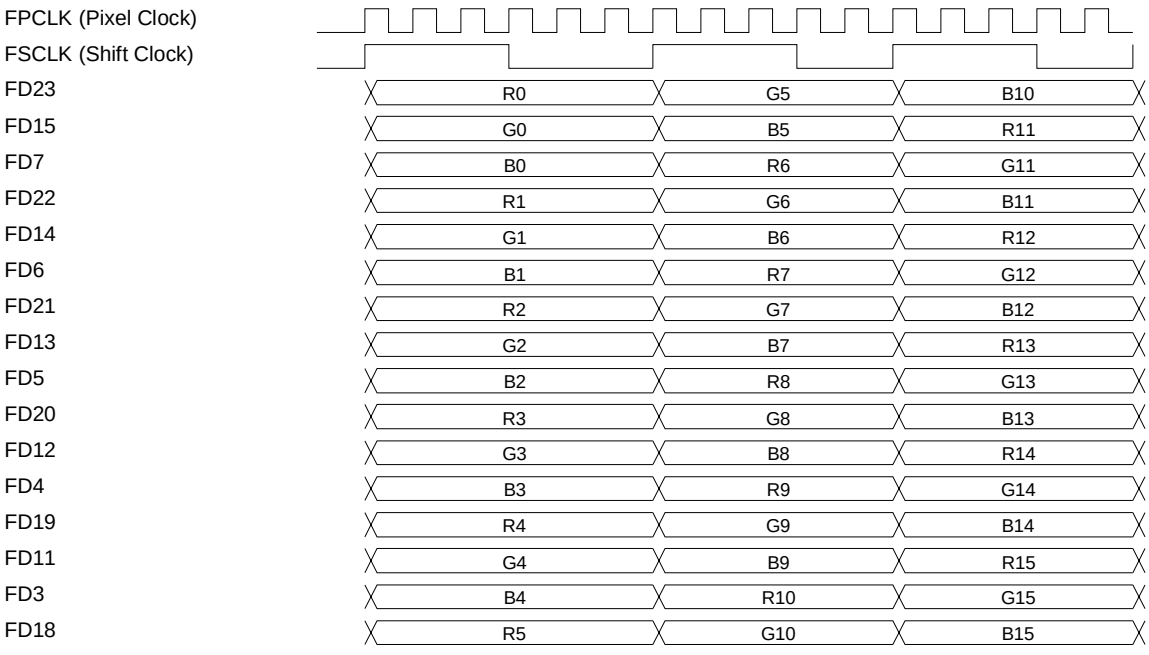
12-Bit Color S-STN

12-bit Color S-STN



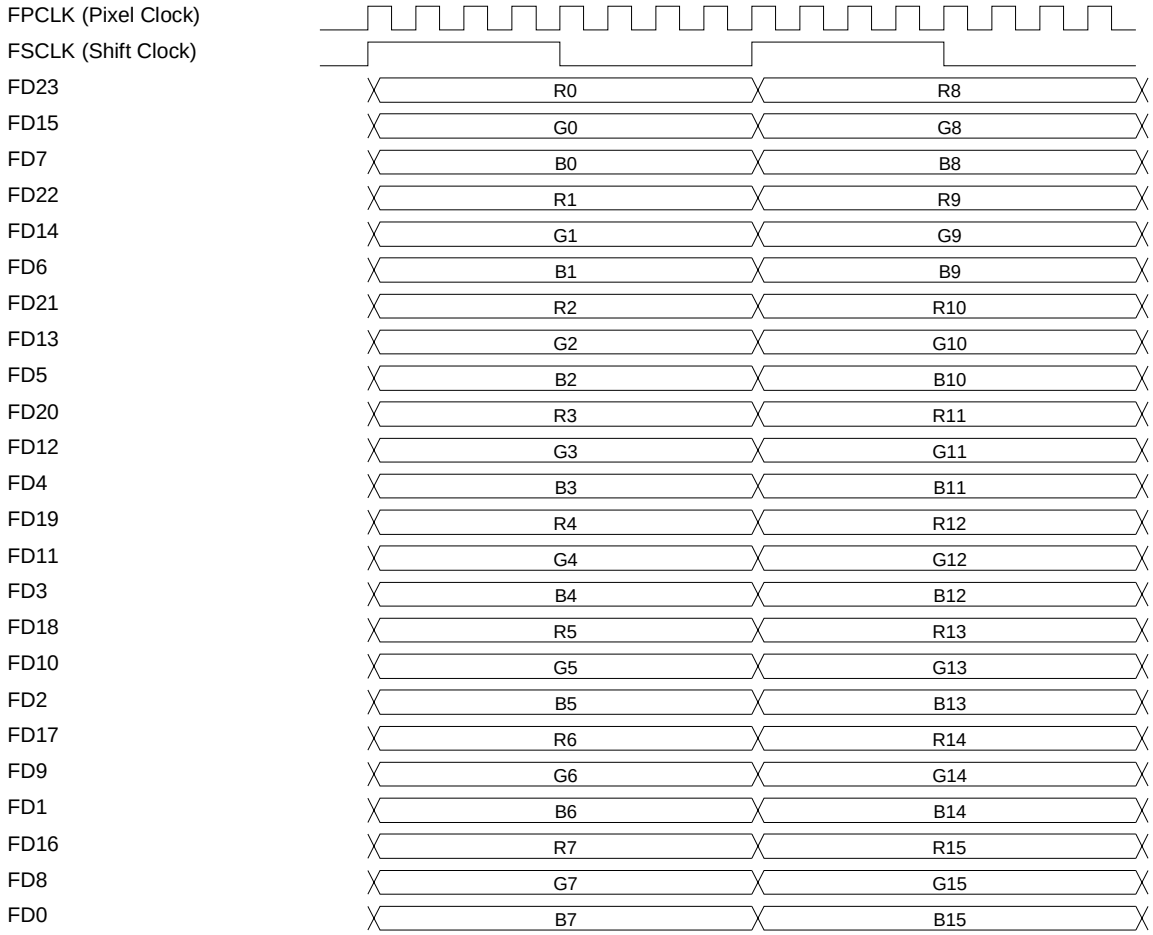
16-Bit Color S-STN

16-bit Color S-STN



24-Bit Color S-STN

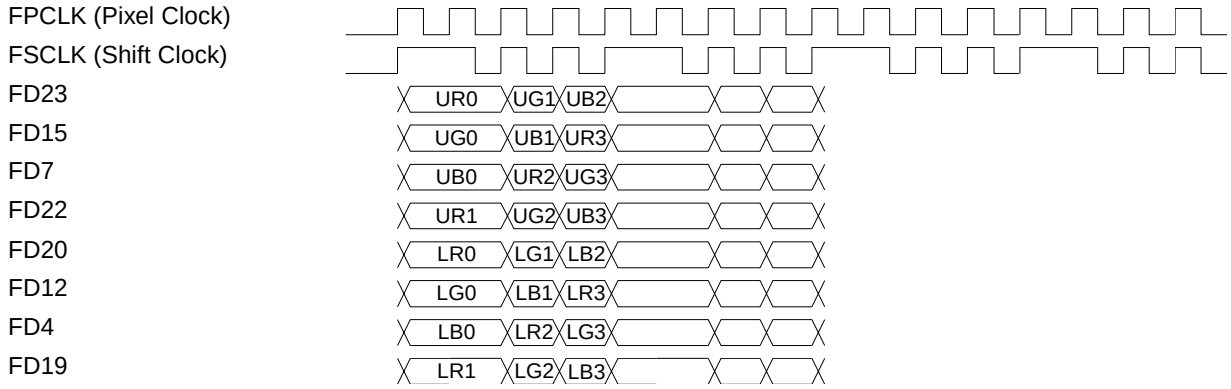
24-bit Color S-STN



Timing Diagrams for Color D-STN panels

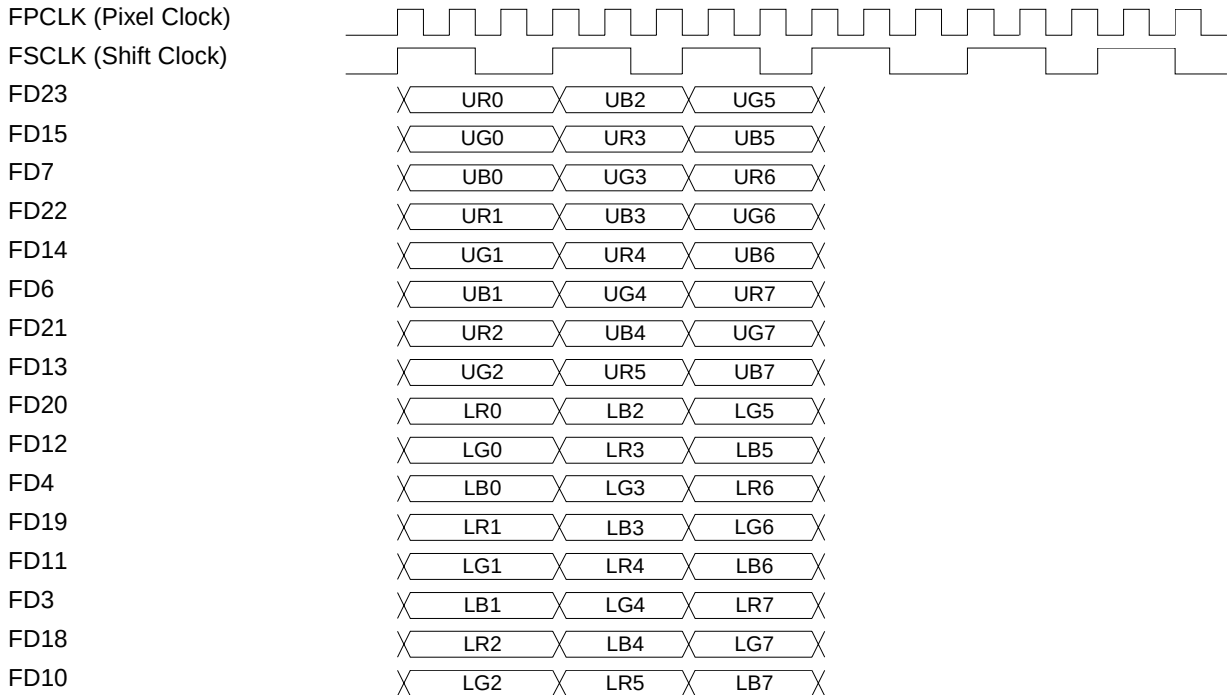
8-Bit Color D-STN

8-bit Color D-STN



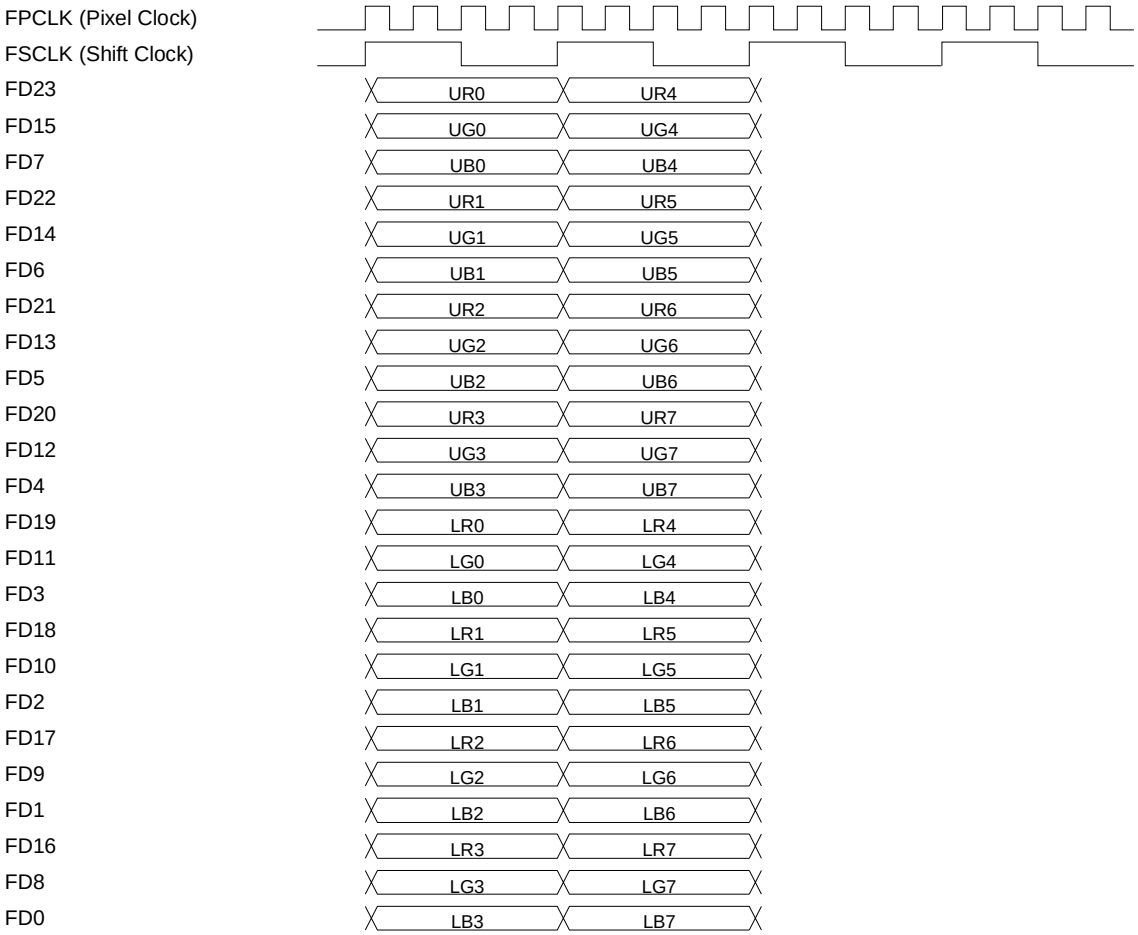
16-Bit Color D-STN

16-bit Color D-STN



24-Bit Color D-STN

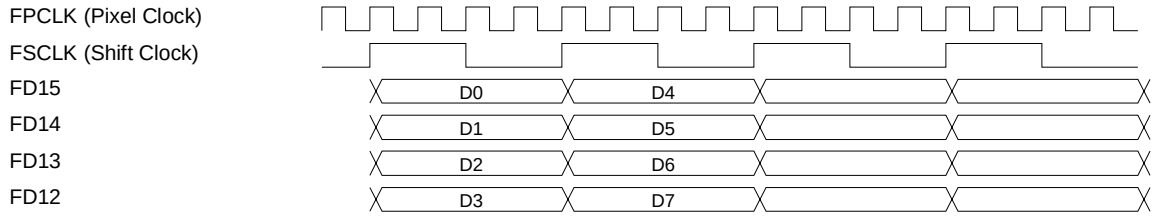
24-bit Color D-STN



Timing Diagrams for Mono S-STN Panels

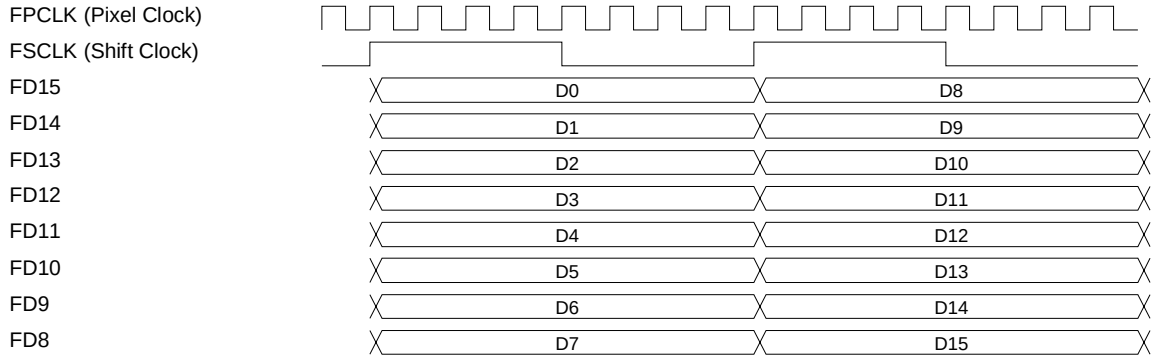
4-Bit Mono S-STN

4-bit Mono S-STN



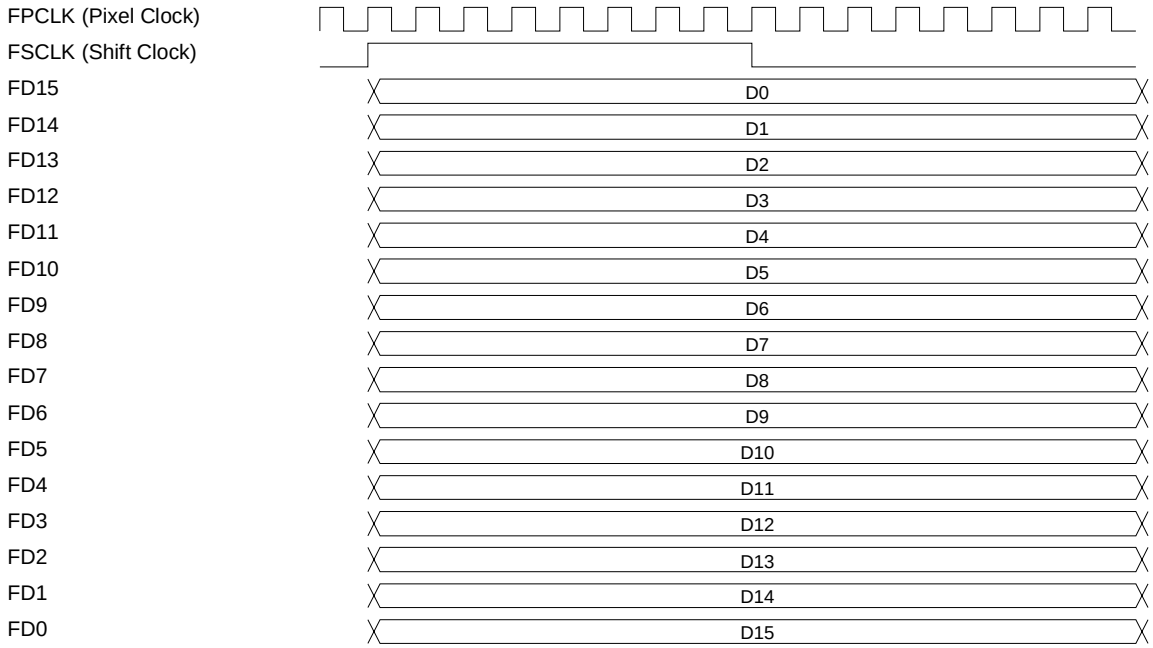
8-Bit Mono S-STN

8-bit Mono S-STN



16-Bit Mono S-STN

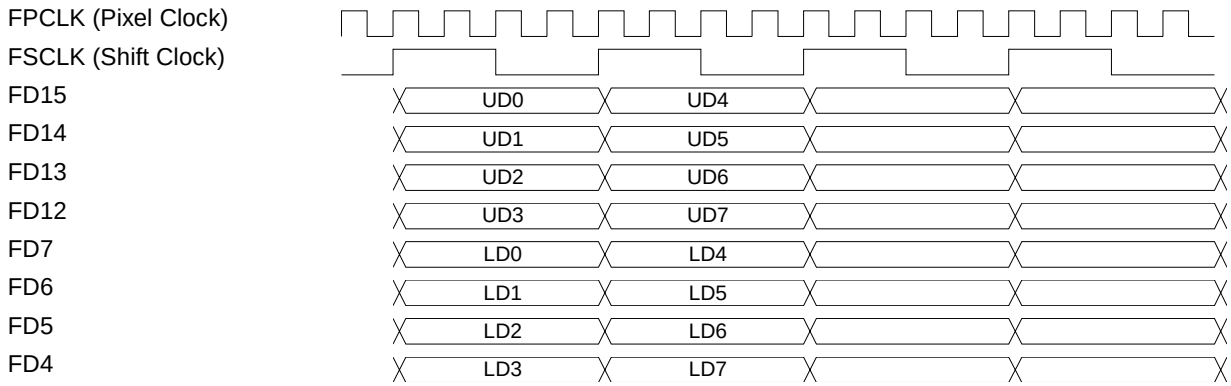
16-bit Mono S-STN



Timing Diagrams for Mono D-STN Panels

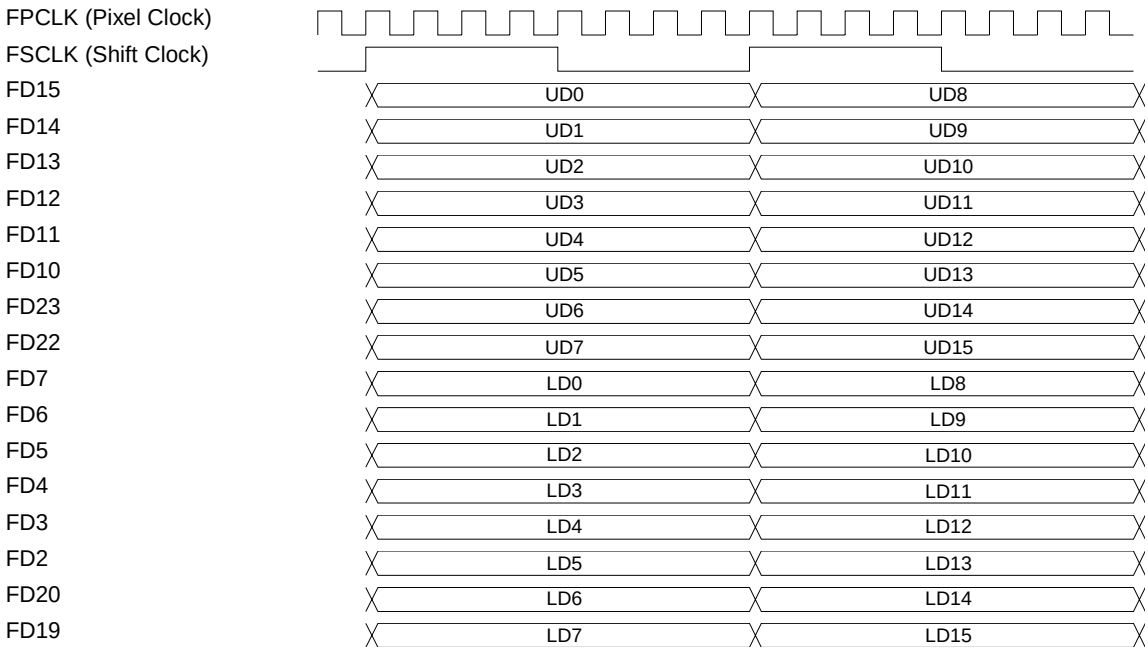
8-Bit Mono D-STN

8-bit Mono D-STN



16-Bit Mono D-STN

16-bit Mono D-STN



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.

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.

General Purpose I/O Port

Three General Purpose I/O pins are provided on the MQ-200. Each pin may be programmed as input or output.

When configured as inputs, input data from these pins is not synchronized to any MQ-200 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.

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.

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.

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-200 by the CRT monitor.

Digital-to-Analog Conversion

The MQ-200 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.

Blanking and SYNC Pedastal

The MQ-200 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.

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.

Graphics Controller Register Set

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

Chapter 4

Power Management

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-200 logic modules, are in an inactive state. The first operation in the MQ-200 initialization procedure must be to program the PMU to turn on the other functional modules. The MQ-200 device drivers provide all requisite initialization.

Individual MQ-200 logic modules employ DynamiQ^{TM} dynamic power management to minimize the MQ-200 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-200 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:

D0 State

Normal operating mode. In this state, all the functional modules on the MQ-200 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.

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.

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.

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-200 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.

D4 State

This is the lowest power mode on the MQ-200 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-200 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).

Software-Initiated Mode Transitions

Upon a Power-On Reset (POR# asserted and PDWN# is forced high), the MQ-200 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-200 is operating. Power State D4 can be entered only by forcing the PDWN# pin low (active).

Hardware Mode Transitions

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

The MQ-200 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-200 will be reset and will enter D3. When POR# is deasserted, if PDWN# is still asserted the MQ-200 will enter D4 state with memory refresh disabled.

Role of the Operating System in Managing Power

Although the current version of Windows CE does not explicitly support all of the MQ-200 power management modes, the OEM can take advantage of the MQ-200 power management through the system software drivers.

Role of the Device Driver in Managing Power

Each WindowsCE device driver implements the power management API. The device drivers for the MQ-200 will take high-level power management commands from the OS and implement them by programming the device appropriately.

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-200'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.

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.

Power Sequencing

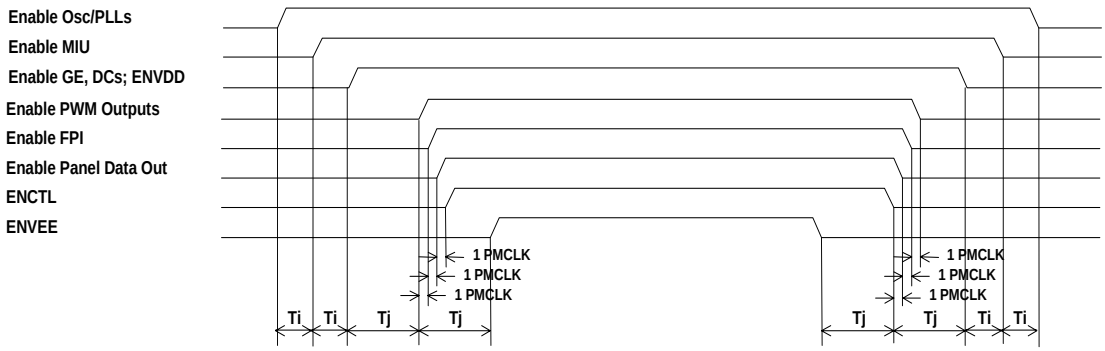


Figure 4-1: Power Sequencing

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 (It 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.

Power Management Register Set

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

Chapter 5

Programming Information

MQ-200 Address Map

The MQ-200 connects directly to Hitachi SH-7750, Intel SA-1110, NEC VR-4121 and Toshiba Tx-3922 microprocessors. These CPUs support a Logical Address Space and a Physical Address Space. The physical address space is decoded internally by the CPU, which generates the chip select signals and the address bus. Each chip select region consists of 64 MBytes of address space for the SH-7750, SA-1110 and Tx-3922 processors.

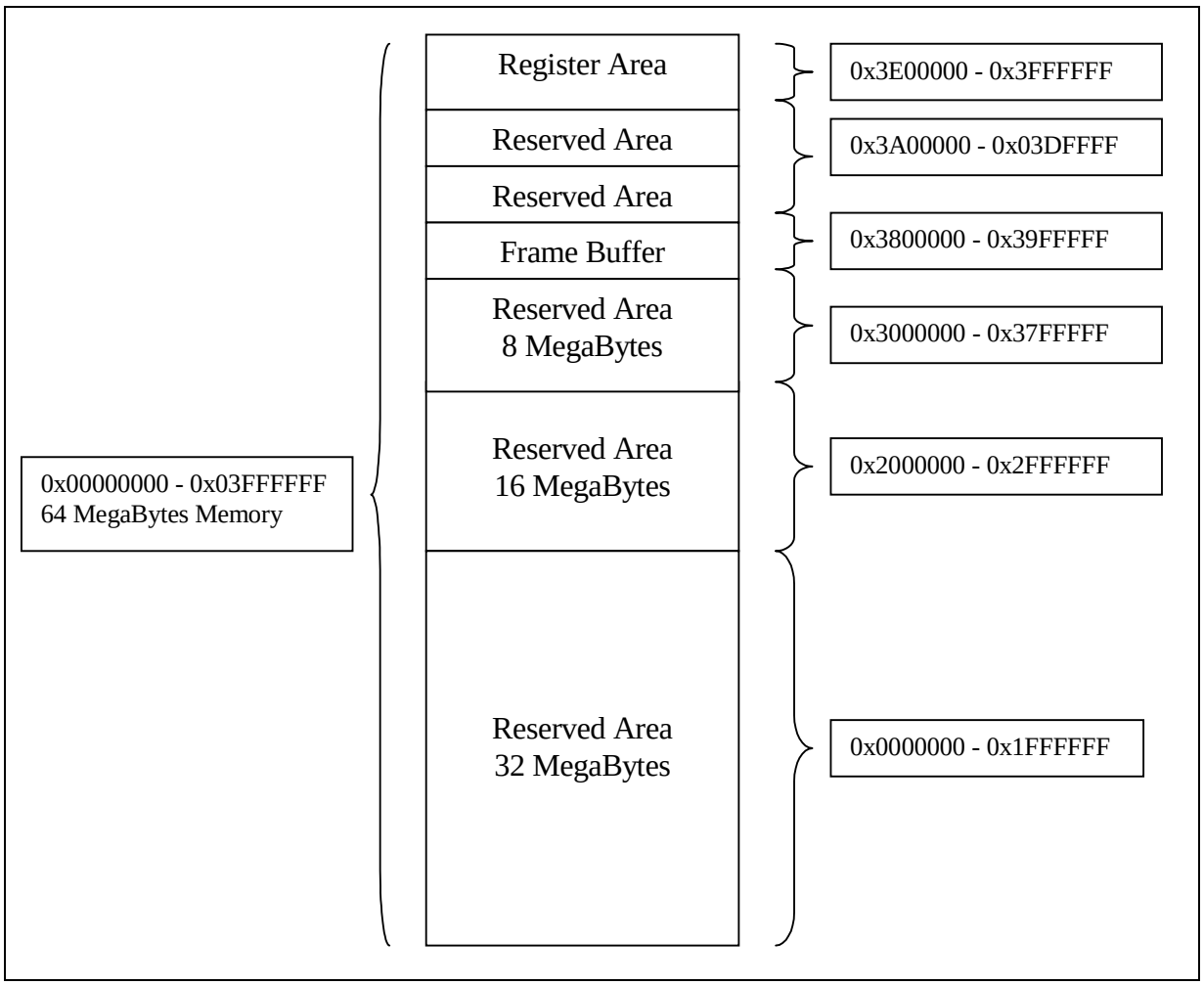


Figure 5-1: MQ-200 Memory Map for SH-7750, SA-1110, Tx-3922

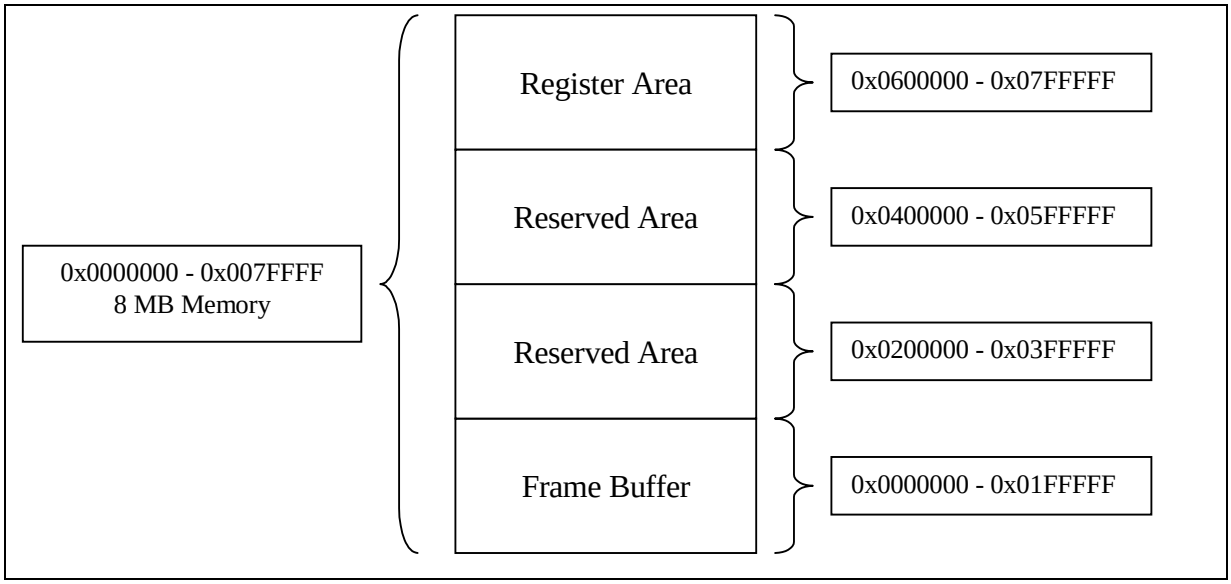


Figure 5-2: MQ-200 Memory Map for NEC VR-4121

When the CPU selects the MQ-200 by asserting MQ-200 CS# pin, address bits A[25:2] are decoded to select a 32-bit MQ-200 register or a 32-bit internal memory location. The register and memory addresses specified in this document are offsets within the 64-MByte region decoded by the MQ-200; 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-200 is assigned.

The MQ-200 CS# input is connected to CS1# or CS4# output of the SH-7750 processor. The chip select region which is mapped to MQ-200 must be programmed to be a 32-bit region in the SH-7750. In SH-7750 systems, the upper half of the CS4# region is assigned to MQ-200. Designers of SH-7750 systems can select either CS4# or CS1# to be connected to the CS# input of MQ-200.

The MQ-200 CS# input is connected to the nCS output of the SA-1110.

In the case of the VR-4121 processor interface, address pin A[24] determines the location of the MQ-200 in the upper 8 MB or lower 8 MB address location. A[24]=0 indicates MQ-200 occupies the lower 8 MB and A[24]=1 indicates MQ-200 occupies the upper 8 MB. The CS# input on the MQ-200 is connected to the LCDCS# output pin on the VR-4121 processor.

The CS# signal on the MQ-200 is connected to the appropriate MCS0/MCS1 output pin the Tx-3922 processor.

MQ-200 registers are aligned on 32-bit boundaries. Registers are in the various MQ-200 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 5-3: Address Mapping of Modules

FUNCTIONAL MOUDLE	SH-7750,SA-1110,TX-3922 ADDRESS	VR-4121 ADDRESS (A[24]=0)***
Frame buffer	0x03800000-0x039FFFFC	0x000000-0x1FFFFC
Power management	0x03E00000-0x03E01FFC	0x600000-0x601FFC
CPU interface	0x03E02000-0x03E03FFC	0x602000-0x603FFC
Memory controller	0x03E04000-0x03E05FFC	0x604000-0x605FFC
Interrupt controller	0x03E08000-0x03E09FFC	0x608000-0x609FFC
Graphics controller 1 & 2	0x03E0A000-0x03E0BFFC	0x60A000-0x60BFFC
Graphics engine	0x03E0C000-0x03E0DFFC	0x60C000-0x60DFFC
Flat panel controller	0x03E0E000-0x03E0FFFC	0x60E000-0x60FFFC
Color palette 1	0x03E10000-0x03E11FFC	0x610000-0x611FFC
Device configuration	0x03E14000-0x03E15FFC	0x614000-0x615FFC
PCI configuration	0x03E16000-0x03E17FFC	0x616000-0x617FFC

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.

Register Locator Table

shows the list of registers and their

Table 5-4: Register Locator Table

Register	Page
Graphics Controller 1 Register Definition	5-16
Graphics Controller CRT Control	5-18
GC02R Horizontal Display 1 Control	5-22
GC03R Vertical Display 1 Control	5-23
GC04R - Horizontal Sync 1 Control	5-24
GC05R - Vertical Sync 1 Control	5-25
GC06R - Reserved	5-26
GC07R - Vertical Display 1 Count (Read Only)	5-27
GC08R - Horizontal Window 1 Control	5-28
GC09R - Vertical Window 1 Control	5-29
GC0AR - Alternate Horizontal Window 1 Control	5-30
GC0BR - Alternate Vertical Window 1 Control	5-31
GC0CR - Window 1 Start Address	5-32
GC0DR - Alternate Window 1 Start Address	5-33
GC0ER - Window 1 Stride	5-34
GC0FR - Reserved	5-35
GC10R - Hardware Cursor 1 Position	5-36
GC11R - Hardware Cursor 1 Start Address and Offset	5-37
GC12R - Hardware Cursor 1 Foreground Color	5-38
GC13R - Hardware Cursor 1 Background Color	5-39
GC14R to GC1FR - Reserved	5-40
GC20R to GC3FR - Graphics Controller 2 Registers	5-41
GC21R - Graphics Controller CRC Control	5-42
GC40R to GCFFR – Reserved	5-43
C100R to C1FFR - Graphics Controller Color Palette	5-44
GC20R: Graphics Controller 2 Control Register	5-45
GC21R: Graphics Controller CRC Result Register	5-48
GC22R: Horizontal Display Timing Control Register	5-49
GC23R: Vertical Display Timing Control Register	5-50
GC24R: Horizontal SYNC Control Register	5-51
GC25R: Vertical SYNC Control Register	5-52

Table 5-4: Register Locator Table

Register	Page
GC28R: Graphics Controller 2 Main Window Horizontal Control Register	5-53
GC29R: Graphics Controller 2 Main Window Vertical Control Register	5-54
GC2AR: Graphics Controller 2 Alternate Window Horizontal Control Register	5-55
GC2BR: Graphics Controller 2 Alternate Window Vertical Control Register	5-56
GC2CR: Main Window 2 Start Address	5-57
GC2DR: Alternate Window 2 Start Address	5-58
GC2ER: Window 2 Stride Register	5-59
GC2FR: Window 2 Line Size Register	5-60
GC30R: Hardware Cursor 2 Position	5-61
GC31R: Hardware Cursor 2 Start Address and Offset Register	5-62
GC32R: Hardware Cursor 2 Foreground Color Register	5-63
GC33R: Hardware Cursor 2 Background Color Register	5-64
FP00R: Flat Panel Control Register	5-66
FP01R: Flat Panel Output Pin Control Register	5-69
FP02R: Flat Panel General Purpose Output Control Register	5-73
FP03R: General Purpose I/O Port Control Register	5-75
FP04R: STN Panel Control Register	5-77
FP05R: D-STN Half-Frame Buffer Control Register	5-78
FP0FR: Pulse Width Modulation Control Register	5-79
FP10R to FP2FR: Frame-Rate Control Pattern Registers	
FP30R to FP37R: Frame-Rate Control Weight Registers	5-82
FIFO Pixel Data Relationship	5-86
GE00R: Primary Drawing Command Register	5-94
GE01R: Primary Width and Height Register	5-97
GE02R: Primary Destination Address Register	5-99
GE03R: Primary Source XY Register	5-100
GE04R: Primary Color Compare Register	5-101
GE05R: Primary Clip Left/Top Register	5-102
GE06R: Primary Clip Right/Bottom Register	5-103
GE07R: Primary Source and Pattern Offset Register	5-104
GE08: Primary Foreground Color Register / Rectangle Fill Color	5-105
GE09R - Source Stride/Offset Register	5-105
GE0AR – Destination Stride Register and Color Depth	5-107

Table 5-4: Register Locator Table

Register	Page
GE0BR - Image Base Address Register	5-108
GE1FR - Test Mode Read Register	5-109
GE20R to GE3FR	5-109
GE40R to GE5FR - Primary Color Pattern Register	5-110
GE40R - Mono Pattern Register 0	5-111
GE41R - Mono Pattern Register 1	5-112
GE42R - Foreground Color Register	5-112
GE43R - Background Color Register	5-112
Reserved	5-112
GEA0R to GEBFR	5-112
GEC0R to GEDFR - Secondary Color Pattern Register	5-113
CPU Interface (CIF)	5-114
Source FIFO/Command FIFO/GE Status Register	5-114
PMR – Power Management Register	5-115
PMCSR – Power Management Control/Status Register	5-116
Memory Interface Control 1	5-117
MM01R – MIU Interface Control 2	5-117
MM02R-Memory Interface Control 3	5-119
MM03R-Memory Interface Control 4	5-121
MM04R-Memory Interface Control 5	5-121
DC00R: Device Configuration Register 0	5-123
DC01R: Device Configuration Register 1 (Read Only)	5-125
DC02R: Software Register 0	5-126
DC03R: Software Register 1	5-126
Power Management Unit Configuration Register	5-127
Power Management Unit PM01R – D1 State Control	5-130
Power Management Unit – D2 State Control	5-133
Power Management Unit – PLL 2 Programming	5-136
Power Management Unit – PLL 3 Programming	5-137

Device Initialization

Recommended Initialization Procedure

Upon Power On Reset (with PDWN# forced high), the Power Management State is set to D3. The MQ-200 will be disabled until the power-up sequence is completed. The MQ-200 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-200 in the order dictated by the PMU. Initialization software must wait for the power-up sequencing to complete before further programming of the MQ-200. Completion is indicated for example, when the PMCSR register reads 0, confirming that state D0 has been entered and the MQ-200 is ready to be programmed.

The MQ-200 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.

Memory Configuration

No user configuration is required for the embedded memory in the MQ-200. 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.

Graphics Controller Programming Information

Each MQ-200 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:

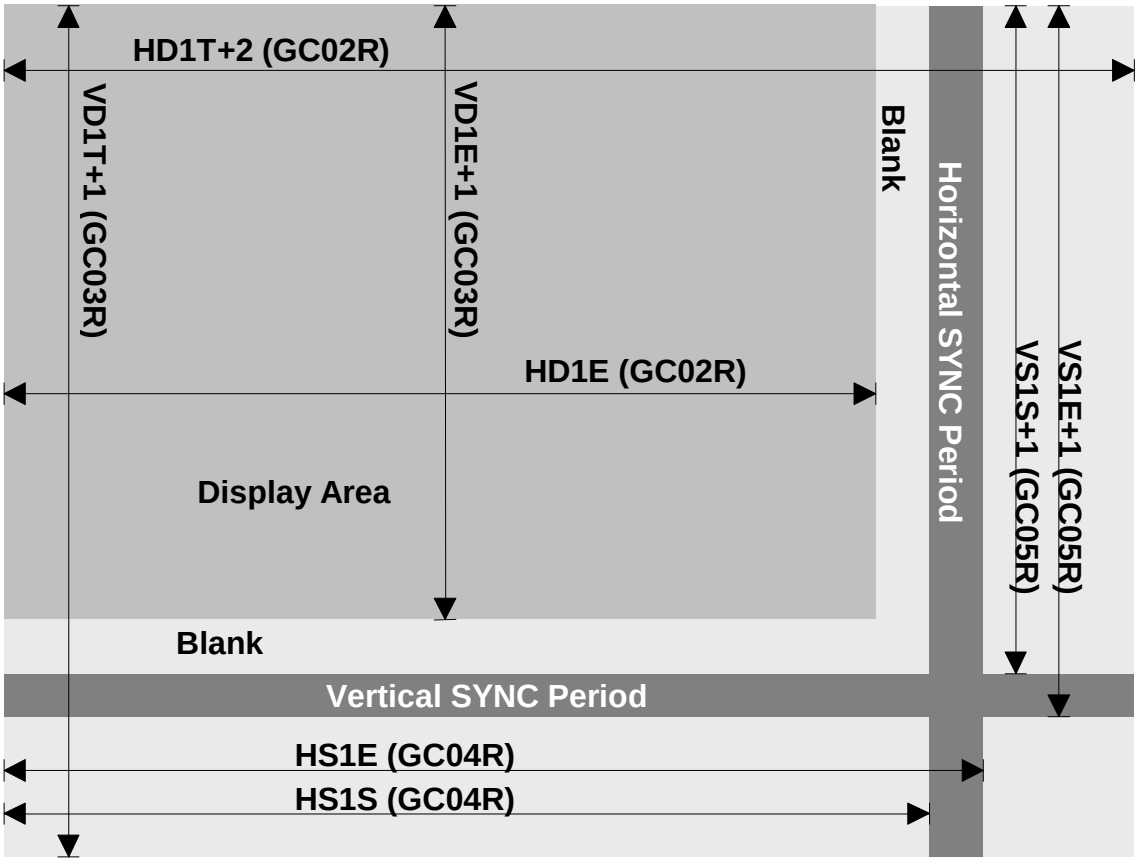


Figure 5-5: Horizontal and Vertical Parameters

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:

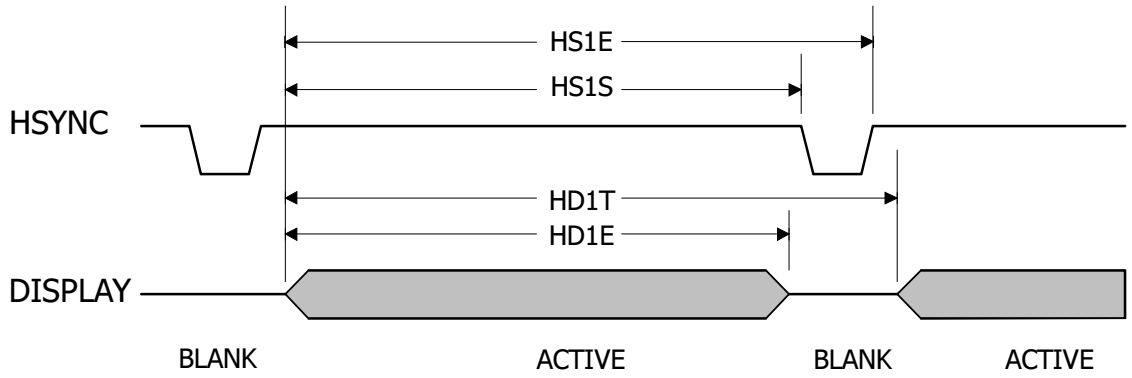


Figure 5-6: HSYNC Parameters

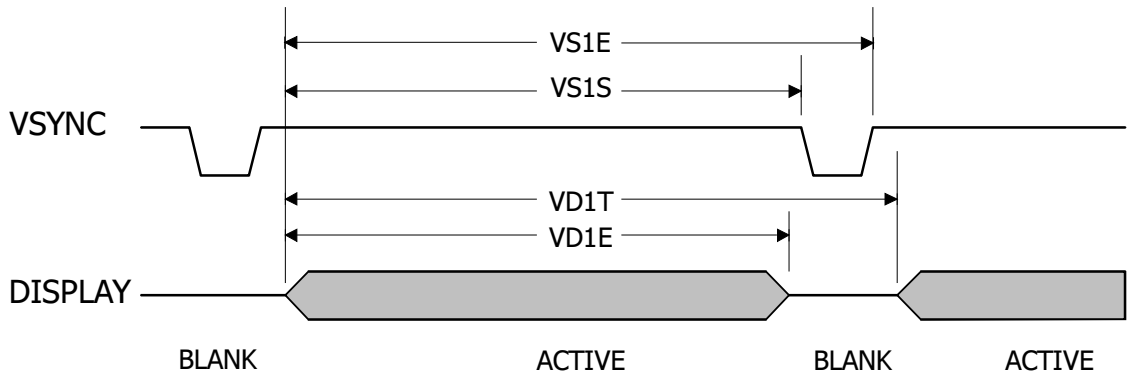


Figure 5-7: VSYNC Parameters

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.

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.

Flat Panel Interface

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

Single-Display Configuration

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

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.

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.

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.

Graphics Controller (GC)

The Graphics Controller module supports the following features:

- Display resolutions up to:
  - 800x600 24/32-bpp for single/dual-scan display (CRT, TFT, S-STN, or D-STN).
  - 1024x768 24-bpp for single/dual-scan display (CRT, TFT, S-STN, or D-STN).
  - 1280x1024 8-bpp for CRT display.
- 2 MBytes of display memory with 128-bit interface.
- One graphics image window can be displayed on each monitor (CRT/FP). Size of the graphics window can be smaller than the display resolution.
- Color depth of 1-, 2-, 4-, 8-, 16-, 24-, and 32-bpp graphics modes. In 16-bpp modes, RGB565 format is supported. In 32-bpp modes, both ARGB888 (PAL-BGR) and ABGR888 (PAL-RGB) formats are supported.
- Triple 8-bit color palettes for 1-, 2-, 4-, and 8-bpp modes. These color palettes can be used for gamma correction for the graphics image window in 16-, 24-, and 32-bpp modes. Color palette is shared for the two graphics controllers.
- Hardware cursor with resolution up to 64x64 2-bpp. One cursor can be displayed on each display (CRT/FP).
- Simultaneous display mode of CRT and FP (flat panel) of same image, resolution, color depth, and refresh rate. The supported modes depend on frame buffer memory size (and possibly bandwidth).
- Dual display mode of CRT and FP with different image, resolution, color depth, refresh rate. The supported modes depend on frame buffer memory size (and possibly bandwidth).
- Maximum pixel clock rate is 135 MHz for CRT (for 1280x1024 75 Hz) and 110 MHz for TFT panels (for 1280x1024 60 Hz) and 65 MHz for STN panels.

Graphics Controller Register Definition

Table 5-8: Graphics Controller 1 Register Definition

GC1	**GC00R - Graphics Controller 1 Control** **Index: 00h 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.
7-4	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).	
	0000	1-bpp graphics. Color palette is enabled.
	0001	2-bpp graphics. Color palette is enabled.
	0010	4-bpp graphics. Color palette is enabled.
	0011	8-bpp graphics. Color palette is enabled.
	0100	16-bpp (RGB565) graphics with color palette enabled.
	0101	24-bpp (RGB888) graphics with color palette enabled.
	0110	32-bpp (ARGB888 or PAL-BGR) graphics with color palette enabled.
	0111	32-bpp (ABGR888 or PAL-RGB) graphics with color palette enabled.
	1000	Reserved.
	1001	Reserved.
	1010	Reserved.
	1011	Reserved.
	1100	16-bpp (RGB565) graphics with color palette bypassed.
	1101	24-bpp (RGB888) graphics with color palette bypassed.

Table 5-8: Graphics Controller 1 Register Definition

	1110	32-bpp (ARGB888 or PAL-BGR) graphics with color palette bypassed.
	1111	32-bpp (ABGR888 or PAL-RGB) graphics with color palette bypassed.
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/ Width, Vertical Window 1 Start/End/Height, Window 1 Start Address, and Window 1 Stride parameters are used.
	1	Alternate Window 1 is enabled. In this mode, Alternate Graphics Color Depth, Alternate Horizontal Window 1 Start/End/Width, Alternate Vertical Window 1 Start/End/Height, Alternate Window 1 Start Address, Alternate Window 1 Stride, 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.
15-12	*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).	
	0000	1-bpp graphics. Color palette is enabled.
	0001	2-bpp graphics. Color palette is enabled.
	0010	4-bpp graphics. Color palette is enabled.
	0011	8-bpp graphics. Color palette is enabled.
	0100	16-bpp (RGB565) graphics with color palette enabled.
	0101	24-bpp (RGB888) graphics with color palette enabled.
	0110	32-bpp (ARGB888 or PAL-BGR) graphics with color palette enabled.
	0111	32-bpp (ABGR888 or PAL-RGB) graphics with color palette enabled.
	1000	Reserved.
	1001	Reserved.
	1010	Reserved.
	1011	Reserved.
	1100	16-bpp (RGB565) graphics with color palette bypassed.
	1101	24-bpp (RGB888) graphics with color palette bypassed.
	1110	32-bpp (ARGB888 or PAL-BGR) graphics with color palette bypassed.
	1111	32-bpp (ABGR888 or PAL-RGB) graphics with color palette bypassed.

Table 5-8: Graphics Controller 1 Register Definition

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 1 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.	

Table 5-9: Graphics Controller CRT Control

GC1	**GC01R - Graphics Controller CRT Control** **Index: 04h Reset value: 0xxx-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 is set to 00.	
	0	CRT HSYNC output is forced low during power down mode and bits 5-4 is set to 00.
	1	CRT HSYNC output is Power Management clock (PMCLK) during power down mode and bits 5-4 is 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 when bits 7-6 is set to 00.	
	0	CRT VSYNC output is forced low during power down mode and bits 7-6 is set to 00.
	1	CRT VSYNC output is Power Management clock (PMCLK) during power down mode and bits 7-6 is set to 00.
5-4	*CRT HSYNC Control* These bits control output on CRT HSYNC pin. Note that external level shiftier 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 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 is 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 is 00.	

Table 5-9: Graphics Controller CRT Control

	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	*Composite Sync Enable* When bits 7-6 is 00 and this bit is set to 1, VSYNC pin will output simple Composite Sync (CSYNC) instead.	
	0	VSYNC goes out on VSYNC output pin when bits 7-6 is 00.
	1	CSYNC goes out on VSYNC output pin when bits 7-6 is 00. CSYNC is (VSYNC xor HSYNC) when bit 8 is 0 and it is (VSYNC xnor HSYNC) when bit 8 is 1.
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, Blue DAC 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, Blue DAC when monitor sensing is enabled (bit 14 = 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.

Table 5-9: Graphics Controller CRT Control

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.
27	*Reserved*	
28	*Mono/Color Monitor Select* This bit selects between color (RGB) and mono (G only) monitor.	
	0	Color monitor. When DAC is enabled, Red, Green, and Blue DAC are all enabled.
	1	Mono monitor. When DAC is enabled, only Green DAC is enabled.
31-29	*Reserved (R/W)* These bits must be programmed to 0.	

Table 5-10: GC02R Horizontal Display 1 Control

GC1	**GC02R - Horizontal Display 1 Control** **Index: 08h 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*
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*

Table 5-11: GC03R Vertical Display 1 Control

GC1	**GC03R - Vertical Display 1 Control** **Index: 0Ch 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 + 2 where VS1E is Vertical Sync 1 End (see GC05R). Programmed value = actual value – 1
15-12	*Reserved*

Table 5-11: GC03R Vertical Display 1 Control

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*

Table 5-12: GC04R - Horizontal Sync 1 Control

GC1	**GC04R - Horizontal Sync 1 Control** **Index: 10h 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*
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*

Table 5-13: GC05R - Vertical Sync 1 Control

GC1	**GC05R - Vertical Sync 1 Control** **Index: 14h 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*
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 ≥ VS1S + 1. Programmed value = Vertical Sync 1 Start + Vertical Sync 1 Height.
31-28	*Reserved*

Table 5-14: GC06R - Reserved

GC1	**GC06R - Reserved** **Index: 18h Reset value:?h**
31-0	*Reserved*

Table 5-15: GC07R - Vertical Display 1 Count (Read Only)

GC1	**GC07R - Vertical Display 1 Count (Read Only)** **Index: 1Ch Reset value: 0000-0000h**

Table 5-15: GC07R - Vertical Display 1 Count (Read Only)

11-0	*Vertical Display 1 Count (Read Only)* These bits reflect the state of the internal vertical display counter. Note that the counter is asynchronous to the read operation therefore it is possible that the counter is being updated when this register is read. To make sure that the counter value is accurate, two consecutive reads must return the same value.
31-12	*Reserved*

Table 5-16: GC08R - Horizontal Window 1 Control

GC1	**GC08R - Horizontal Window 1 Control** **Index: 20h 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*
27-16	*Horizontal Window 1 Width (HW1W)* This parameter specifies the width of the graphics image window in terms of number of pixels when alternate window is disabled. Constraint: HW1W ≥ 8. Programmed value = actual value - 1. The Horizontal Window 1 End (HW1E) position with respect to the left edge of the display area is calculated from: HW1S + HW1W + 1.
31-28	*Window 1 Additional Line Data (W1ALD)* This parameter specifies the additional number of bytes of data to be fetched from frame buffer for each line when alternate window is disabled. The total number of bytes of data that will be fetched from frame buffer for each line is calculated from: ((HW1W * bpp / 8) + W1ALD + n) where: 'bpp' is the number of bits/pixel (color depth), and: 'n' is 0 for 1/2/4/8-bpp, 1 for 16/24-bpp, and 3 for 32-bpp mode. The calculated value will be rounded down to the nearest byte. Normally, this parameter can be set to 1 for all modes.

Table 5-17: GC09R - Vertical Window 1 Control

GC	**GC09R - Vertical Window 1 Control** **Index: 24h 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*
27-16	*Vertical Window 1 Height (VW1H)* This parameter specifies the height of the graphics image window in terms of number of lines when alternate window is disabled. Constraint: VW1H ≥ + 1. Programmed value = actual value -1. The Vertical Window 1 End (VW1E) position with respect to the top edge of the display area is calculated from: VW1S + VW1H + 1.
31-28	*Reserved*

Table 5-18: GC0AR - Alternate Horizontal Window 1 Control

GC1	**GC0AR - Alternate Horizontal Window 1 Control** **Index: 28h 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*
27-16	*Alternate Horizontal Window 1 Width (AHW1W)* This parameter specifies the width of the graphics image window in terms of number of pixels when alternate window is enabled. Constraint: AHW1W ≥ 8. Programmed value = actual value - 1. The Alternate Horizontal Window 1 End (AHW1E) position with respect to the left edge of the display area is calculated from: AHW1S + AHW1W + 1.
31-28	A*lternate Window 1 Additional Line Data (AW1ALD)* This parameter specifies the additional number of bytes of data to be fetched from frame buffer for each line when alternate window is disabled. The total number of bytes of data that will be fetched from frame buffer for each line is calculated from: ((AHW1W * bpp / 8) + AW1ALD + n) where: 'bpp' is the number of bits/pixel (color depth), and: 'n' is 0 for 1/2/4/8-bpp, 1 for 16/24-bpp, and 3 for 32-bpp mode. The calculated value will be rounded down to the nearest byte. Normally, this parameter can be set to 1 for all modes.

Table 5-19: GC0BR - Alternate Vertical Window 1 Control

GC1	**GC0BR - Alternate Vertical Window 1 Control** **Index: 2Ch 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*
27-16	*Alternate Vertical Window 1 Height (AVW1H)* This parameter specifies the height of the graphics image window in terms of number of lines when alternate window is enabled. Constraint: AVW1H ≥ + 1. Programmed value = actual value -1. The Alternate Vertical Window 1 End (AVW1E) position with respect to the top edge of the display area is calculated from: AVW1S + AVW1H + 1.
31-28	*Reserved*

Table 5-20: GC0CR - Window 1 Start Address

GC1	**GC0CR - Window 1 Start Address** **Index: 30h Reset value: xxxx-xxxxh**

Table 5-20: GC0CR - Window 1 Start Address

20-0	*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. If the mode is more than 8 bits/pixel then the start address is restricted to pixel boundary.
31-21	*Reserved*

Table 5-21: GC0DR - Alternate Window 1 Start Address

GC1	**GC0DR - Alternate Window 1 Start Address** **Index: 34h Reset value: xxxx-xxxxh**
20-0	*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. If the mode is more than 8 bits/pixel then the start address is restricted to pixel boundary.
24-21	*Reserved*
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.

Table 5-22: GC0ER - Window 1 Stride

GC1	**GC0ER - Window 1 Stride** **Index: 38h Reset value: xxxx-xxxxh**
15-0	*Window 1 Stride (W1ST)* 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 and is programmed in 2's complement (signed) format. It is recommended that this parameter be specified in multiple 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 and is programmed in 2's complement (signed) format. It is recommended that this parameter be specified in multiple of frame buffer bus width (128 bits).

Table 5-23: GC0FR - Reserved

GC1	**GC0FR - Reserved** **Index: 3Ch Reset value:?h**
31-0	*Reserved*

Table 5-24: GC10R - Hardware Cursor 1 Position

GC1	**GC10R - Hardware Cursor 1 Position** **Index: 40h 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. Note that hardware cursor can only be displayed within the graphics image window area.
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 image window area. Programmed value = actual value.
15-12	*Reserved*
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 image window area. Programmed value = actual value.
31-28	*Reserved*

Table 5-25: GC11R - Hardware Cursor 1 Start Address and Offset

GC1	**GC11R - Hardware Cursor 1 Start Address and Offset** **Index: 44h 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.
10-0	*Hardware Cursor 1 Start Address (HC1SA)* This parameter specifies bits 20-10 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.
15-11	*Reserved*
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*

Table 5-26: GC12R - Hardware Cursor 1 Foreground Color

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

Table 5-27: GC13R - Hardware Cursor 1 Background Color

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

Table 5-28: GC14R to GC1FR - Reserved

GC1	**GC14R to GC1FR - Reserved** **Index: 50h to 7Ch Reset value:?h**
31-0	*Reserved*

Table 5-29: GC20R to GC3FR - Graphics Controller 2 Registers

GC1	**GC20R to GC3FR - Graphics Controller 2 Registers** **Index: 80h to FFh Reset value:** These registers control Controller 2 and their functionality are identical to GC00R to GC1FR except for GC01R, which is specific for CRT DAC.
31-0	Graphics Controller 2 register bits

Table 5-30: GC21R - Graphics Controller CRC Control

GC	**GC21R - Graphics Controller CRC Control** **Index: 84h Reset value: xxxx-xx00h**	
0	*CRC Enable* This bit enable 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* 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 29-8. 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	Read CRC result for blue datapath.
	01	Read CRC result for green datapath.
	10	Read CRC result for red datapath.
	11	Reserved.

Table 5-30: GC21R - Graphics Controller CRC Control

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* These bits are not implemented.

Table 5-31: GC40R to GCFFR – Reserved

GC	**GC40R to GCFFR – Reserved** **Index: 100h to 3FCh Reset value:?h**
31-0	*Reserved*

C100R Through C1FFR: Color Palette

These 256 registers specify the color lookup table for graphics controller 1 and 2. In the Main Window, 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 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

Table 5-32: C100R to C1FFR - Graphics Controller Color Palette

C1	**C100R to C1FFR - Graphics Controller Color Palette** **Index: 000h to 3FCh Reset value: xxxx-xxxxh** Color palette is shared between Controller 1 and Controller 2 therefore putting restriction that when both controllers are using the palette the content of the palette must be common for the two controllers. Also when both Controller 1 and Controller 2 are using the color palette, the palette can only be written during the vertical blank time of Controller 1. The color palette can be written at any time when only one graphics controller is using the color palette.
7-0	*Red Color Palette* This parameter specifies the red color values.
15-8	*Green / Gray Color Palette* This parameter specifies the green color values in color modes or the gray level values in monochrome modes.

C100R Through C1FFR: Color Palette

These 256 registers specify the color lookup table for graphics controller 1 and 2. In the Main Window, 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 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

Table 5-32: C100R to C1FFR - Graphics Controller Color Palette

23-16	*Blue Color Palette* This parameter specifies the blue color values.
31-24	*Reserved* These bits are not implemented.

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-, 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

Table 5-33: GC20R: Graphics Controller 2 Control Register

GC2	**GC20R to GC3FR - Graphics Controller 2 Registers** **Index: 80h 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	Reserved
	101	16-bpp (RGB565) graphics.
	others	Reserved.

Table 5-33: GC20R: Graphics Controller 2 Control Register

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	Reserved
	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.

Table 5-33: GC20R: Graphics Controller 2 Control Register

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.	

GC21R: Graphics Controller CRC Result Register

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

Table 5-34: GC21R: Graphics Controller CRC Result Register

GC2	**GC21R - Graphics Controller CRC Control** **Index: 84H 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)*	

GC22R: Horizontal Display Timing Control Register

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

Table 5-35: GC22R: Horizontal Display Timing Control Register

GC2	**GC20R to GC3FR - Graphics Controller 2 Registers** **Index: 88h Reset value:???** These registers control Controller 2 and their functionality are identical to GC00R to GC1FR except for GC01R, which is specific for CRT DAC. If image window expansion is supported in Controller 2, then GC38R and GC39R are also used
31-0	Graphics Controller 2 register bits
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)*

GC23R: Vertical Display Timing Control Register

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

Table 5-36: GC23R: Vertical Display Timing Control Register

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)*

GC24R: Horizontal SYNC Control Register

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

Table 5-37: GC24R: Horizontal SYNC Control Register

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)*

GC25R: Vertical SYNC Control Register

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

Table 5-38: GC25R: Vertical SYNC Control Register

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)* These bits are not implemented.
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)*

GC26R and GC27R: Reserved

These registers are reserved for future use.

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.

Table 5-39: GC28R: Graphics Controller 2 Main Window Horizontal Control Register

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 $\geq$ HW2S + 4. Programmed value = Horizontal Window 2 Start + Horizontal Window 2 Width.
31-28	*Reserved (0)*

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.

Table 5-40: GC29R: Graphics Controller 2 Main Window Vertical Control Register

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)* These bits are not implemented.
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 $\geq$ VW2S + 1. Programmed value = Vertical Window 2 Start + Vertical Window 2 Height.
31-28	*Reserved (0)*

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.

Table 5-41: GC2AR: Graphics Controller 2 Alternate Window Horizontal Control Register

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 $\geq$ AHW2S + 4. Programmed value = Alternate Horizontal Window 2 Start + Alternate Horizontal Window 2 Width.
31-28	*Reserved (0)*

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.

Table 5-42: GC2BR: Graphics Controller 2 Alternate Window Vertical Control Register

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 $\geq$ AVW2S + 1. Programmed value = Alternate Vertical Window 2 Start + Alternate Vertical Window 2 Height.
31-28	*Reserved (0)*

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.

Table 5-43: GC2CR: Main Window 2 Start Address

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)*

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.

Table 5-44: GC2DR: Alternate Window 2 Start Address

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.

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.

Table 5-45: GC2ER: Window 2 Stride Register

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).

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.

Table 5-46: GC2FR: Window 2 Line Size Register

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)* These bits are not implemented.
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)*

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).

Table 5-47: GC30R: Hardware Cursor 2 Position

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)*

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.

Table 5-48: GC31R: Hardware Cursor 2 Start Address and Offset Register

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)*

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.

Table 5-49: GC32R: Hardware Cursor 2 Foreground Color Register

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)*

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.

Table 5-50: GC33R: Hardware Cursor 2 Background Color Register

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)*

GC34R Through GC3FR: Reserved

These registers are reserved for future use.

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.

FP00R: Flat Panel Control Register

Table 5-51: 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 enables flat panel interface logic. If these bits are 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 monochrome 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.

Table 5-51: FP00R: Flat Panel Control Register

	11	Dither pattern 3.
11-10	*Reserved (R/W)* This bit must be programmed to 0.	
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.
23-18	*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 value of ‘011000’ is expected to produce the best result.	
26-24	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 most panels, the default value of ‘000’ is expected to produce the best result.	
27	Dither Pattern Adjust 3 This bit may modify the selected dither pattern. For most panels, the default value of '0' is expected to produce the best result.	
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.	

Table 5-51: FP00R: Flat Panel Control Register

	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.

FP01R: Flat Panel Output Pin Control Register

Table 5-52: 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	*Flat panel Display Enable Control* This bit controls output on FDE pin. For STN panels, this bit is effective only if bit 3 is programmed to 1.	
	0	FDE pin outputs flat panel composite display enable signal.
	1	FDE pin outputs flat panel horizontal display enable signal.
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 FD2 pin (STN panel only)* Setting this bit will output shift clock on FD2 pin and pixel clock on FSCLK pin. Setting this bit is effective for STN panel only.	
	0	FD2 pin will be used to output data and FSCLK will output shift clock.
	1	FD2 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).

Table 5-52: FP01R: Flat Panel Output Pin Control Register

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). When set to '0', this bit overrides Bits '8' and '9'.	
	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 FD2 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 FD2 pin) will also be disabled during vertical blank time if bit 7 is set.
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 when bit 7 is set. 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.	
	0	Extra LP is disabled.
	1	Extra LP is enabled.
11	*Reserved (R/W)* This bit must be programmed to 0.	
13-12	*FD2 Drive Strength* These bits control output drive strength of FD2 pin. This is useful when FD2 pin is used to output shift clock.	
	00	Maximum drive strength (approximately 16 mA).
	01	Medium drive strength.
	10	Medium drive strength
	11	Minimum drive strength.
15-14	Flat Panel Data Drive Strength These bits control output drive strength of FD[23:3,1:0], FDE, and FDI pins. Note that FHSYNC and FVSYNC are not controlled by these bits and that they have minimum drive strength.	
	00	Maximum drive strength (approximately 16 mA).
	01	Medium drive strength.
	10	Medium drive strength
	11	Minimum drive strength.

Table 5-52: FP01R: Flat Panel Output Pin Control Register

16	*Flat Panel Data bit 2 Polarity* This bit controls polarity of flat panel data bit 2 (FD2) pin. This is effective regardless of whether FD2 is used to output data bit 2 or shift clock. This bit is effective only when flat panel is enabled.	
	0	FD2 output is active high. If FD2 is used to output shift clock then flat panel data output is on rising edge of FD2 and should be latched externally with falling edge of FD2.
	1	FD2 output is active low. If FD2 is used to output shift clock then flat panel data output is on falling edge of FD2 and should be latched externally with rising edge of FD2.
17	*Flat Panel Data Polarity* This bit controls polarity of most flat panel data pins (FD[23:0] except for FD2). This bit is effective only when the flat panel is enabled. FD2 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 FD2 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	Flat Panel Shift Clock (FSCLK) Drive Strength These bits control output drive strength of FSCLK pin.	
	00	Maximum drive strength (approximately 16 mA).
	01	Medium drive strength.
	10	Medium drive strength
	11	Minimum drive strength.
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 FD2 pin.	

Table 5-52: FP01R: Flat Panel Output Pin Control Register

31-27	*Reserved (R/W)* These bits must be programmed to 0.

FP02R: Flat Panel General Purpose Output Control Register

Table 5-53: FP02R: Flat Panel General Purpose Output Control Register

FP	**FP02R – Flat Panel General Purpose Output Control Index:8 Reset value: xx00-0000h**	
1-0	*GPO0 Enable* These bits control the ENCTL pin.	
	00	ENCTL pin is used for Enable CTL output.
	01	ENCTL pin is used as general-purpose output 0 (GPO0).
	10	ENCTL pin is used to output oscillator clock.
	11	ENCTL pin is used to output PLL3 clock.
3-2	*GPO1 Enable* These bits control the 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.
9-8	*GPO4 Enable* This bit controls ENVDD pin.	
	00	ENVDD pin is used for ENVDD output.
	01	ENVDD pin is used as general-purpose output 4 (GPO4).
	1x	Reserved.
11-10	*Flat Panel PWM0, PWM1 Drive Strength* These bits control output drive strength of PWM0, PWM1 pins.	
	00	Maximum drive strength (approximately 16 mA).
	01	Medium drive strength.
	10	Medium drive strength
	11	Minimum drive strength.
13-12	*Flat Panel GPO0, GPO1, GPO2 Drive Strength* These bits control output drive strength of GPO0, GPO1, GPO2 pins when they are in output mode.	

Table 5-53: FP02R: Flat Panel General Purpose Output Control Register

	00	Maximum drive strength (approximately 16 mA).
	01	Medium drive strength.
	10	Medium drive strength
	11	Minimum drive strength.
15-14	*Flat Panel ENVDD, ENCTL, ENVEE Drive Strength* These bits control output drive strength of ENVDD, ENVCTL, ENVEE pins.	
	00	Maximum drive strength (approximately 16 mA).
	01	Medium drive strength.
	10	Medium drive strength
	11	Minimum drive strength.
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.
20	*GPO4 Data* This bit is effective when GPO4 Enable bit is set.	
	0	ENVDD pin is driven low when GP04 Enable bit is set.
	1	ENVDD pin is driven high when GP04 Enable bit is set.
23-21	*Reserved (R/W)* These bits must be programmed to 0.	
31-24	*Reserved (0)*	

FP03R: General Purpose I/O Port Control Register

Table 5-54: 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* These bit control the 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	GPIO0 pin is used o output CRC Blue output flag.
3-2	*GPIO1 Enable* These bits control the 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	GPIO1 pin is used to output CRC Green output flag.
5-4	*GPIO2 Enable* These bits control the 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 power management clock (PMCLK).
	11	GPIO2 pin is used to output CRC Red output flag.
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.	

Table 5-54: FP03R: General Purpose I/O Port Control Register

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)*	

FP04R: STN Panel Control Register

Table 5-55: FP04R: STN Panel Control Register

FP	FP04R - STN Panel Control Index:10H 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	FRC Tuning Control bits 23-16 This parameter specifies tuning value for FRC algorithm.	
30-24	Flat Panel Modulation Period This parameter specifies half of the period of the FMOD signal for STN panels. FMOD signal is generated either using line clock or frame clock depending on the setting of bit 31. Programmed value = (actual period / 2) – 1.	
31	Flat Panel Modulation Clock Select This bit control the clock used to generate FMOD signal.	
	0	FMOD is generated using frame clock.
	1	FMOD is generated using line clock.

FP05R: D-STN Half-Frame Buffer Control Register

Table 5-56: FP05R: D-STN Half-Frame Buffer Control Register

FP	**FP05R - D-STN Frame Buffer Control** **Index: Reset value: XXXX-XXXXH**
13-0	*D-STN Frame Buffer Start Address bits 20-7* 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 1K-bit (128-byte) boundary.
15-14	*Reserved (R/W)* These bits must be programmed to 0.
31-16	*D-STN Frame Buffer End Address 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. NOTE: Bit [20] of the DSTN frame buffer end address is not programmed since DSTN buffer size cannot exceed 1 MByte. Therefore, Bit [20] the DSTN frame buffer end address should be omitted in the programmed value after calculation.

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.

Table 5-57: FP0FR: Pulse Width Modulation Control Register

FP	**FP0FR - Pulse Width Modulation (PWM) Control Index: 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 clock 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 output will be 1/256 duty cycle signal and if this parameter is programmed to FF then the PWM 0 output 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. Note: In order to output a static low signal on PWM 0, bits [7:4] must be programmed to 0.	
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.

Table 5-57: FP0FR: Pulse Width Modulation Control Register

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 output will be 1/256 duty cycle signal and if this parameter is programmed to FFh then the PWM 1 output will be static high signal. If enabled, the PWM 1 signal is 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 1 is inactive, the PWM 1 signal will also be driven low.	

FP10R to FP2FR: Frame-Rate Control Pattern Registers

Table 5-58: FP10R to FP2FR: Frame-Rate Control Pattern Registers

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

FP30R to FP37R: Frame-Rate Control Weight Registers

These registers should not be written to. They are used by the MQ-200 device drivers.

Table 5-59: FP30R to FP37R: Frame-Rate Control Weight Registers

FP	**FP30R to FP37R - FRC Weight** **Index: Reset value: xxxx-xxxxh** FP30R, FP31R, and FP32R are also used to specify the expected CRC result during the CRC testing for blue, green, and red datapaths for CRT.
31-0	*FRC Pattern* These registers specify FRC weights. FRC utility will be provided to change this data.

Graphics Engine Programming Information

The MQ-200 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. Full Bresenham line drawing algorithm including arbitrary line draw functionality is implemented in the MQ-200. This hardware implementation is designed with consideration to Windows CE driver architecture to achieve the most optimized performance. The MQ-200 line draw implementation also supports ROP functions which means that when a line draw command is issued, it can be combined with a destination ROP.

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-200 will program the GE directly. These examples are provided for reference only. The GE register set is described in detail in the next section.

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
```

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.

Overlapping 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
```

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,

Table 5-60: FIFO 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,

Table 5-61:

	……..			

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)

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.

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
```

Pattern BitBLT

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

Example A: Color Pattern BitBLT

```
8-bpp Screen mode
Destination coordinate = DestX, DestY
Width W and height H
Pattern data : ColorPattern[64] of 8 x 8 8-bpp 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 1-bpp 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
```

Regular BitBLT - Screen to Memory

This operation is currently not supported by the MQ-200 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.

Line Drawing

Full Bresenham line drawing algorithm including arbitrary line draw functionality is implemented in the MQ-200. This hardware implementation is designed with consideration to Windows CE driver architecture to achieve the most optimized performance. The MQ-200 is capable of drawing any non-patterned line of a single pixel width between any 2 points on screen and the line drawn is fully compatible with internal Windows CE line drawing implementation. The MQ-200 line draw implementation also supports ROP functions which means that when a line draw command is issued, it can be combined with a destination ROP.

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:

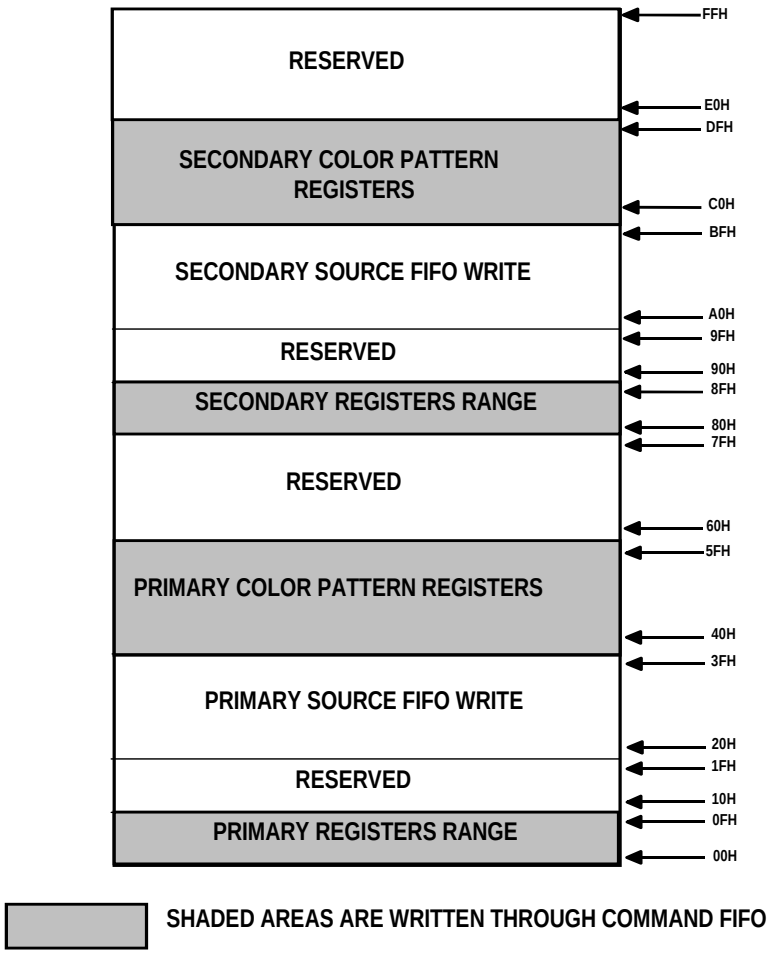


Figure 5-62: Figure 5F: Double-Word Address Space for Graphics Engine Registers

GE00R: Primary Drawing Command Register

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

Table 5-63: GE00R: Primary Drawing Command Register

GE	*GE00R - Drawing Command Register* Index: Power on Reset value: xxxx-xxxxh This register defines the drawing command to be performed by the Graphics Engine.	
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.
	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.
	011	Anti-Aliasing For anti-aliasing command, the source actually is 4-bpp and destination read is needed. MQ200 supports anti-aliasing for 16-bpp and 32 bpp.
	100	Line Draw
	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.	
	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.

Table 5-63: GE00R: Primary Drawing Command Register

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 color pattern register GE40R and GE41R and the foreground and background color is programmed in register GE42R and GE43R.**
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 Source or Mono Pattern Transparency Enable* Setting this bit enables transparency depending on mono pattern data (bit 15 is set) or mono source data (bit 14 is set). If both bits are set, this bit will enable mono source transparency.	
	0	Mono transparency is disabled.
	1	Mono source or mono pattern transparency is enabled. Either foreground color or background color is defined as transparent depending on bit 19.
19	*Mono Transparency Polarity* This bit defines polarity for mono pattern transparency.	
	0	Background color for mono source or mono pattern is transparent.
	1	Foreground color for mono source or mono pattern is transparent.
20	Memory to screen or off screen to screen mode select.	
	0	Use lined mode definition in **GE09R**.
	1	Use packed mode definition in register **GE09R**.

Table 5-63: GE00R: Primary Drawing Command Register

22-21	*Alpha byte mask selection* For 32bpp, the MSB byte is the alpha byte. This byte can be written to the frame buffer depends on these two bits, they also apply to the color comparison.	
	00	Enable four bytes write to frame buffer in 32bpp mode. If color compare is enable, GE will compare the whole 32 bits against color compare register
	01	Don't write LSB three bytes to frame buffer in 32bpp mode, and only write alpha byte to frame buffer. If color compare is enable, GE will only compare alpha byte against MSB byte of color compare register
	10	Don't write alpha byte to frame buffer, and only write LSB three bytes of 32bpp to frame buffer. If color compare is enable, GE will only compare LSB 24 bits against LSB 24 bits of color compare register.
	11	Don't write all the 32 bits to frame buffer. In this case, there will be no color comparison.
23	*Solid color* Setting this bit forces all pattern data to foreground color if mono pattern bit is set or force all source data to foreground color if mono source bit is set.	
	0	Pattern/source data is specified by contents of Pattern registers or source FIFO.
	1	Solid color. All pattern/source data is forced to the value of Foreground Color register.
24	*Source stride is equal to destination stride*	
	0	Source stride is equal to destination stride, GE09R[11:0] is ignored
	1	Source stride is not equal to destination stride, GE09R[11:0] has to be programmed
25	*Rop2 Code selection*	
	0	The ROP code is bit7 to bit0
	1	The ROP code is ROP2, which means the bit3 to bit0 will be duplicated to bit7 to bit4.
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.
28		
29		
31-30	*Reserved*	

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.

Table 5-64: GE01R: Primary Width and Height Register

GE	GE01R - Width and Height/ line draw Bresenham parameter Register Index: Power on Reset value: xxxx-xxxxh This register is for both BitBlt or line draw.	
For BitBlt, 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*	
27-16	*Source/Destination Window Height* This parameter defines the number of lines in the source/destination window.	
28	*Reserved*	
31-29	*Reserved*	
For line draw, GE use this value to calculate the P_0 in Bresenham algorithm. It is always a negative value		
16-0	This value is either –(dm >>1) – 1 or –(dm >> 1) depends on the Quadrant. Since it is always negative value, it is 2's complement in this register.	
28-17	Major-axis height, if it is x major, this is the width of the line; if it is y major, this is the height of the line	
29	*x-major or y-major*	
	0	x-major
	1	y-major
30	*Decision to draw or not to draw the last pixel of the line*	
	0	Always draw the last pixel of the line (the length of the line in major axis is [27:16])
	1	Not to draw the last pixel of the line (the length of the line in major axis is [27:16] – 1)
31	*Reserved*	

GE02R: Primary Destination Address Register

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

GE03R: Primary Source XY Register

Table 5-65: GE02R: Primary Destination Address Register

GE	GE02R - Destination XY Register/line draw starting X coordinate and major axis delta Index: Power on Reset value: xxxx-xxxxxh This register defines the X and Y starting position of the destination window or line draw starting XY	
11-0	*Destination X Position* This parameter defines the X start position of destination window.	
12	*Reserved*	
15-13	*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.
27-16	*Destination Y Position* This parameter defines the Y start position of destination window.	
28	*Reserved*	
31-29	*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.
For line draw command, this register has different definition:		
11-0	*Starting X coordinate*. It is programmed as the pixel number starts to draw.	
28-12	*17 bits major-axis delta (dm).*	
31-29	*Reserved*	

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

Table 5-66: GE03R: Primary Source XY Register

GE	GE03R - Source XY Register/line draw starting Y coordinate and minor axis delta Index: Power on Reset value: xxxx-xxxxh This register defines the X and Y starting position of the source window or in line draw, it defines delta in major and minor axis.
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.
For line draw command, this register is used to define the delta in major and minor axis, the LSB four bits are fraction bits for precision purpose.	
11-0	*Starting Y coordinate*. It is programmed as the line number starts to draw.
28-12	*17 bits minor-axis delta (dn).*
31-29	*Reserved*

GE04R: Primary Color Compare Register

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

Table 5-67: GE04R: Primary Color Compare Register

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.
31-0	*Destination Transparent Color* This is either 8-bpp or 16-bpp or 32-bpp color for destination color transparency.

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 clipping rectangle will be clipped.

Table 5-68: GE05R: Primary Clip Left/Top Register

GE	GE05R - Clip Left/Top Register Index: Power on Reset value: xxxx-xxxxh This register specifies the left edge and the top edge of the clipping window. All destination (output) pixels outside the clipping rectangle will be clipped.
11-0	*Left Edge of Clipping Rectangle* These parameters specify the left edge of the clipping rectangle. All destination (output) pixels with X coordinate less than this value will not be written.
15-12	*Reserved*
27-16	*Top Edge of Clipping Rectangle* These parameters specify the top edge of the clipping rectangle. All destination (output) pixels with Y coordinate less than this value will not be written.
31-28	*Reserved*

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 clipping rectangle will be clipped.

Table 5-69: GE06R: Primary Clip Right/Bottom Register

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

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.

Table 5-70: GE07R: Primary Source and Pattern Offset Register

GE	*GE07R - Foreground Color Register for mono source* Index: Power on Reset value: xxxx-xxxxh This register defines the non-transparent foreground color when mono source is set.
31-0	*Foreground Color* This is 8bpp, 16bpp or 32bpp foreground color value. Bits 7-0 are used for 8-bpp mode. Bits 15-0 are used for 16bpp and bits 31-0 are used for 32bpp.

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.

Table 5-71: GE08: Primary Foreground Color Register / Rectangle Fill Color

GE	**GE08R - Background Color Register** **Index: Power on Reset value: xxxx-xxxxh** This register defines the non-transparent background color when mono source is reset
31-0	*Background Color* *This is 8-bpp, 16-bpp or 32bpp background color value. Bits 7-0 are used for 8-bpp mode. Bits 15-0 are used for 16bpp and bits 31-0 are used for 32bpp.*

Table 5-72: GE09R - Source Stride/Offset Register

GE	**GE09R - Source Stride/Offset Register.** **Index: Power on Reset value: xxxx-xxxxh** This register defines the stride/offset for source line. There are two different definitions for source line depends on lined mode or packed mode. For lined mode set up, usually it needs stride and initial offset, if it is mono source, there will be bit and byte offset, if it is color source, only byte offset is needed. It is BIU's responsibility to write dummy write to the source FIFO if the stride is less than 16 bytes.	
11-0	*Source Line Stride* This parameter specifies the number of bytes from 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). 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.	
24-12	*Reserved*	
27-25	These bits are reserved in on screen to on screen BitBlt. But for mono to on screen lined mode, these three bits are for bits offset for the first pixel of each line.	
	000	Mono source starts at bit 0.
	001	Mono source starts at bit 1.
	010	Mono source starts at bit 2.
	011	Mono source starts at bit 3.
	100	Mono source starts at bit 4.
	101	Mono source starts at bit 5.
	110	Mono source starts at bit 6.
	111	Mono source starts at bit 7.
30-28	These bits are reserved for on screen to on screen BitBlt. But for color source system to screen BitBlt, these three bits are for byte offset of the first line. For mono source off screen to on screen lined mode, the byte offset can be calculated from source address, byte offset are not needed.	
	000	Mono/color source line starts at byte 0.
	001	Mono/color source line starts at byte 1.
	010	Mono/color source line starts at byte 2.
	011	Mono/color source line starts at byte 3.

Table 5-72: GE09R - Source Stride/Offset Register

	100	Mono/color source line starts at byte 4.
	101	Mono/color source line starts at byte 5.
	110	Mono/color source line starts at byte 6.
	111	Mono/color source line starts at byte 7.
31	reserved	
For packed mode set up, it can happen at both mono source and color source, For mono source, it needs initial byte and bit offset and subsequent byte and bit space. For off screen to on screen BitBlt, it also needs amount of 128 bits pass to MIU. For color source, the space is always in byte boundary		
2-0	*Initial Mono Source Bit Offset* This parameter is used to indicate starting bit offset for first line of mono source from either memory or screen. For color source, these bits have to set to zero.	
	000	Mono source line starts at bit 0.
	001	Mono source line starts at bit 1.
	010	Mono source line starts at bit 2.
	011	Mono source line starts at bit 3.
	100	Mono source line starts at bit 4.
	101	Mono source line starts at bit 5.
	110	Mono source line starts at bit 6.
	111	Mono source line starts at bit 7.
5-3	*Initial Mono/Color Source Byte Offset* This parameter is used to indicate starting byte for first line of mono source data from system memory or the initial byte offset for mono off screen source data. For 32 bit CPU interface, only two bits are needed, for 64 bit DMA mode, three bits are needed.	
	000	Mono/color source line starts at byte 0.
	001	Mono/color source line starts at byte 1.
	010	Mono/color source line starts at byte 2.
	011	Mono/color source line starts at byte 3.
	100	Mono/color source line starts at byte 4.
	101	Mono/color source line starts at byte 5.
	110	Mono/color source line starts at byte 6.
	111	Mono/color source line starts at byte 7.
15-6	*Number of 16 bytes amount that MIU need to fetch from frame buffer* For off screen to on screen packed mode, these 10 bits represents Amount of 16 bytes need to be fetched by MIU minus one, the Maximum amount it can address is 16k bytes, which is the off screen mono font or mono image GE can support for packed mode.	
27-16	*Reserved*	

Table 5-72: GE09R - Source Stride/Offset Register

27-25	*Bit Space between lines* For system to screen or off screen to screen BitBlt, these bits are used for bit space between lines. For color source, these bits have to set to zero. GE always uses these bits and concatenate with Byte space (bit 31-28) to calculate the total bit space for packmode. Therefore if it is mono source, bit[31:25] becomes total bit space between two line. If it is color source, the byte space is programmed into bit[31-28]
31-28	*Byte space between lines (actual space byte number plus one)* This parameter is used to indicate the byte space between the end of the first line and the beginning of the next line. The space indication is in the double word boundary (if it is 32 bits CPU mode) or quad word boundary (if it is 64 bits DMA mode). For color source packed mode, it represents LSB four bits of stride. In 32bits mode, use three bits and bit 31 force to zero. In 64 bits mode, use four bits.

Table 5-73: GE0AR – Destination Stride Register and Color Depth

GE	**GE0AR – Destination stride register and color depth** **Index: Power on Reset value: xxxx-xxxxh** This is the destination stride and color depth register, usually this is programmed once only.	
11-0	*Destination Line Stride* This parameter specifies the number of bytes from 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 destination window and the first pixel of mono/color source window if source comes from screen (display memory). For clipping, the destination stride has to be on the 16 bytes (128 bits) boundary, the least significant four bits are always zero.	
29-12	*Reserved*	
31-30	*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 address of first pixel of mono source window from screen.	
	00	8 bpp
	01	16 bpp
	10	reserved
	11	32 bpp

Table 5-74: GE0BR - Image Base Address Register

GE	**GE0BR - Base Address Register** **Index: Power on Reset value: xxxx-xxxxh** This is the base (start) address of the image in the display memory.	
20-0	*Base Address* This parameter defines the start address of the image in the display memory. This is byte address but the lower 3 bits are internally forced to zeros therefore this parameter is aligned to quad-word boundary. During the register read, always read the bits 20-3, because the LSB three bits are not implemented.	
28-21	*Reserved*	
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.

Table 5-75: GE1FR - Test Mode Read Register

GE	**GE1FR - Test Mode Read Register (Read Only)** **Index: Power on Reset value: xxxx-xxxxh** This is a read-only register which can be used to read internal status in test mode. If test mode is disabled, this reading this register will return unpredictable value.	
	For address calculation results: (GE0DR[31:30] are set to 00)	
	20-0	It contains the source address results from the ALU.
	31-21	Don't care, set to 0.
	For clipping left and right: (GE0DR[31:30] are set to 01)	
	8-0	It contains the left difference between shcleft and shdstx.
	15-9	Don't care, set to 0.
	24-16	It 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	It contains the top difference between shctop and shdsty.
	15-12	Don't care, set to 0.
	27-16	It contains the bottom difference between shcbot and shdsty.
	31-28	Don't care, set to 0.

Table 5-76: GE20R to GE3FR

GE	**GE20R to GE3FR** **Index: Power on Reset value: xxxx-xxxxh** Reserved

Table 5-77: GE40R to GE5FR - Primary Color Pattern Register

GE	**GE40R to GE5FR - Primary Color Pattern Register** **Index: Power on Reset value: xxxx-xxxxh** These registers store the 8x8 color pattern data in packed mode for 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. In 32-bpp mode, each register stores 1 pixel of pattern data. The first line of pattern data is stored in GE40R to GE47R. for this mode, the amount of color pattern registers are configured as 8x4x32, so only half of the pattern are stored
31-0	*Primary Color Pattern Data*

*Note: *** For mono pattern cases, since it is mutually exclusive to color pattern, GR40R to GR43R can be used for mono pattern definition:*

Table 5-78: GE40R - Mono Pattern Register 0

GE	GE40R - Mono Pattern Register 0 Index: Power on Reset value: xxxx-xxxxh This register contains the first four lines of mono pattern data.
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.

Table 5-79: GE41R - Mono Pattern Register 1

GE	**GE41R - Mono Pattern Register 1** **Index: Power on Reset value: xxxx-xxxxh** This register contains the second four lines of mono pattern data.
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.

Table 5-80: GE42R - Foreground Color Register

GE	**GE42R - Foreground Color Register** **Index: Power on Reset value: xxxx-xxxxh** This register defines the non-transparent foreground color when mono pattern is set
31-0	*Foreground Color* This is 8-bpp, 16-bpp or 32-bpp foreground color value. Bits 7-0 are used for 8-bpp mode.

Table 5-81: GE43R - Background Color Register

GE	**GE43R - Background Color Register** **Index: Power on Reset value: xxxx-xxxxh** This register defines the non-transparent background color when mono pattern is '0'.
31-0	*Background Color* *This is 8-bpp, 16-bpp or 32-bpp background color value. Bits 7-0 are used for 8-bpp mode.*

Table 5-82: Reserved

GE	**GE80R to GE8BR - Secondary Drawing Command Register** **Index: Power on Reset value:** These registers are used for secondary command pipeline and their functionality is identical to GE00R to GE0BR.

Table 5-83: GEA0R to GEBFR

GE	**GEA0R to GEBFR** **Index: Power on Reset value: xxxx-xxxxh** *Reserved.*

Table 5-84: GEC0R to GEDFR - Secondary Color Pattern Register

GE	**GEC0R to GEDFR - Secondary Color Pattern Register** **Index: Power on Reset value: xxxx-xxxxh** 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. In 32-bpp mode, each register stores 1 pixel of pattern data. The first line of pattern data is stored in GE40R to GE47R. for this mode, the amount of color pattern registers are configured as 8x4x32, so only half of the pattern are stored
31-0	*Secondary Color Pattern Data*

*Note: ******* Both rectangle fill and line draw can be implemented in mono source BitBlt or mono pattern BitBlt. If it is implemented in source BitBlt, the mono source and solid color bit in drawing command register have to be set. If it is implemented in pattern*

BitBlt, the mono pattern and solid color bit have to be set.

***** *For anti-aliasing operation, the ROP code has to be source and destination ROP. The foreground color is from source foreground color register (GE07R).*

CPU Interface Programming Information

CPU Register Definition

Table 5-85: CPU Interface (CIF)

CC00R	CPUCTRL Register Index: Reset value: 0000-0030h	
1-0	*Reserved*	
1	*Reserved*	
2	*Reserved*	
3	CLKRUN Enable for PCI interface	
	0	CLKRUN is disabled
	1	CLKRUN is enabled (PCI Bus mode)
5-4	CPU interface drive strength control.	
	11	Maximum drive strength
	10	Medium drive strength
	01	Medium drive strength
	00	Minimum drive strength
31-6	*Reserved*	

Table 5-86: Source FIFO/Command FIFO/GE 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 FULL
	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

Table 5-86: Source FIFO/Command FIFO/GE Status Register

	01110	Command FIFO has 14 locations free
	01111	Command FIFO has 15 locations free
	10000	Command FIFO is EMPTY
7-5	Reserved	
11-8	Source FIFO Status Register (SFS). Source FIFO is 8 - deep.	
	0000	Source FIFO is FULL
	0001	Source FIFO has 1 location free
	0010	Source FIFO has 2 locations free
	0011	Source FIFO has 3 locations free
	0100	Source FIFO has 4 locations free
	0101	Source FIFO has 5 locations free
	0110	Source FIFO has 6 locations free
	0111	Source FIFO has 7 locations free
	1000	Source FIFO is Empty
15-12	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 command left in it. This signal is provided by the Graphics Engine.
23-17	Reserved	
31-24	Reserved	

Table 5-87: PMR - Power Management Register

PC40R	**PMR – Power Management Register** Index: 40h in Configuration space Reset value: 0621-0001h This register is part of 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.*

Table 5-88: PMCSR – Power Management Control/Status Register

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 make transition to D4 State. In this state, all modules in the chip are powered down except for small portion of CPU interface, which is needed to monitor register write 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 pin is monitored: CKIO, FRAME#, CS#. When FRAME# and CS# are active in D3 state then DATA[1:0] are enabled and latched into this register bit.
31-2	*Reserved (Read Only)* These bits are set to 0's.	

Memory Interface Unit Programming Information

The following sections describe the memory interface unit registers.

Memory Interface Unit Register

Table 5-89: Memory Interface Control 1

MIU	MM00R – MIU interface Control 1 Index: Reset value: XXXX-XXX0h	
0	*MIU enable bit* Setting this bit enables MIU for 2 MB embedded DRAM. 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.	
	0	MIU reset is enabled (reset value)
	1	MIU reset is disabled.
2	*Embedded DRAM 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.	
	0	Internal DRAM hardware reset is enabled (reset value)
	1	Internal DRAM hardware reset is disabled.
(31-3)	Reserved	

Table 5-90: MM01R – MIU Interface Control 2

MIU	MM01R – MIU interface Control 2 Index: Reset value: XXXX-XXX0h	
0	*Memory clock source* This bit specifies the clock source for the memory clock.	
	1	Bus clock is used as memory clock source.
	0	PLL 1 output is used as memory clock source.
1	*Memory slow refresh enable bit* This bit controls the refreshing during MIU power down mode for D0 mode only.	
	0	Memory refreshing is disabled during MIU power down mode.
	1	Memory refreshing is enabled during MIU power down mode.
2	*Page break enable for CPU* This bit controls the handling of the paging for CPU requests.	
	0	Leave the page open after CPU memory cycles are performed.
	1	Enable the page break after CPU memory cycles are performed.

Table 5-90: MM01R – MIU Interface Control 2

3	*Page break enable for GC1* This bit controls the handling of the paging for GC1 requests.	
	0	Leave the page open after GC1 memory cycles are performed.
	1	Enable the page break after GC1 memory cycles are performed.
4	*Page break enable for GC2* This bit controls the handling of the paging for GC1 requests.	
	0	Leave the page open after GC2 memory cycles are performed.
	1	Enable the page break after GC2 memory cycles are performed.
5	*Page break enable for STN read* This bit controls the handling of the paging for GC1 requests.	
	0	Leave the page open after STN read memory cycles are performed.
	1	Enable the page break after STN read memory cycles are performed.
6	*Page break enable for STN write* This bit controls the handling of the paging for GC1 requests.	
	0	Leave the page open after STN write memory cycles are performed.
	1	Enable the page break after STN write memory cycles are performed.
7	*Page break enable for GE* This bit controls the handling of the paging for GE requests.	
	0	Leave the page open after GE memory cycles are performed.
	1	Enable the page break after GE memory cycles are performed.
11-8	*Reserved (R/W)*	
25-12	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 memory clock. The normal refresh period is 64 ms but it may reduce according to the junction temperature (the other more likely refresh periods are 32 ms and 16 ms). The formula for this parameter is as following: Refresh period = (Most likely 64 ms or 32 ms or 16 ms) ((Refresh period / (MCLK period in ns * 1024) * (10 ** 6)) - 5	
30-26	*Reserved (R/W)*	
31	*EDRAM standby enable for EDRAM normal mode operation* This bit enables the generation of EDRAM power supply standby signal for the normal operation; when the chip is not in the sleep mode.	
	0	Disables the generation of the standby signal when MIU module is active.
	1	Enables the generation of the standby signal when MIU module is active.

Table 5-91: MM02R-Memory Interface Control 3

MIU	**MM02R-Memory Interface Control 3** **Index: Reset value: XXXX-XXXXh**
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 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.

Table 5-91: MM02R-Memory Interface Control 3

	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 1111.	
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 1111.	
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 1111.	

Table 5-91: MM02R-Memory Interface Control 3

25-22	*STN write FIFO threshold* Whenever the number of filled locations in STN write FIFO is equal 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 1111.
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.

Table 5-92: MM03R-Memory Interface Control 4

MIU	**MM03R-Memory Interface Control 4** **Index: Reset value: XXXX-XXXXh**
0	*Read latency request* Writing '1' into this bit causes the generation of an input pulse which triggers the EDRAM latency updating. Before writing 1 into this register bit, the latency value needs to be updated.
31-1	*Reserved*

Table 5-93: MM04R-Memory Interface Control 5

MIU	**MM04R-Memory Interface Control 5** **Index: Reset value: XXXX-XXXXh**	
2-0	*EDRAM latency*	
	001	Latency 1.
	101	Latency 2.
	111	Latency 3.
	other	Reserved.
3	*Enable for the dummy cycle insertion between read and write cycles.*	
	0	Dummy clock cycle insertion between consecutive read/write or write/read commands is disabled.
	1	Dummy clock cycle insertion between consecutive read/write or write/read commands is disabled.
4	*Enable for the dummy cycle insertion between read/write and precharge cycles for the same bank*	
	0	Dummy clock cycle insertion between read/write and precharge cycles for the same bank is disabled.
	1	Dummy clock cycle insertion between read/write and precharge cycles for the same bank is enabled.
7-5	*Reserved (R/W)*	
9-8	*Bank activate 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 >= 40 ns.	
	01	3 memory clocks.
	10	4 memory clocks.
	11	5 memory clocks.
	00	Reserved.

Table 5-93: MM04R-Memory Interface Control 5

10	*Bank activate 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 >= 20 ns.	
	0	2 memory clocks.
	1	3 memory clocks.
11	*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 >= 20 ns.	
	0	2 memory clocks.
	1	3 memory clocks.
12	*Row Cycle time (TRC)* 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 >= 60 ns.	
	0	6 memory clocks.
	1	8 memory clocks.
15-13	*Reserved*	
17-16	*Programmable delay for read clock* These bits specify the delay which needs to inserted between EDRAM clock and memory read clock.	
	00	No delay.
	01	0.5 ns delay.
	10	1 ns delay.
	11	1.5 ns delay
19-18	*Programmable delay for internal memory clock* These bits specify the delay which needs to inserted between EDRAM clock and memory interface clock.	
	00	No delay.
	01	0.5 ns delay.
	10	1 ns delay.
	11	1.5 ns delay
31-20	*Reserved*	

Power Management Unit Programming Information

The following sections describe the power management registers.

Power Management Unit Register Definition

Table 5-94: DC00R: Device Configuration Register 0

DCR	**DC00R: Device Configuration Register 0** **Index: 01h Reset value: 0000-0000h** This register together with PMCSR is read/writable in D3 state for all microprocessor system. For all microprocessor systems, this register must be programmed prior to leaving D3 state upon reset. The PLL output frequency (f_{OUT}) is calculated from $f_{OUT} = f_{REF} * (M+1) / (N+1) / 2^P$ where M, N, and P parameters are programmed in this register and f_{REF} is the reference frequency. The following restrictions are required when programming the PLL: $f_{REF} / (N+1) = f_{IN}$ must not be lower than 1 MHz $f_{REF} * (M+1) / (N+1) = f_{VCO}$ must be in the range of 170 MHz to 440 MHz M value must be in the range of 40 to 256	
0	Oscillator Bypass This bit optionally bypasses the internal clock oscillator.	
	0	Internal clock oscillator is not bypassed. External clock crystal need to be attached.
	1	Internal clock oscillator is bypassed and powered down. External oscillator can be used to drive the oscillator input pin when the clock oscillator is enabled.
1	Oscillator Enable This bit controls clock oscillator (OSCCLK). Note that OSCCLK must be enabled when any of the PLL is enabled. This bit is effective whether or not the internal oscillator is bypassed.	
	0	Clock oscillator is powered down. External oscillator can be used to drive the oscillator input pin.
	1	Clock oscillator can be enabled. This setting can be overwritten in D1 and D2 states. This oscillator is powered down in D3 and D4 states.
2	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	PLL 1 Enable This bit controls PLL 1. This PLL can be used to supply clock for Controller 1 or Controller 2.	
	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.
6-4	PLL 1 P Parameter This parameter determines the output divisor for PLL 1.	
	000	Output divisor is 1.
	001	Output divisor is 2.
	010	Output divisor is 4.
	011	Output divisor is 8.

Table 5-94: DC00R: Device Configuration Register 0

	100	Output divisor is 16.
	others	Reserved.
7	CPU Interface Clock Divisor This parameter determines the PLL 1 output clock divisor factor to generate CPU/Bus Interface clock for systems where this clock is not provided externally.	
	0	CPU Interface clock is PLL 1 clock divided by 1.
	1	CPU Interface clock is PLL 1 clock divided by 2.
12-8	PLL 1 N Parameter This parameter is the divisor factor for the PLL 1. Programmed value = actual value – 1.	
13	Strong Arm Interface Synchronizer Control This parameter determines how fast SA bus interface signals are synchronized internally.	
	0	SA bus interface signals are synchronized on rising edges of internal bus interface clock. This setting should be used if the bus interface clock is very fast (66 MHz or more).
	1	SA bus interface signals are synchronized on rising and falling edges of internal bus interface clock. This setting should be used if the bus interface clock is not very fast (less than 66 MHz).
14	Software Chip Reset Setting this bit will reset the chip except for bus interface unit.	
	0	Software reset is disabled.
	1	Software reset is enabled.
15	Memory Standby Enable Resetting this bit will put the memory power unit in standby mode when the memory is disabled and not being refreshed. This reduces power consumption of the memory power unit when the memory is not being used/ refreshed.	
	0	Memory power unit standby enabled.
	1	Memory power unit standby disabled.
23-16	PLL 1 M Parameter This parameter is the multiplier factor for the PLL 1. Programmed value = actual value – 1.	
24	Oscillator Shaper Disable Setting this bit will disable the shape disable function when internal oscillator is enabled and not bypassed. When oscillator is bypassed or disabled, this bit has no effect. This bit should normally be reset.	
	0	Oscillator waveform shaper is enabled.
	1	Oscillator waveform shaper is disabled.
25	Fast Power Sequencing Resetting this bit will shorten power-sequencing intervals to one PMCLK cycle for boot-up or for testing purpose. This will override bits PM00R[21-18].	
	0	Fast power sequencing enabled.
	1	Fast power sequencing disabled.
27-26	Oscillator Frequency Select This parameter determines the operating frequency of the oscillator.	

Table 5-94: DC00R: Device Configuration Register 0

	00	Oscillator frequency is in the range of .. MHz to .. MHz.
	01	Oscillator frequency is in the range of .. MHz to .. MHz.
	10	Oscillator frequency is in the range of .. MHz to .. MHz.
	11	Oscillator frequency is in the range of .. MHz to .. MHz.
31-28	PLL 1 Trim Value This parameter fine tune PLL 1 characteristics.	

Table 5-95: DC01R: Device Configuration Register 1 (Read Only)

DCR	**DC01R: Device Configuration Register 1 (Read Only)** **Index: 04h Reset value: 0000-00ddh** This register is readable in D3 state for all microprocessor system as well as in D0, D1, and D2 states.	
5-0	Bus Interface mode (Read Only) This parameter returns the current bus interface mode.	
	000001	Reserved
	000010	SH7750 mode.
	000100	VR4111 / VR4121 mode.
	001000	SA1110 mode.
	010000	TX3922 mode.
	100000	PCI mode.
	others	Reserved.
31-6	Reserved (0) These bits are not implemented and will always return zeros.	

Table 5-96: DC02R: Software Register 0

DCR	**DC02R: Software Register 0** **Index: 08h Reset value: 0000-0000h** This register is read/writable in D0-D2 states for display driver purpose only.
31-0	Software Register 0 This is used by display driver only.

Table 5-97: DC03R: Software Register 1

DCR	**DC03R: Software Register 1** **Index: 0Ch Reset value: 0000-0000h** This register is read/writable in D0-D2 states for display driver purpose only.
31-0	Software Register 1 This is used by display driver only.

PMR: Power Management Unit Configuration Register

Table 5-98: Power Management Unit Configuration Register

PM	PM00R – Power Management Miscellaneous Control Index: 00h Reset value: xx00-0000h	
1-0	*Reserved (R/W)* These bits must be programmed to 0.	
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 of 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 when switching power states.
	1	PMCSR[1-0] reflect the power state immediately after it is programmed (before power sequencing is completed).
7-6	*Reserved (R/W)* These bits must be programmed to 0.	
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 D3and 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 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.

Table 5-98: Power Management Unit Configuration Register

	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 is Reset.
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	*Memory Interface Unit Power Sequencing Enable* Setting this bit enables power sequencing when MIU is being enabled/disabled. Normally there should be no need to do power sequencing when enabling or disabling MIU.	
	0	Power sequencing is disabled when MIU is being enabled/disabled.
	1	Power sequencing is enabled when MIU is being enabled/disabled.
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. Power Management Clock is used to refresh the memory.	
	0	Frame buffer memory is not refreshed in D3 State.
	1	Frame buffer memory is refreshed in D3 State.
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. Power Management Clock is used to refresh the memory.	
	0	Frame buffer memory is not refreshed in D4 state.
	1	Frame buffer memory is refreshed in D4 state.
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 cycle.
	01	32 PMCLK cycles.
	10	64 PMCLK cycles.

Table 5-98: Power Management Unit Configuration Register

	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.
23-22	*Reserved (R/W)* These bits must be programmed to 0.	
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*	

Table 5-99: Power Management Unit PM01R – D1 State Control

PM	**PM01R – D1 State Control** **Index: 04h 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 PLL is 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.

Table 5-99: Power Management Unit PM01R – D1 State Control

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 D1 State when MIU is powered-down.
	1	Internal memory is refreshed 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.

Table 5-99: Power Management Unit PM01R – D1 State Control

15-10	*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.
23-20	*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.

Table 5-99: Power Management Unit PM01R – D1 State Control

	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.	

Table 5-100: Power Management Unit – D2 State Control

PM	**PM02R – D2 State Control** **Index: 08h 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 PLL is 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 D2 State when MIU is powered-down.
	1	Internal memory is refreshed D2 State when MIU is powered-down. Note that

Table 5-100: Power Management Unit – D2 State Control

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.
15-10	*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 enable in D2 State if bit 16 is 1 and bit 17 is 1.
	1	Alternate Window 1 is enable 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.

Table 5-100: Power Management Unit – D2 State Control

	1	Cursor 1 can be enabled in D2 State if bit 16 is 1.
23-20	*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.	

Table 5-101: Power Management Unit – PLL 2 Programming

PM	**PM06R – PLL 2 Programming** **Index: 18h Reset value: 0000-0000h** PLL 2 output frequency is calculated in the same manner as PLL 1 output frequency and PLL 2 programming has the same restrictions as PLL 1 programming (see DCR).	
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 L2CLK input.
1	*PLL 2 Bypass* This bit optionally bypasses the PLL 2.	
	0	PLL 2 output is not bypassed.

Table 5-101: Power Management Unit – PLL 2 Programming

	1	PLL 2 is bypassed. L2CLK 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.	
6-4	*PLL 2 P Parameter* This parameter determines the output divisor for PLL 2.	
	000	Output divisor is 1.
	001	Output divisor is 2.
	010	Output divisor is 4.
	011	Output divisor is 8.
	100	Output divisor is 16.
	others	Reserved.
7	*Reserved (R/W)* This bit must be programmed to 0.	
12-8	*PLL 2 N Parameter* This parameter is the divisor factor for the PLL 2. Programmed value = actual value – 1.	
15-13	*Reserved (R/W)* These bits must be programmed to 0.	
23-16	*PLL 2 M Parameter* This parameter is the multiplier factor for the PLL 2. Programmed value = actual value – 1.	
27-24	*Reserved (R/W)* These bits must be programmed to 0.	
31-28	*PLL 2 Trim Value* This parameter fine tune PLL 2 characteristics.	

Table 5-102: Power Management Unit – PLL 3 Programming

PM	**PM07R – PLL 3 Programming** **Index: 1Ch Reset value: 0000-0000h** PLL 3 output frequency is calculated in the same manner as PLL 1 output frequency and PLL 3 programming has the same restrictions as PLL 1 programming (see DCR).	
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 L3CLK input.
1	*PLL 3 Bypass* This bit optionally bypasses the PLL 3.	
	0	PLL 3 output is not bypassed.
	1	PLL 3 is bypassed. L3CLK input is used instead. In this case bit 0 has no effect and PLL 3 should be disabled to save power.

Table 5-102: Power Management Unit – PLL 3 Programming

3-2	*Reserved (R/W)* These bits must be programmed to 0.	
6-4	*PLL 3 P Parameter* This parameter determines the output divisor for PLL 3.	
	000	Output divisor is 1.
	001	Output divisor is 2.
	010	Output divisor is 4.
	011	Output divisor is 8.
	100	Output divisor is 16.
	others	Reserved.
7	*Reserved (R/W)* This bit must be programmed to 0.	
12-8	*PLL 3 N Parameter* This parameter is the divisor factor for the PLL 3. Programmed value = actual value – 1.	
15-13	*Reserved (R/W)* These bits must be programmed to 0.	
23-16	*PLL 3 M Parameter* This parameter is the multiplier factor for the PLL 3. Programmed value = actual value – 1.	
27-24	*Reserved (R/W)* These bits must be programmed to 0.	
31-28	*PLL 3 Trim Value* This parameter fine tune PLL 3 characteristics.	

IN - Interrupt Controller

The Interrupt Controller is part of the CPU Interface. The interrupt Controller has been designed to assert an interrupt to the host processor and this interrupt signal can be programmed to be either active high or active low. Details associate with these registers are as follows:

Table 5-103: Global Interrupt Control Register

IN	**IN00RGlobal 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.

Table 5-104: Interrupt Mask Register

IN	**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	

Table 5-104: Interrupt Mask Register

	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

Table 5-104: Interrupt Mask Register

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

Table 5-105: Interrupt Status Register

IN	**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.

Table 5-105: Interrupt Status Register

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.

Table 5-106: Interrupt Pin Raw Status

IN	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

Chapter 6

Electrical Specifications

Table 6-1: Absolute Maximum Conditions

Symbol	Parameter	Min	Max	Units
V_{CCCORE}	Supply Voltage for the internal Core	-0.5	3.5	V
V_{CC}	Supply Voltage for I/O	-0.5	4.0	V
V_I	Input Voltage for I/O	-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_{CCCORE}	Supply Voltage for the internal Core	2.25	2.5	2.75	V
V_{CC}	Supply Voltage for I/O	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	–	17.62	–	mA
	Full Scale Error		–	–	± 5	%
	DAC to DAC Correlation		–	5	–	%
	DAC Linearity		±2	–	–	LSB

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

Note: Asynchronous timings are shown for a strong arm processor with MCLK = 100 MHz.

- *NEC VR4121 timings are worst-case timings.*

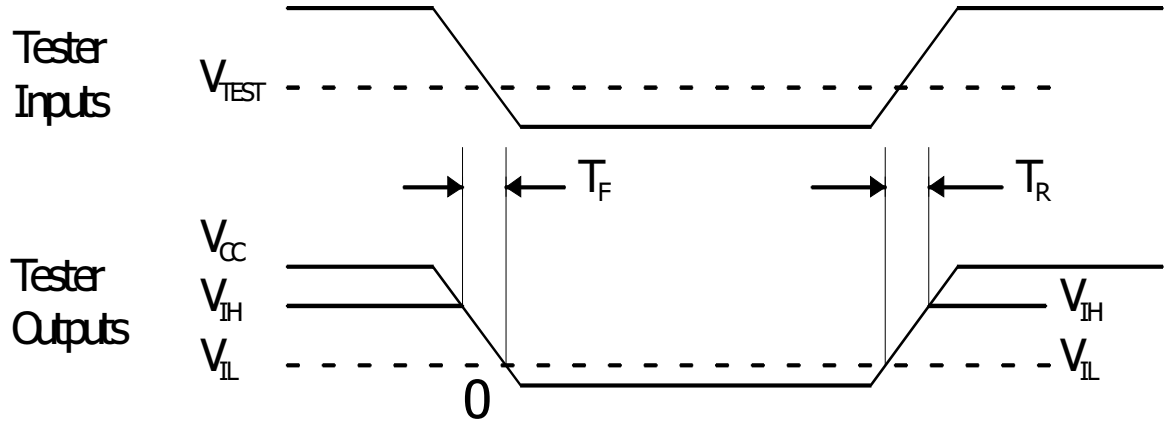


Figure 6-4: AC Test Timing

Table 6-5: 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	%

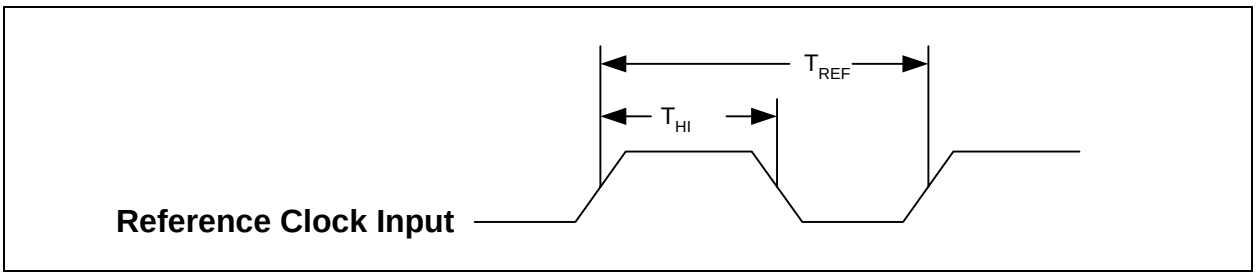


Figure 6-6: Reference Clock Timing

Table 6-7: AC Timing Characteristics - Clock Generator

Symbol	Parameter	Notes	Min	Typical	Max	Units
F_{DCLK}	DCLK Frequency		–	–	135	MHz
F_{MCLK}	MCLK Frequency		–	–	75	MHz

Table 6-8: 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: 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: 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: 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.

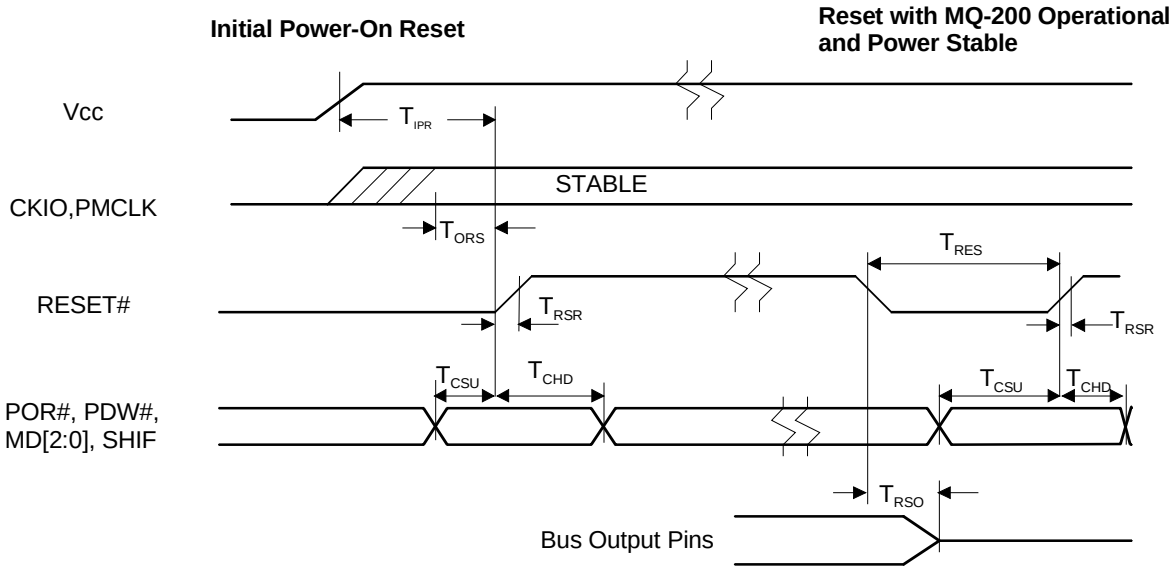


Figure 6-9: Reset Timing

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.

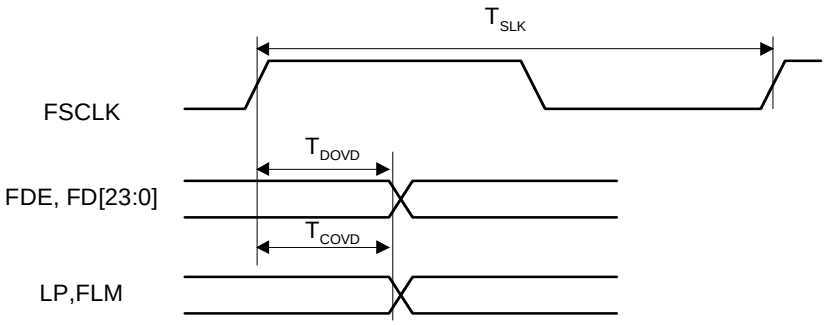


Figure 6-10: Panel Output Timing

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.

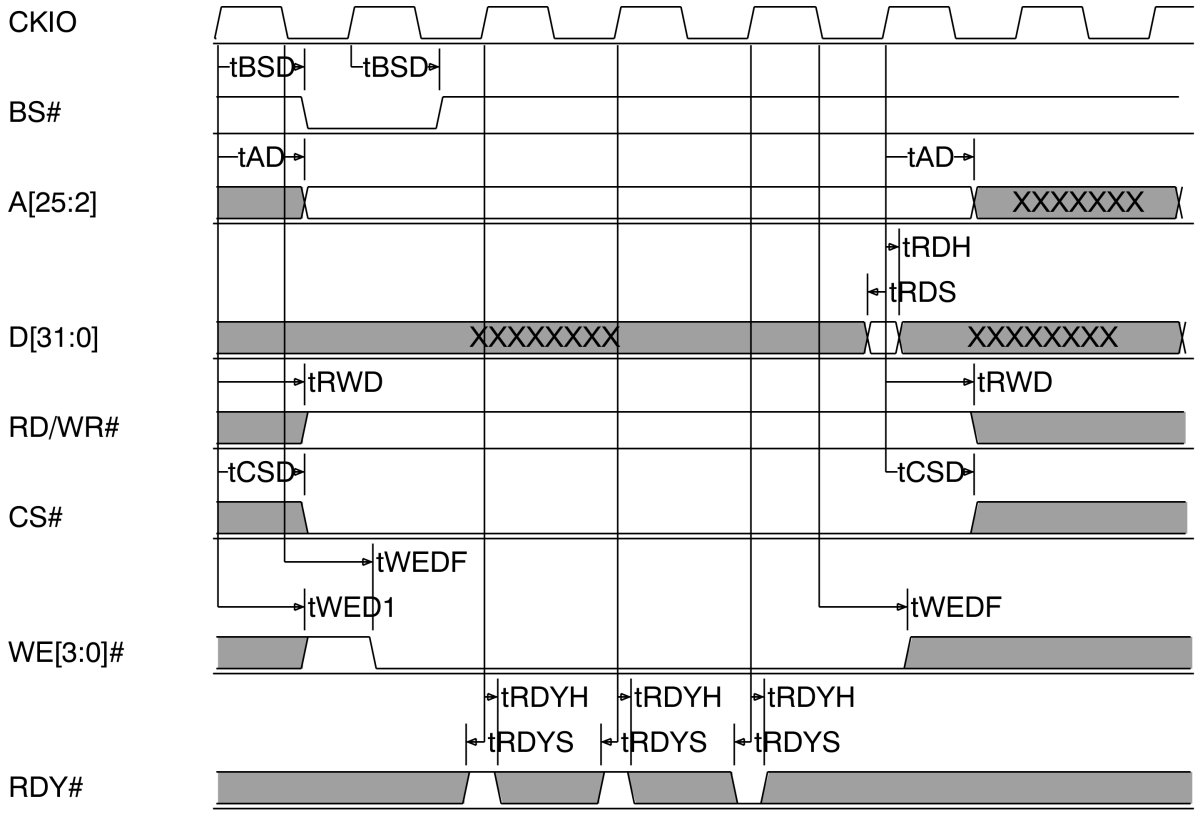


Figure 6-11: CPU Read Timing

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.

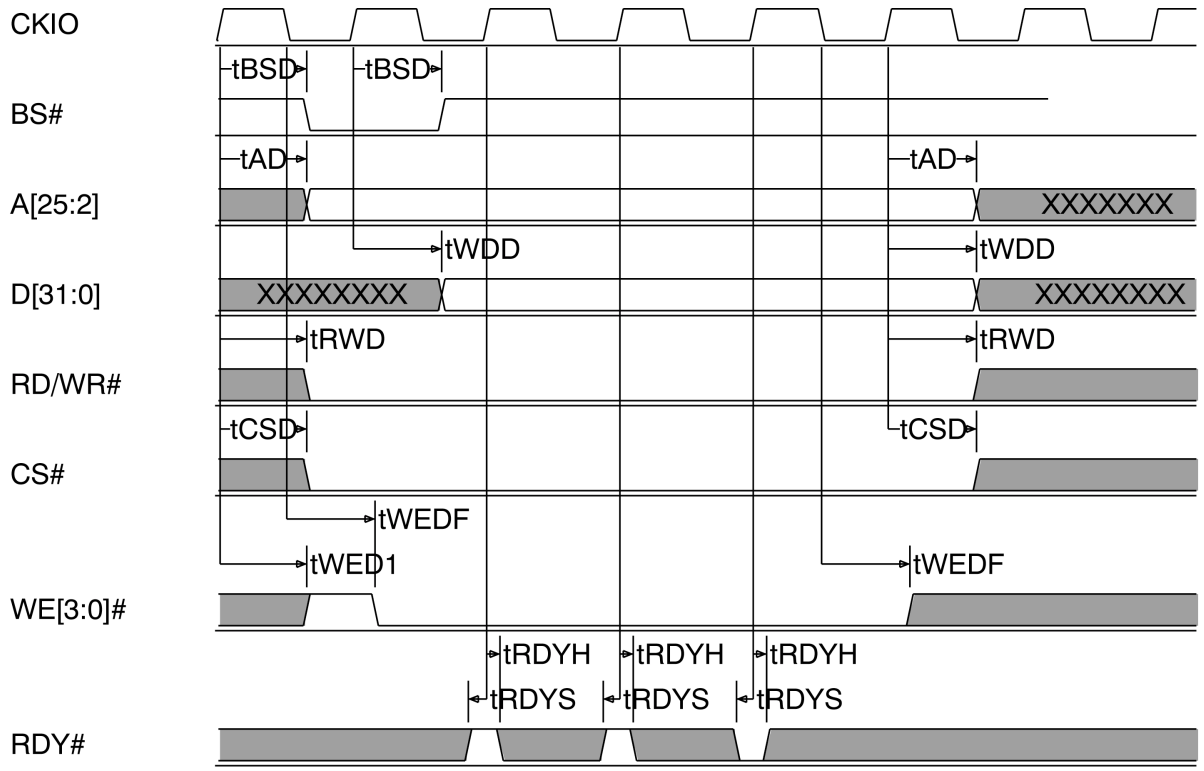


Figure 6-12: CPU Write Cycle Timing

Table 6-13: Intel StrongARM SA-1110 Read Timing

Symbol	Parameter	Min	Max	Units
t_{ASU}	Address setup time with respect to CS*	-	5	ns
t_{DWE}	Output enable delay from CS* asserted	-	20	ns
t_{DCS}	Chip select delay after nOE is deasserted	-	10	ns
t_{DRDY}	nOE deasserted delay after RDY is asserted	15		ns
t_{DRDYA}	RDY signal is asserted based on CS* and address decode	-	2	ns
t_{AH}	Address hold time with respect to nOE decode	-	5	ns

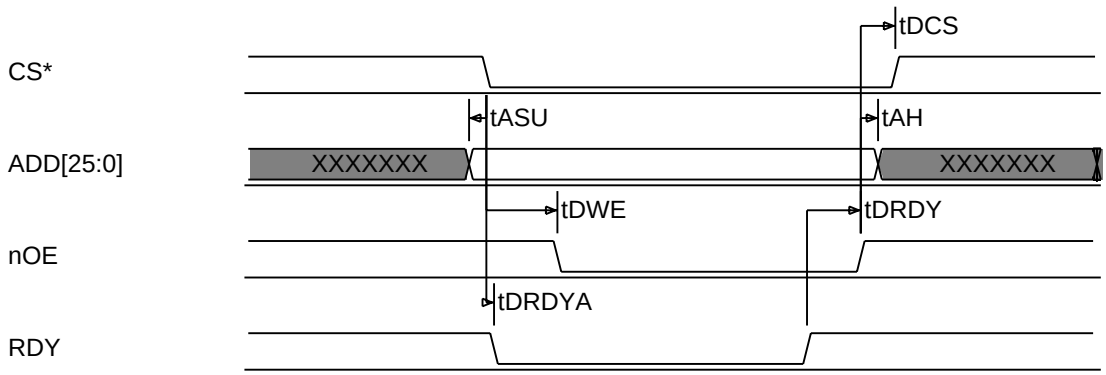


Figure 6-14: Intel StrongARM 1110 Processor Read Timing

Table 6-15: Intel StrongARM 11xx Processor Burst Read Timing

Symbol	Parameter	Min	Max	Units
t_{CSDLY}	Chip select assertion delay after address is valid	-	5	ns
t_{DRD}	Read signal delay from the time CS* is asserted	-	20	ns
t_{DRDYA}	RDY signal is asserted based on CS* and address decode	-	2	ns
t_{ASU}	Address setup time with respect to read	-	5	ns
t_{DREC}	Read recovery time	-	20	ns
t_{DRDY}	Read deassertion delay after RDY is asserted	15		ns
t_{DCSD}	Chip select deassertion delay after read is deasserted		10	
t_{AH}	Address hold time with respect to nOE		5	

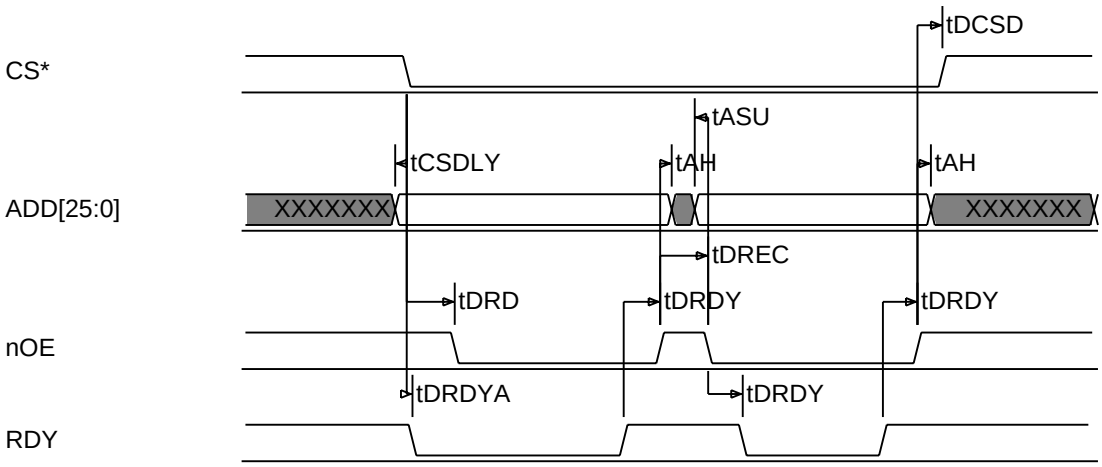


Figure 6-16: Intel StrongARM 1110 Processor Burst Read Timing

Table 6-17: Intel StrongARM 11xx Processor Write Timing

Symbol	Parameter	Min	Max	Units
t_{ASU}	Address setup time with respect to CS*	-	5	ns
t_{DWE}	Write enable signal delay from CS* assertion	-	20	ns
t_{DRDYA}	RDY signal is asserted based on CS* and address decode	-	2	ns
t_{DATS}	Write data setup time with respect to write signal	-	5	ns
t_{DCS}	Chip select deassertion delay after write signal is deasserted	-	10	ns
t_{AH}	Address hold time with respect to write signal		5	
t_{DRDY}	Write enable signal deassertion delay after RDY is asserted		15	
t_{DATH}	Data hold time with respect to write enable		5	

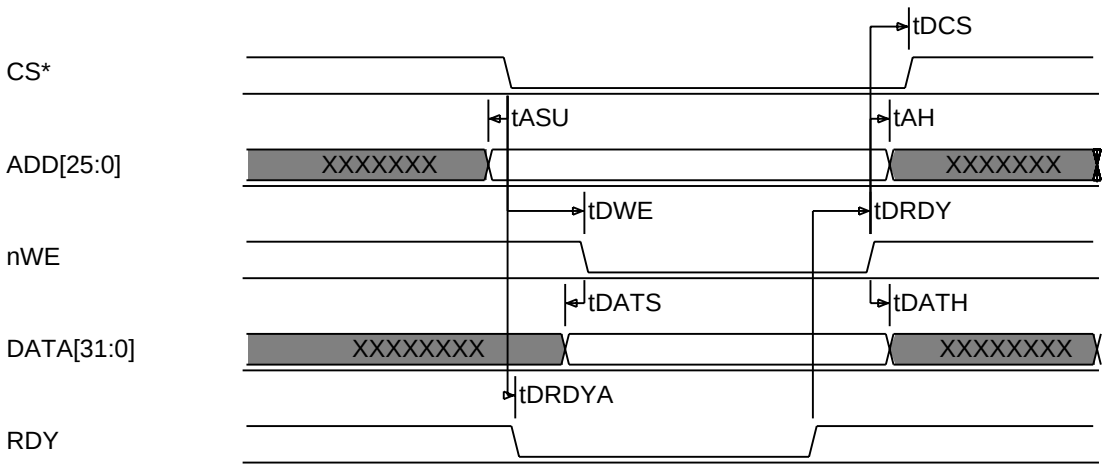


Figure 6-18: Intel StrongARM 11xx Processor Write Timing

Table 6-19: Intel StrongARM 11xx Processor Burst Write Timing

Symbol	Parameter	Min	Max	Units
t_{CSDLY}	Chip select assertion delay after address is valid	-	5	ns
t_{DWE}	Write enable signal delay from CS* assertion	-	20	ns
t_{DATS}	Write data setup time with respect to write signal	-	5	ns
t_{DRDYA}	RDY signal is asserted based on CS* and address decode	-	15	ns
t_{ASU}	Address setup time with respect to CS*		5	
t_{DREC}	Write recovery time		20	
t_{DATH}	Data hold time with respect to write enable signal		5	
t_{DCSD}	Chip select deassertion delay after read is deasserted		10	
t_{AH}	Address hold time with respect to write signal		5	

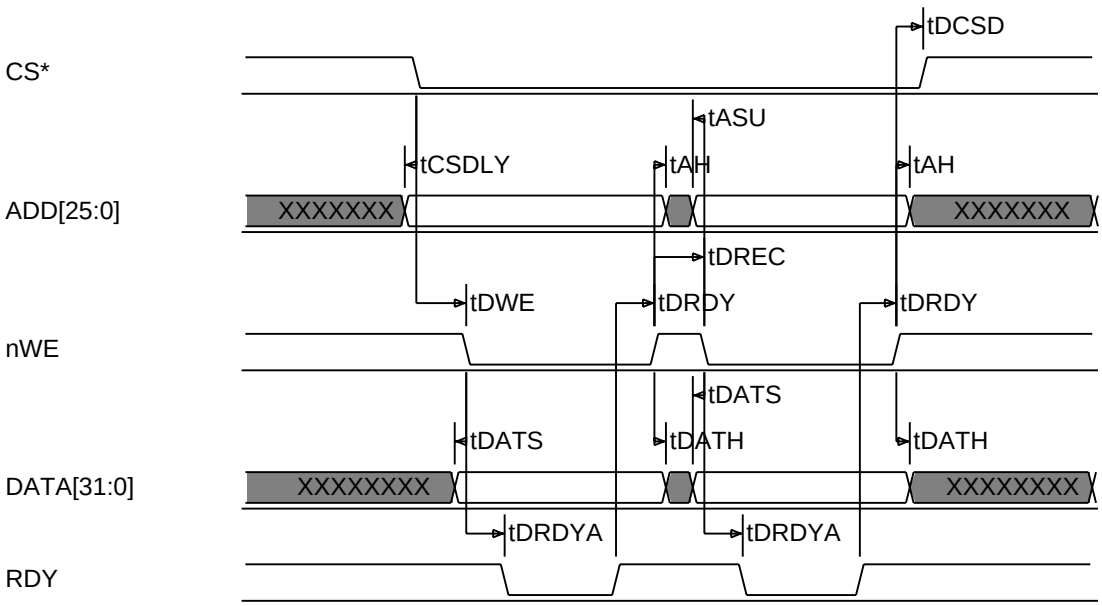


Figure 6-20: Intel StrongARM 11xx Processor Burst Write Timing

Table 6-21: NEC VR4121 Processor Read Timing at 83MHz

Symbol	Parameter	Min	Max	Units
t_{A2C}	Address to chip select delay	-	1	ns
t_{AS}	Read assertion delay from CS* assertion	-	100	ns
t_{MRDA}	RDY signal assertion delay based on CS* and address decode	-	2	ns
t_{DCSD}	Chip select deassertion delay from RD# deassertion		50	
t_{RHCH}	Read signal deassertion from RDY assertion		70	

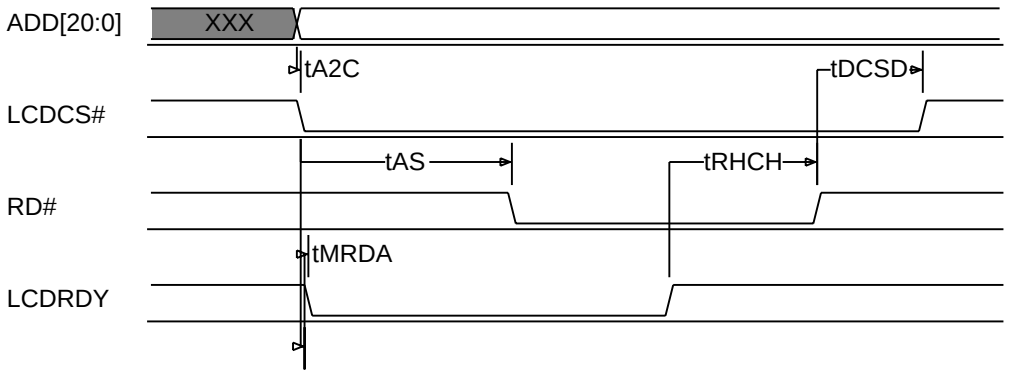


Figure 6-22: NEC VR4121 Processor Read Timing at 83MHz

Table 6-23: v141xwr83 NEC VR41xx Processor Write Timing at 83MHz

Symbol	Parameter	Min	Max	Units
t_{DBESU}	Byte enables setup time with respect to write enable	-	17.35	ns
t_{A2C}	Address to chip select delay	-	1	ns
t_{AS}	Write assertion delay from CS* deassertion		100	
t_{MRDA}	RDY assertion delay based on CS* and address decode	-	2	ns
t_{DATSU}	Data setup time with respeect to write enable		17.35	
t_{RHCH}	Write signal deassertion delay after RDY# is asserted		71.4	
t_{DCSD}	Chip select deassertion delay after write signal is deasserted		50	

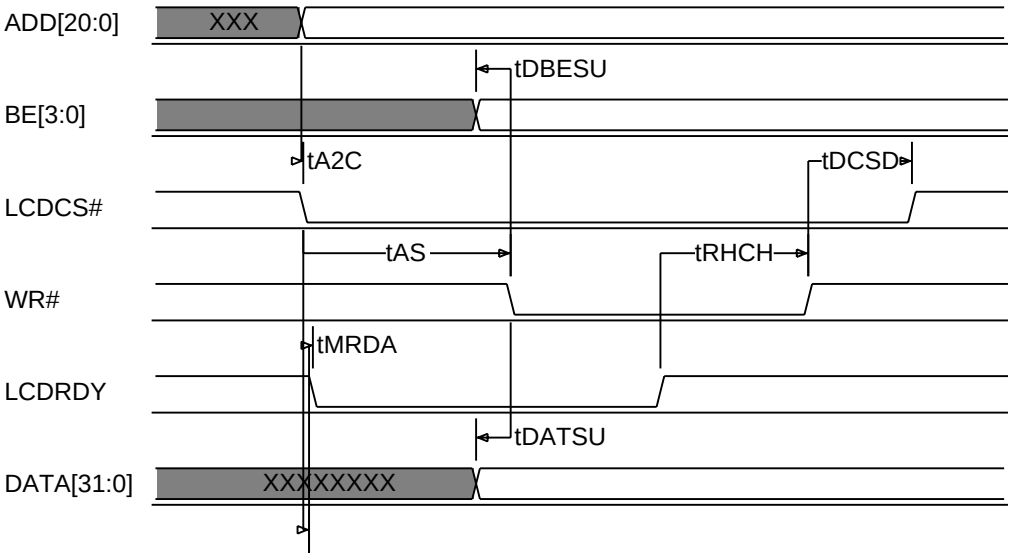


Figure 6-24: v141xwr83 NEC VR41xx Processor Write Timing at 83MHz

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

Title: MQ-200 Data Book
Publication #: MQ-200-DB-1001-A
Revision No.: 0.03
Date: 05/07/1999