

# **MB86277 <MINT>**

## **Graphics Controller Specifications**

Revision 1.0b

24th Jan, 2007



Copyright © FUJITSU LIMITED 2004-2006

ALL RIGHTS RESERVED

## <Notes>

- The specifications in this manual are subject to change without notice. Contact our Sales Department before purchasing the product described in this manual.
- Information and circuit diagrams in this manual are only examples of device applications, they are not intended to be used in actual equipment. Also, Fujitsu accepts no responsibility for infringement of patents or other rights owned by third parties caused by use of the information and circuit diagrams.
- The contents of this manual must not be reprinted or duplicated without permission of Fujitsu.
- Fujitsu's semiconductor devices are intended for standard uses (such as office equipment (computers and OA equipment), industrial/communications/measuring equipment, and personal/home equipment). Customers using semiconductor devices for special applications (including aerospace, nuclear, military and medical applications) in which a failure or malfunction might endanger life or limb and which require extremely high reliability must contact our Sales Department first. If damage is caused by such use of our semiconductor devices without first consulting our Sales Department, Fujitsu will not assume any responsibility for the loss.
- Semiconductor devices fail with a known probability. Customers must use safety design (such as redundant design, fireproof design, over-current prevention design, and malfunction prevention design) so that failures will not cause accidents, injury or death).
- If the products described in this manual fall within the goods or technologies regulated by the Foreign Exchange and Foreign Trade Law, permission must be obtained before exporting the goods or technologies.



Burns      There is a danger of burns because the IC surface is heated depending on the IC operating conditions. In this case, take safety measures.

## Update history

[illegible]

## CONTENTS

<b>1</b>	<b>GENERAL</b>	<b>11</b>
1.1	FEATURES	11
1.2	BLOCK DIAGRAM	12
1.3	FUNCTIONAL OVERVIEW	13
1.3.1	<i>Host CPU interface</i>	13
1.3.2	<i>External memory interface</i>	14
1.3.3	<i>Display controller</i>	15
1.3.4	<i>Video Capture</i>	17
1.3.5	<i>2D Drawing</i>	18
1.3.6	<i>Special effects</i>	20
1.3.7	<i>Others</i>	23
1.3.8	<i>I2C interface</i>	24
<b>2</b>	<b>PINS</b>	<b>25</b>
2.1	SIGNALS	25
2.1.1	<i>Signal lines</i>	25
2.2	PIN ASSIGNMENT	26
2.2.1	<i>LQFP256 Pin assignment diagram</i>	26
2.2.2	<i>LQFP256 Pin assignment table</i>	27
2.2.3	<i>Pin treatment table</i>	29
2.3	PIN FUNCTION	31
2.3.1	<i>Host CPU interface</i>	31
2.3.2	<i>Video output interface</i>	33
2.3.3	<i>Video Capture interface</i>	35
2.3.4	<i>Graphics memory interface</i>	36
2.3.5	<i>Clock input</i>	37
2.3.6	<i>Test pins</i>	38
2.3.7	<i>Reset sequence</i>	38
<b>3</b>	<b>PROCEDURE OF THE HARDWARE INITIALIZATION</b>	<b>39</b>
3.1	HARDWARE RESET	39
3.2	RE-RESET	39
3.3	SOFTWARE RESET	39
<b>4</b>	<b>HOST INTERFACE</b>	<b>40</b>
4.1	OPERATION MODE	40
4.1.1	<i>Host CPU mode</i>	40
4.1.2	<i>Ready signal mode</i>	40

4.1.3	<i>BS signal mode</i> .....	41
4.1.4	<i>Endian</i> .....	41
4.2	ACCESS MODE .....	42
4.2.1	<i>SRAM interface</i> .....	42
4.2.2	<i>FIFO interface (fixed transfer destination address)</i> .....	42
4.3	DMA TRANSFER .....	43
4.3.1	<i>Data transfer unit</i> .....	43
4.3.2	<i>Address mode</i> .....	43
4.3.3	<i>Bus mode</i> .....	44
4.3.4	<i>DMA transfer request</i> .....	44
4.3.5	<i>Ending DMA transfer</i> .....	44
4.4	TRANSFER OF LOCAL DISPLAY LIST .....	45
4.5	INTERRUPT .....	46
4.6	SH3 MODE .....	46
4.7	WAIT .....	46
4.8	MEMORY MAP .....	47
<b>5</b>	<b>GRAPHICS MEMORY</b> .....	<b>49</b>
5.1	CONFIGURATION .....	49
5.1.1	<i>Data type</i> .....	49
5.1.2	<i>Memory Mapping</i> .....	50
5.1.3	<i>Data Format</i> .....	50
5.2	FRAME MANAGEMENT .....	52
5.2.1	<i>Single Buffer</i> .....	52
5.2.2	<i>Double Buffer</i> .....	52
5.3	MEMORY ACCESS .....	52
5.3.1	<i>Memory Access by host CPU</i> .....	52
5.3.2	<i>Priority of memory accessing</i> .....	52
5.4	CONNECTION WITH MEMORY .....	53
5.4.1	<i>Connection with memory</i> .....	53
<b>6</b>	<b>DISPLAY CONTROLLER</b> .....	<b>54</b>
6.1	OVERVIEW .....	54
6.2	DISPLAY FUNCTION .....	54
6.2.1	<i>Layer configuration</i> .....	54
6.2.2	<i>Overlay</i> .....	56
6.2.3	<i>Display parameters</i> .....	58
6.2.4	<i>Display position control</i> .....	59
6.3	DISPLAY COLOR .....	61
6.4	CURSOR .....	62
6.4.1	<i>Cursor display function</i> .....	62

6.4.2	<i>Cursor control</i> .....	62
6.5	DISPLAY SCAN CONTROL .....	63
6.5.1	<i>Applicable display</i> .....	63
6.5.2	<i>Interlace display</i> .....	64
6.6	THE EXTERNAL SYNCHRONOUS SIGNAL .....	65
6.7	VIDEO INTERFACE, NTSC/PAL OUTPUT .....	68
<b>7</b>	<b>VIDEO CAPTURE</b> .....	<b>69</b>
7.1	FORMAT .....	69
7.1.1	<i>Input Data Format</i> .....	69
7.1.2	<i>Video Signal Capture</i> .....	69
7.1.3	<i>Non-interlace Transformation</i> .....	69
7.2	VIDEO BUFFER .....	70
7.2.1	<i>Data Format</i> .....	70
7.2.2	<i>Synchronous Control</i> .....	70
7.2.3	<i>Area Allocation</i> .....	70
7.2.4	<i>Window Display</i> .....	71
7.2.5	<i>Interlace Display</i> .....	71
7.3	SCALING .....	72
7.3.1	<i>Video Reduction Function</i> .....	72
7.3.2	<i>Vertical Interpolation</i> .....	72
7.4	ERROR HANDLING .....	72
7.4.1	<i>Error Detection Function</i> .....	72
<b>8</b>	<b>DRAWING PROCESSING</b> .....	<b>74</b>
8.1	COORDINATE SYSTEM.....	74
8.1.1	<i>Drawing coordinates</i> .....	74
8.1.2	<i>Texture coordinates</i> .....	75
8.1.3	<i>Frame buffer</i> .....	75
8.2	FIGURE DRAWING .....	76
8.2.1	<i>Drawing primitives</i> .....	76
8.2.2	<i>Polygon drawing function</i> .....	76
8.2.3	<i>Drawing parameters</i> .....	77
8.2.4	<i>Anti-aliasing function</i> .....	78
8.3	BIT MAP PROCESSING .....	79
8.3.1	<i>BLT</i> .....	79
8.3.2	<i>Pattern data format</i> .....	79
8.4	TEXTURE MAPPING .....	80
8.4.1	<i>Texture size</i> .....	80
8.4.2	<i>Texture color</i> .....	80
8.4.3	<i>Texture Wrapping</i> .....	80

8.4.4	<i>Filtering</i> .....	81
8.4.5	<i>8.4.5 Texture blending</i> .....	81
8.5	RENDERING .....	82
8.5.1	<i>Tiling</i> .....	82
8.5.2	<i>Alpha blending</i> .....	82
8.5.3	<i>Logic operation</i> .....	83
8.5.4	<i>Hidden plane management (Optional function)</i> .....	83
8.6	DRAWING ATTRIBUTES .....	84
8.6.1	<i>Line drawing attributes</i> .....	84
8.6.2	<i>Triangle drawing attributes</i> .....	84
8.6.3	<i>Texture attributes</i> .....	85
8.6.4	<i>BLT attributes</i> .....	85
8.6.5	<i>Character pattern drawing attributes</i> .....	85
<b>9</b>	<b>DISPLAY LIST</b> .....	<b>86</b>
9.1	OVERVIEW .....	86
9.1.1	<i>Header format</i> .....	87
9.2	RENDERING COMMAND .....	88
9.2.1	<i>Command list</i> .....	88
9.2.2	<i>Details of rendering commands</i> .....	92
<b>10</b>	<b>I2C FUNCTION</b> .....	<b>102</b>
10.1	START CONDITION .....	102
10.2	STOP CONDITION .....	102
10.3	ADDRESSING .....	103
10.4	SYNCHRONIZATION OF SCL .....	103
10.5	ARBITRATION .....	104
10.6	ACKNOWLEDGE .....	104
10.7	BUS ERROR .....	104
10.8	INITIALIZE .....	105
10.9	1-BYTE TRANSFER FROM MASTER TO SLAVE .....	106
10.10	1-BYTE TRANSFER FROM SLAVE TO MASTER .....	107
10.11	RECOVERY FROM BUS ERROR .....	108
10.12	NOTES .....	109
10.12.1	<i>About a 10-bit slave address</i> .....	109
10.12.2	<i>About competition of SCC, MSS, and INT bit</i> .....	109
10.12.3	<i>About setup of S serial transfer clock</i> .....	109
<b>11</b>	<b>REGISTER</b> .....	<b>110</b>
11.1	REGISTER LIST .....	110
11.1.1	<i>Host interface register list</i> .....	110

11.1.2	<i>Graphics memory interface register list</i> .....	112
11.1.3	<i>Display controller register list</i> .....	113
11.1.4	<i>Video Capture register list</i> .....	118
11.1.5	<i>Drawing engine register list</i> .....	119
11.1.6	<i>I2C register list</i> .....	125
11.2	EXPLANATION OF REGISTER .....	126
11.2.1	<i>Host interface registers</i> .....	127
11.2.2	<i>Graphics memory interface registers</i> .....	134
11.2.3	<i>Display control register</i> .....	137
11.2.4	<i>Video Capture Registers</i> .....	185
11.2.5	<i>Drawing control registers</i> .....	192
11.2.6	<i>Drawing mode registers</i> .....	195
11.2.7	<i>Triangle drawing registers</i> .....	210
11.2.8	<i>Line drawing registers</i> .....	213
11.2.9	<i>Pixel drawing registers</i> .....	214
11.2.10	<i>Rectangle drawing registers</i> .....	214
11.2.11	<i>Blt registers</i> .....	215
11.2.12	<i>2D line with XY setup drawing registers</i> .....	216
11.2.13	<i>2D triangle with XY setup drawing registers</i> .....	217
11.2.14	<i>Display list FIFO registers</i> .....	218
11.2.15	<i>I2C registers</i> .....	219
<b>12</b>	<b>TIMING DIAGRAM</b> .....	<b>225</b>
12.1	HOST INTERFACE .....	225
12.1.1	<i>CPU read/write timing diagram in SH3 mode (Normally Not Ready Mode)</i> .....	225
12.1.2	<i>CPU read/write timing diagram in SH3 mode (Normally Ready Mode)</i> .....	226
12.1.3	<i>CPU read/write timing diagram in SH4 mode (Normally Not Ready Mode)</i> .....	227
12.1.4	<i>CPU read/write timing diagram in SH4 mode (Normally Ready Mode)</i> .....	228
12.1.5	<i>CPU read/write timing diagram in V832 mode (Normally Not Ready Mode)</i> .....	229
12.1.6	<i>CPU read/write timing diagram in V832 mode (Normally Ready Mode)</i> .....	230
12.1.7	<i>CPU read/write timing diagram in SPARClite (Normally Not Ready Mode)</i> .....	231
12.1.8	<i>CPU read/write timing diagram in SPARClite (Normally Ready Mode)</i> .....	232
12.1.9	<i>SH4 single-address DMA write (transfer of 1 long word)</i> .....	233
12.1.10	<i>SH4 single-address DMA write (transfer of 8 long words)</i> .....	234
12.1.11	<i>SH3/4 dual-address DMA (transfer of 1 long word)</i> .....	235
12.1.12	<i>SH3/4 dual-address DMA (transfer of 8 long words)</i> .....	235
12.1.13	<i>V832 DMA transfer</i> .....	236
12.1.14	<i>SH4 single-address DMA transfer end timing</i> .....	237
12.1.15	<i>SH3/4 dual-address DMA transfer end timing</i> .....	237
12.1.16	<i>V832 DMA transfer end timing</i> .....	238



12.1.17	<i>SH4 dual DMA write without ACK</i> .....	239
12.1.18	<i>Dual-address DMA (without ACK) end timing</i> .....	240
12.2	GRAPHICS MEMORY INTERFACE .....	241
12.2.1	<i>Timing of read access to same row address</i> .....	241
12.2.2	<i>Timing of read access to different row addresses</i> .....	242
12.2.3	<i>Timing of write access to same row address</i> .....	243
12.2.4	<i>Timing of write access to different row addresses</i> .....	244
12.2.5	<i>Timing of read/write access to same row address</i> .....	245
12.2.6	<i>Delay between ACTV commands</i> .....	246
12.2.7	<i>Delay between Refresh command and next ACTV command</i> .....	246
12.3	DISPLAY TIMING .....	247
12.3.1	<i>Non-interlace mode</i> .....	247
12.3.2	<i>Interlace video mode</i> .....	248
12.3.3	<i>Composite synchronous signal</i> .....	249
12.4	CPU CAUTIONS .....	249
12.5	SH3 MODE .....	250
12.6	SH4 MODE .....	250
12.7	V832 MODE .....	251
12.8	SPARCLITE .....	251
12.9	SUPPORTED DMA TRANSFER MODES .....	251
<b>13</b>	<b>ELECTRICAL CHARACTERISTICS</b> .....	<b>252</b>
13.1	INTRODUCTION .....	252
13.2	MAXIMUM RATING .....	252
13.3	RECOMMENDED OPERATING CONDITIONS .....	253
13.3.1	<i>Recommended operating conditions</i> .....	253
13.3.2	<i>Note at power-on</i> .....	254
13.4	DC CHARACTERISTICS .....	255
13.4.1	<i>DC Characteristics</i> .....	255
13.4.2	<i>V-I characteristics diagram</i> .....	256
13.5	AC CHARACTERISTICS .....	257
13.5.1	<i>Host interface</i> .....	257
13.5.2	<i>Video interface</i> .....	258
13.5.3	<i>Video Capture Interface</i> .....	259
13.5.4	<i>Graphics memory interface</i> .....	260
13.5.5	<i>PLL specifications</i> .....	264
13.5.6	<i>I2C Interface</i> .....	265
13.6	AC CHARACTERISTICS MEASURING CONDITIONS .....	267
13.7	TIMING DIAGRAM .....	268
13.7.1	<i>Host interface</i> .....	268

<i>13.7.2</i>	<i>Video interface .....</i>	<i>272</i>
<i>13.7.3</i>	<i>Video Capture Interface .....</i>	<i>273</i>
<i>13.7.4</i>	<i>Graphics memory interface .....</i>	<i>274</i>
<i>13.7.5</i>	<i>I2C interface .....</i>	<i>276</i>

# 1 GENERAL

## 1.1 Features

CMOS 0.18μm technology

Internal and memory frequency : 100MHz (generated by on-chip PLL)

Base-clock for display clocks : 400.9MHz (generated by on-chip PLL)

Display resolutions typically from 320x234 up to 1024x768

6 layers of overlay display (windows)

Alpha Plane and constant alpha value for each layer

Digital Video input (various formats including YUV)

Video Scaler (down scaling)

RGB digital output (8bit x 3)

RGB Analog Output

Built-in alpha blending, anti-aliasing and chroma-keying

Rendering Engine for various kinds of 2D graphic acceleration functions

Texture Mapping Unit for 2D polygon support up to 4096x4096 textures

Bit-Blt Unit for transfers up to 4096x4096 areas

Alpha Bit-Blt and ROP2 functions

External 32-bit SDRAM interface for up to 64MB graphic memory

Parallel host interface (FR,SH3,SH4,V850,SparcLite etc)

Internal and external DMA support

I2C Master interface

Supply voltage 3.3V (I/O), 1.8V (Internal)

LQFP-256 Package

Typical power consumption < 1.0W (estimated)

Temperature range -40..+85 °C

## 1.2 Block Diagram

MINT general block diagram is shown below:

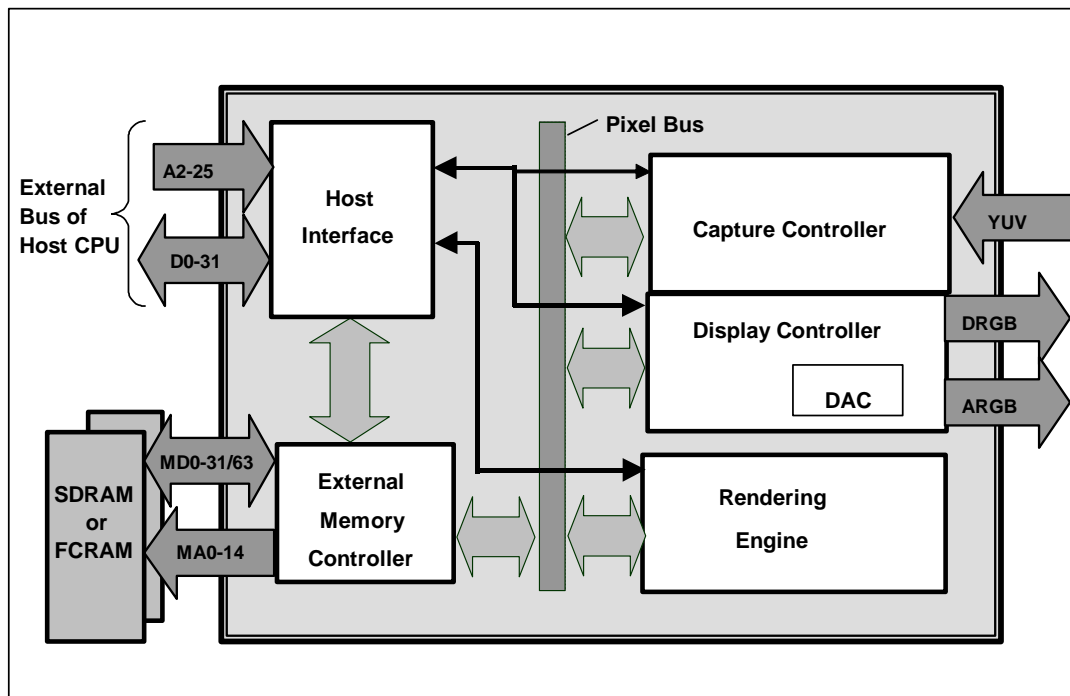


Fig.1.1 MINT Block Diagram

## 1.3 Functional Overview

### 1.3.1 Host CPU interface

#### Supported CPU

MINT can be connected to SH3 and SH4 manufactured by HITACHI, V832 by NEC, SPARClite (MB86833) by Fujitsu.

#### External Bus Clock

Can be connected at max. 100 MHz (when using SH4 interface)

#### Ready Mode

Supports normal ready/not ready.

#### Endian

Supports little endian.

#### Access Mode

SRAM interface

FIFO interface (transfer destination address fixed)

#### DMA transfer

Supports 1-double word (32 bits) /8-double word (32 bytes) (only SH4) for transfer unit.

ACK used/unused mode can be selected as protocol (only for DAM in dual address mode)

Supports dual address/mode single address mode (only SH4).

Supports cycle steal/burst.

Supports local display list transfer.

#### Interrupt

Vertical (frame) synchronous detection

Field synchronous detection

External synchronous error detection

Drawing command error

Drawing command execution end

### 1.3.2 External memory interface

SDRAM or FCRAM can be connected.

64 bits or 32 bits can be selected for data bus.

Max. 100MHz is available for operating frequency.

Connectable memory configuration is as shown below.

**External Memory Configuration**

Type	Data bus width	Use count	Total capacity
FCRAM 16 Mbits (x16 Bits)	32 Bits	2	4 Mbytes
FCRAM 16 Mbits (x16 Bits)	64 Bits	4	8 Mbytes
SDRAM 64 Mbits (x32 Bits)	32 Bits	1	8 Mbytes
SDRAM 64 Mbits (x32 Bits)	64 Bits	2	16 Mbytes
SDRAM 64 Mbits (x16 Bits)	32 Bits	2	16 Mbytes
SDRAM 64 Mbits (x16 Bits)	64 Bits	4	32 Mbytes
SDRAM 128 Mbits (x32 Bits)	32 Bits	1	16 Mbytes
SDRAM 128 Mbits (x32 Bits)	64 Bits	2	32 Mbytes
SDRAM 128 Mbits (x16 Bits)	32 Bits	2	32 Mbytes
SDRAM 128 Mbits (x16 Bits)	64 Bits	4	64 Mbytes
SDRAM 256 Mbits (x16 Bits)	32 Bits	2	64 Mbytes

### 1.3.3 Display controller

#### Video data output

Analog RGB video output is provided. And setting graphics memory bus to 32 bits, digital RGB video output is also provided.

#### Screen resolution

LCD panels with wide range of resolutions are supported by using a programmable timing generator as follows:

Screen Resolutions	
	Resolutions
	1024 × 768
	1024 × 600
	800 × 600
	854 × 480
	640 × 480
	480 × 234
	400 × 234
	320 × 234

#### Hardware cursor

MINT supports two hardware cursor functions. Each of these hardware cursors is specified as a 64 × 64-pixel area. Each pixel of these hardware cursors is 8 bits and uses the same look-up table as indirect color mode.

#### Double buffer method

Double buffer method in which drawing window and display window is switched in units of 1 frame enables the smooth animation.

Flipping (switching of display window area) is performed in synchronization with the vertical blanking period using program.

#### Scroll method

Independent setting of drawing and display windows and their starting position enables the smooth scrolling.

#### Display colors

- Supports indirect color mode which uses the look-up table (color palette) in 8 bits/pixels.
- Entry for look-up table (color palette) corresponds to color code for 8 bits, in other words, 256. Color data is each 6 bits of RGB. Consequently, 256 colors can be displayed out of 260,000 colors.
- Supports direct color mode which specifies RGB with 16 bits/pixels.

## Overlay

### Compatibility mode

Up to four extra layers (C, W, M and B) can be displayed overlaid.

The overlay position for the hardware cursors is above/below the top layer (C).

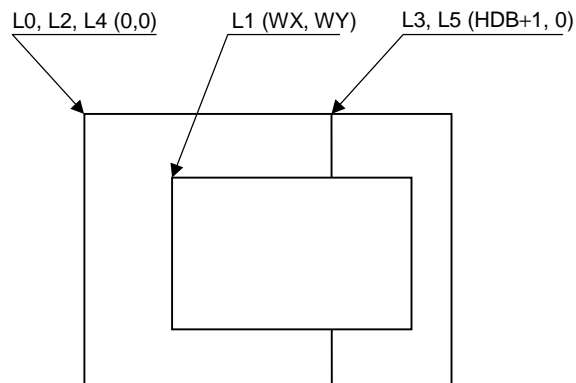
The transparent mode or the blend mode can be selected for overlay.

The M- and B-layers can be split into separate windows.

Window display can be performed for the W-layer.

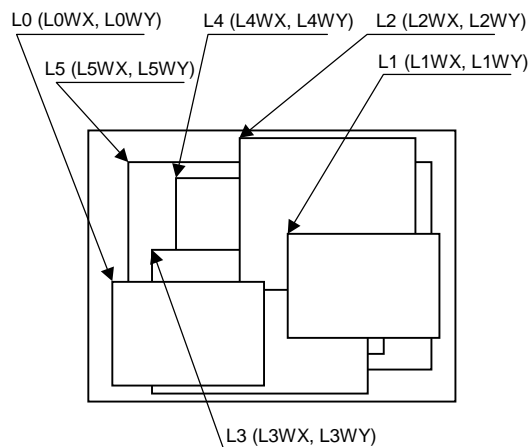
Two palettes are provided: C-layer and M-/B-layer.

The W-layer is used as the video input layer.



### Window mode

- Up to six screens (L0 to 5) can be displayed overlaid.
- The overlay sequence of the L0- to L5-layers can be changed arbitrarily.
- The overlay position for the hardware cursors is above/below the L0-layer.
- The transparent mode or the blend mode can be selected for overlay.
- The L5-layer can be used as the blend coefficient plane (8 bits/pixel).
- Window display can be performed for all layers.
- Four palettes corresponded to L0 to 3 are provided.
- The L1-layer is used as the video input layer.
- Background color display is supported in window display for all layers.





### **1.3.4 Video Capture**

The video capture function captures ITU RBT-656 format videos. Video data is stored in graphics memory once and then displayed on the screen in synchronization with the display scan.

Both NTSC and PAL video formats are supported.

### 1.3.5 2D Drawing

#### 2D Primitives

MINT can perform 2D drawing for graphics memory (drawing plane) in direct color mode or indirect color mode.

Bold lines with width and broken lines can be drawn. With anti-aliasing smooth diagonal lines also can be drawn.

A triangle can be tiled in a single color or 2D pattern (tiling), or mapped with a texture pattern by specifying coordinates of the 2D pattern at each vertex (texture mapping). At texture mapping, drawing/non-drawing can be set in pixel units. Moreover, transparent processing can be performed using alpha blending. When drawing in single color or tiling without Gouraud shading or texture mapping, high-speed 2DLine and high-speed 2DTriangle can be used. Only vertex coordinates are set for these primitives. High-speed 2DTriangle is also used to draw polygons.

#### 2D Primitives

Primitive type	Description
Point	Plots point
Line	Draws line
Bold line strip (provisional name)	Draws continuous bold line This primitive is used when interpolating the bold line joint.
Triangle	Draws triangle
High-speed 2DLine	Draws lines Compared to line, this reduces the host CPU processing load.
Arbitrary polygon	Draws arbitrary closed polygon containing concave shapes consisting of vertices

#### Arbitrary polygon drawing

Using this function, arbitrary closed polygon containing concave shapes consisting of vertices can be drawn. (There is no restriction on the count of vertices, however, the polygon with its sides crossed are not supported.) In this case, as a work area for drawing, polygon drawing flag buffer is used on the graphics memory. In drawing polygon, draw triangle for polygon drawing flag buffer using high-speed 2DTriangle. Decide any vertex as a starting point to draw triangle along the periphery. It enables you to draw final polygon form in single color or with tiling/texture mapping in a drawing frame.

### BLT/Rectangle drawing

This function draws a rectangle using logic operations. It is used to draw pattern and copy the image pattern within the drawing frame. It is also used for clearing drawing frame and Z buffer.

**BLT Attributes**

Attribute	Description
Raster operation	Selects two source logical operation mode
Transparent processing	Performs BLT without drawing pixel consistent with the transparent color.
Alpha blending	The alpha map and source in the memory is subjected to alpha blending and then copied to the destination.

### Pattern (Text) drawing

This function draws a binary pattern (text) in a specified color.

**Pattern (Text) Drawing Attributes**

Attribute	Description
Enlarge	Vertically $2 \times 2$ Horizontally $\times 2$ Vertically and Horizontally $\times 2$
Shrink	Vertically $1/2 \times 1/2$ Horizontally $1/2$ Vertically and Horizontally $1/2$

### Drawing clipping

This function sets a rectangle frame in drawing frame to prohibit the drawing of the outside the frame.

### 1.3.6 Special effects

#### Anti-aliasing

Anti-aliasing manipulates line borders of polygons in sub-pixel units and blend the pre-drawing pixel color with color to make the jaggies be seen smooth. It is used as a functional option for 2D drawing (in direct color mode only).

#### Bold line and broken line drawing

This function draws lines of a specific width and a broken line.

**Line Drawing Attributes**

Attribute	Description
Line width	Selectable from 1 to 32 pixels
Broken line	Set by 32 bit or 24 bit of broken line pattern

- Not support the Anti-aliasing of dashed line patterns.

## Alpha blending

Alpha blending blends two image colors to provide a transparent effect. MINT supports two types of blending; blending two different colors at drawing, and blending overlay planes at display. Transparent color is not used for these blending options.

There are two ways of specifying alpha blending for drawing:

- (1) Set a transparent coefficient to the register; the transparent coefficient is applied for transparency processing of one plane.
- (2) Set a transparent coefficient for each vertex of the plane; as with Gouraud shading, the transparent coefficient is linear-interpolated to perform transparent processing in pixel units.

In addition to the above, the following settings can be performed at texture mapping. When the most significant bit of each texture cell is 1, drawing or transparency can be set. When the most significant bit of each texture cell is 0, non-drawing can be set.

### Alpha Blending

Type	Description
Drawing	Transparent ratio set in particular register  While one primitive (polygon, pattern, etc.), being drawn, registered transparent ratio applied  A transparent coefficient set for each vertex. A linear-interpolated transparent coefficient applied.  This is possible only in direct color mode.
Overlay display	Blends top layer pixel color with lower layer pixel color  Transparent coefficient set in particular register  Registered transparent coefficient applied during one frame scan

## Gouraud Shading

Gouraud shading can be used in the direct color mode to provide 3D object real shading and color gradation.

## Texture mapping

MINT supports texture mapping to map an image pattern onto the surface of plane. The texture pattern can be laid out in the graphics memory. In this case, max.  $4096 \times 4096$  pixels can be used.

For drawing 8-bit color, only point sampling can be specified for texture interpolation; only de-curl can be specified for the blend mode.

**Texture Mapping**

Function	Description
Filtering	Point sample Bi-linear filter
Coordinates correction	Linear Perspective
Blend	De-curl Modulate Stencil
Alpha blend	Normal Stencil Stencil alpha
Wrap	Repeat Cramp Border

### **1.3.7 Others**

#### **Drawing color**

8-bit indirect color and 16-bit direct color are supported as a drawing input data.

### **1.3.8 I2C interface**

- Master transmission and receipt
- Arbitration
- Clock synchronization
- Detection of slave address
- Detection of general call address
- Detection of transfer direction
- Repeated generation and detection of START condition
- Detection of bus error
- Correspondence to standard-mode (100kbit/s ) / high-speed-mode (400kbit/s)



## 2 PINS

### 2.1 Signals

#### 2.1.1 Signal lines

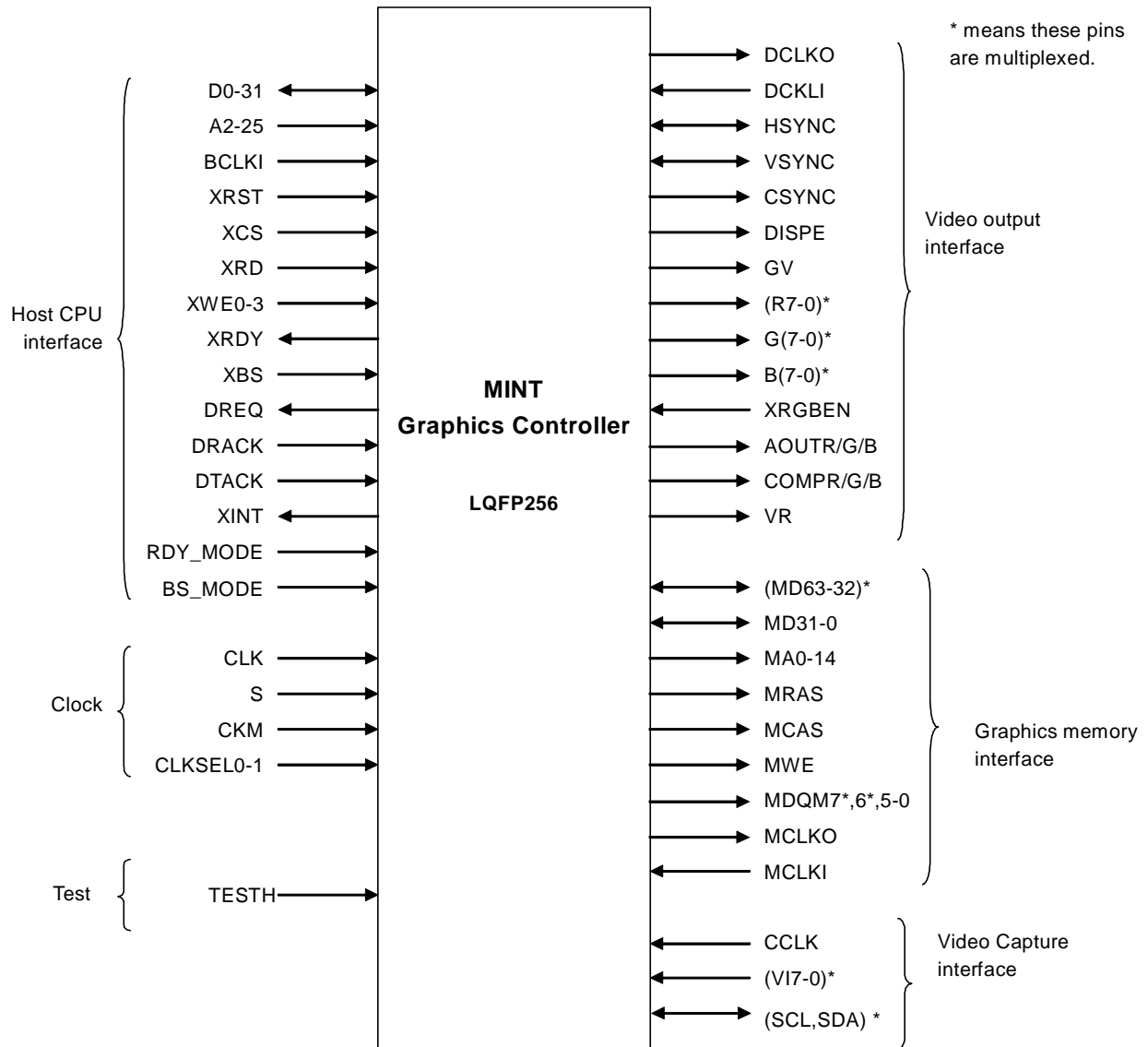


Fig. 2.1 MINT Signal Lines

## 2.2 Pin Assignment

### 2.2.1 LQFP256 Pin assignment diagram

		MODE0	256			192	MD33 / B3
		RDY_MODE1	255			191	MD32 / B2
		BS_MODE	254			190	VDDH
		XRGBEN	253			189	VSS
		COMPR	252			188	DQM7 / B1
		VREF	251			187	DQM6 / B0
		AVS0	250			186	DQM5
		VRO	249			185	DQM4
		AVD0	248			184	MWE
		AOUTR	247			183	MCAS
		OPEN	246			182	VSS
		AVS1	245			181	VDDL
		COMPG	244			180	MRAS
		AVD1	243			179	MA14
		AOUTG	242			178	MA13
		AVS2	241			177	MA12
		COMPB	240			176	MA11
		AVD2	239			175	MA10
		AOUTB	238			174	MA9
		TESTH	237			173	VDDH
		TESTH	236			172	MA8
		TESTH	235			171	VDDL
		TESTH	234			170	MA7
		TESTH	233			169	MA6
		VSS	232			168	VSS
		CCLK	231			167	MA5
		VDL	230			166	MA4
		MD63 / V17	229			165	MA3
		MD62 / V16	228			164	MA2
		MD61 / V15	227			163	MA1
		MD60 / V14	226			162	MA0
		MD59 / V13	225			161	VDDL
		VDL	224			160	DQM3
		MD58 / V12	223			159	DQM2
		MD57 / V11	222			158	DQM1
		MD56 / V10	221			157	DQM0
		MD55 / V9	220			156	MCLKO
		MD54 / SDA	219			155	VDDH
		VDDH	218			154	VSS
		VSS	217			153	VSS
		MD53 / R7	216			152	MCLKI
		MD52 / R6	215			151	VDDL
		MD51 / R5	214			150	VSS
		VDL	213			149	MD31
		MD50 / R4	212			148	MD30
		MD49 / R3	211			147	MD29
		MD48 / R2	210			146	MD28
		MD47 / R1	209			145	MD27
		MD46 / R0	208			144	MD26
		MD45 / G7	207			143	MD25
		MD44 / G6	206			142	MD24
		VDDH	205			141	VDDL
		MD43 / G5	204			140	MD23
		VSS	203			139	VSS
		MD42 / G4	202			138	MD22
		VDL	201			137	MD21
		MD41 / G3	200			136	MD20
		MD40 / G2	199			135	MD19
		MD39 / G1	198			134	MD18
		MD38 / G0	197			133	MD17
		MD37 / B7	196			132	MD16
		MD36 / B6	195			131	VDDH
		MD35 / B5	194			130	MD15
		MD34 / B4	193			129	MD14
MODE1	1					128	
MODE2	2					127	
DCLKI	3					126	
VDDH	4					125	
VSYN	5					124	
HSYN	6					123	
DISPE	7					122	
GV	8					121	
CSYN	9					120	
DCLKO	10					119	
VSS	11					118	
VDDL	12					117	
XRDY	13					116	
XINT	14					115	
DREQ	15					114	
VDDH	16					113	
VSS	17					112	
BCLKI	18					111	
XCS	19					110	
XRD	20					109	
XBS	21					108	
VDDL	22					107	
D0	23					106	
D1	24					105	
VSS	25					104	
D2	26					103	
D3	27					102	
D4	28					101	
D5	29					100	
D6	30					99	
D7	31					98	
VDDL	32					97	
VSS	33					96	
D8	34					95	
D9	35					94	
VDDH	36					93	
D10	37					92	
D11	38					91	
D12	39					90	
VSS	40					89	
D13	41					88	
D14	42					87	
VDDL	43					86	
D15	44					85	
D16	45					84	
D17	46					83	
D18	47					82	
D19	48					81	
D20	49					80	
D21	50					79	
VDDH	51					78	
D22	52					77	
D23	53					76	
VSS	54					75	
VDDL	55					74	
D24	56					73	
D25	57					72	
D26	58					71	
D27	59					70	
D28	60					69	
D29	61					68	
D30	62					67	
D31	63					66	
VSS	64					65	
		MODE0	256			192	
		RDY_MODE1	255			191	
		BS_MODE	254			190	
		XRGBEN	253			189	
		COMPR	252			188	
		VREF	251			187	
		AVS0	250			186	
		VRO	249			185	
		AVD0	248			184	
		AOUTR	247			183	
		OPEN	246			182	
		AVS1	245			181	
		COMPG	244			180	
		AVD1	243			179	
		AOUTG	242			178	
		AVS2	241			177	
		COMPB	240			176	
		AVD2	239			175	
		AOUTB	238			174	
		TESTH	237			173	
		TESTH	236			172	
		TESTH	235			171	
		TESTH	234			170	
		TESTH	233			169	
		VSS	232			168	
		CCLK	231			167	
		VDL	230			166	
		MD63 / V17	229			165	
		MD62 / V16	228			164	
		MD61 / V15	227			163	
		MD60 / V14	226			162	
		MD59 / V13	225			161	
		VDL	224			160	
		MD58 / V12	223			159	
		MD57 / V11	222			158	
		MD56 / V10	221			157	
		MD55 / V9	220			156	
		MD54 / SDA	219			155	
		VDDH	218			154	
		VSS	217			153	
		MD53 / R7	216			152	
		MD52 / R6	215			151	
		MD51 / R5	214			150	
		VDL	213			149	
		MD50 / R4	212			148	
		MD49 / R3	211			147	
		MD48 / R2	210			146	
		MD47 / R1	209			145	
		MD46 / R0	208			144	
		MD45 / G7	207			143	
		MD44 / G6	206			142	
		VDDH	205			141	
		MD43 / G5	204			140	
		VSS	203			139	
		MD42 / G4	202			138	
		VDL	201			137	
		MD41 / G3	200			136	
		MD40 / G2	199			135	
		MD39 / G1	198			134	
		MD38 / G0	197			133	
		MD37 / B7	196			132	
		MD36 / B6	195			131	
		MD35 / B5	194			130	
		MD34 / B4	193			129	
						128	
						127	
						126	
						125	
						124	
						123	
						122	
						121	
						120	
						119	
						118	
						117	
						116	
						115	
						114	
						113	
						112	
						111	
						110	
						109	
						108	
						107	
						106	
						105	
						104	
						103	
						102	
						101	
						100	
						99	
						98	
						97	
						96	
						95	
						94	
						93	
						92	
						91	
						90	
						89	
						88	
						87	
						86	
						85	
						84	
						83	
						82	
						81	
						80	
						79	
						78	
						77	
						76	
						75	
						74	
						73	
						72	
						71	
						70	

## 2.2.2 LQFP256 Pin assignment table

Pin No	Pin Name	Pin No	Pin Name	Pin No	Pin Name	Pin No	Pin Name
1	MODE1	65	XWE0	129	MD14	193	MD34 / B4
2	MODE2	66	XWE1	130	MD15	194	MD35 / B5
3	DCLKI	67	XWE2	131	VDDH	195	MD36 / B6
4	VDDH	68	XWE3	132	MD16	196	MD37 / B7
5	VSYN	69	DTACK	133	MD17	197	MD38 / G0
6	HSYN	70	DRACK	134	MD18	198	MD39 / G1
7	DISPE	71	A2	135	MD19	199	MD40 / G2
8	GV	72	A3	136	MD20	200	MD41 / G3
9	CSYN	73	A4	137	MD21	201	VDDL
10	DCLKO	74	A5	138	MD22	202	MD42 / G4
11	VSS	75	VSS	139	VSS	203	VSS
12	VDDL	76	VDDL	140	MD23	204	MD43 / G5
13	XRDY	77	A6	141	VDDL	205	VDDH
14	XINT	78	A7	142	MD24	206	MD44 / G6
15	DREQ	79	A8	143	MD25	207	MD45 / G7
16	VDDH	80	A9	144	MD26	208	MD46 / R0
17	VSS	81	A10	145	MD27	209	MD47 / R1
18	BCLKI	82	VDDH	146	MD28	210	MD48 / R2
19	XCS	83	A11	147	MD29	211	MD49 / R3
20	XRD	84	A12	148	MD30	212	MD50 / R4
21	XBS	85	A13	149	MD31	213	VDDL
22	VDDL	86	A14	150	VSS	214	MD51 / R5
23	D0	87	VDDL	151	VDDL	215	MD52 / R6
24	D1	88	A15	152	MCLKI	216	MD53 / R7
25	VSS	89	VSS	153	VSS	217	VSS
26	D2	90	A16	154	VSS	218	VDDH
27	D3	91	A17	155	VDDH	219	MD54/SDA
28	D4	92	A18	156	MCLKO	220	MD55/SCL
29	D5	93	A19	157	DQM0	221	MD56 / VI0
30	D6	94	A20	158	DQM1	222	MD57 / VI1
31	D7	95	A21	159	DQM2	223	MD58 / VI2
32	VDDL	96	A22	160	DQM3	224	VDDL
33	VSS	97	VDDL	161	VDDL	225	MD59 / VI3
34	D8	98	A23	162	MA0	226	MD60 / VI4
35	D9	99	A24	163	MA1	227	MD61 / VI5
36	VDDH	100	A25	164	MA2	228	MD62 / VI6
37	D10	101	CKM	165	MA3	229	MD63 / VI7
38	D11	102	XRST	166	MA4	230	VDDL
39	D12	103	PLLVSS	167	MA5	231	CCLK
40	VSS	104	VSS	168	VSS	232	VSS
41	D13	105	CLK	169	MA6	233	TESTH
42	D14	106	S	170	MA7	234	TESTH
43	VDDL	107	PLLVDD	171	VDDL	235	TESTH
44	D15	108	VDDL	172	MA8	236	TESTH
45	D16	109	VSS	173	VDDH	237	TESTH
46	D17	110	CLKSEL0	174	MA9	238	AOUTB
47	D18	111	CLKSEL1	175	MA10	239	AVD2
48	D19	112	VDDH	176	MA11	240	COMPB
49	D20	113	MD0	177	MA12	241	AVS2
50	D21	114	MD1	178	MA13	242	AOUTG
51	VDDH	115	MD2	179	MA14	243	AVD1
52	D22	116	MD3	180	MRAS	244	COMP
53	D23	117	VDDL	181	VDDL	245	AVS1
54	VSS	118	VSS	182	VSS	246	OPEN
55	VDDL	119	MD4	183	MCAS	247	AOUTR
56	D24	120	MD5	184	MWE	248	AVD0
57	D25	121	MD6	185	DQM4	249	VRO
58	D26	122	MD7	186	DQM5	250	AVS0
59	D27	123	MD8	187	DQM6 / B0	251	VREF
60	D28	124	MD9	188	DQM7 / B1	252	COMPR
61	D29	125	MD10	189	VSS	253	XRGBEN
62	D30	126	MD11	190	VDDH	254	BS_MODE
63	D31	127	MD12	191	MD32 / B2	255	RDY_MODE
64	VSS	128	MD13	192	MD33 / B3	256	MODE0

#### Notes

$V_{SS}/PLL V_{SS}$	:	Ground
$V_{DDH}$	:	3.3-V power supply
$V_{DDL}/PLL V_{DD}$	:	1.8-V power supply
$PLL V_{DD}$	:	PLL power supply (1.8 V)
OPEN	:	Do not connect anything.
TESTH	:	Input a 3.3 V-power supply.
AVS	:	Analog Ground
AVD	:	Analog power supply (3.3 V)

- It is recommended that  $PLL V_{DD}$  should be isolated on the PCB.
- It is recommended that AVD should be isolated on the PCB.
- Insert a bypass capacitor with good high frequency characteristics between the power supply and ground.

Place the capacitor as near as possible to the pin.

## 2.2.3 Pin treatment table

Table 1. Pin treatment table  
(Host interface, Video output interface)

	Pin Name	Direction *1	Default Treatment	Treatment of unused *2	Comment
Host Interface	MODE0-2	I	Connect to VDDH or GND according to the CPU mode.	<=	See "4.1 Operation Mode"
	RDY_MODE	I	Connect to VDDH or GND according to the Ready signal mode.	<=	See "4.1 Operation Mode"
	BS_MODE	I	Connect to VDDH or GND according to the BS signal mode.	<=	See "4.1 Operation Mode"
	D0-31	I/O	Connect to CPU data bus	<=	
	A2-A25	I	Connect to CPU address bus	<=	Connect A24 to XMWR in the V832 mode
	BCLKI	I	Connect to CPU bus clock	<=	Max 100MHz. Input the clock when power-on. See "12.2.2 Power on Precaution"
	XBS	I	Connect to CPU bus cycle start indicating signal	VDD when BS_MODE=VDD	This signal is 1 shot BCLKI pulse that indicates the bus cycle start. See "4.1 Operation Mode"
	XCS	I	Connect to chip select signal	<=	
	XRD	I	Connect to CPU read strobe signal	<=	
	XWE0-XWE4	I	Connect to CPU write byte enable signals	<=	Connect byte enable signal in V832 mode
	XRDY	0 (T)	Connect to CPU Ready (Wait) signal and Pull Up/Down according to RDY_MODE	<=	See "4.1 Operation Mode"
	DREQ	0	Connect to CPU DREQ signal	OPEN	SH3/4,V832=Low Active, See "4.3 DMA Transfer"
	DRACK/DMAAK	I	Connect to CPU DRACK signal	Connect to GND	Connect to DMAAK signal in V832 Mode, SH3/4,V832=High Active. See "4.3 DMA Transfer"
	DTACK/XTC	I	Connect to CPU DTACK signal	SH3/4=GND, V832=VDDH	Connect to XTC signal in V832 mode, SH3/4=High Active, V832=Low Active. See "4.3 DMA Transfer"
	XINT	0	Connect to CPU interrupt signal	OPEN	SH3/4=Low Active,V832=High Active
Video Output Interface	DCLKO	0	Connect to dot clock	<=	Selectable clock source, DCLKI or output of internal PLL. See DCM Register in "10.2.3 Display Controller Register"
	DCLKI	I	Connect to clock for dot clock	GND	
	HSYNC	I/O	Connect to HSYNC signal and Pull Up	<=	
	VSYSN	I/O	Connect to VSYSN signal and Pull Up	<=	
	CSYNC	0	Connect to CSYNC signal	OPEN	
	DISPE	0	Connect to display enable signal	OPEN	
	GV	0	Connect to select signal of analog video switch	OPEN	GDC's display=High Level
	XRGBEN	I	Connect to VDDH or GND according to the usage of upper bit of graphics memory	<=	See "2.3.2 Video Output Interface", "2.3.4 Graphics Memory Interface"
	AOUTR, G, B	Analog 0	terminate at 75 ohm	GND*4	
	VREF	Analog	Input 1.1V. A bypass capacitor (with good high-frequency characteristics) must be inserted between VREF and AVS.	GND*4	
	ACOMPR, G, B	Analog	Tied to analog AVD via 0.1uF ceramic capacitor	GND*4	
	VRO	Analog	Pull-down to analog ground by a 2.7K ohm resistor.	GND*4	
	R7-R0, G7-G0, B7-B0	0	Connect to video signals. <b>Available when XRGBEN=0 only.</b> Multiplexed MD53-MD32, MDQM7-MDQM6.	When XRGBEN=0, OPEN	See "2.3.2 Video Output Interface", "2.3.4 Graphics Memory Interface"

Table 2. Pin treatment table  
(Video capture interface, graphics memory interface, Clock/System)

	Pin Name	Direction *1	Default Treatment	Treatment of unused *2	Comment
Video Capture Interface	CCLK	I	Connect to RBT656 clock signal (27MHz)	<=	
	VI0-VI7	I	Connect to RBT656 video stream signals. <b>Available when XRGB=0 only.</b> Multiplexed MD56-MD63.	When XRGBEN=0, Pull-Up	See "2.3.3 Video Capture Interface",
	SDA,SCL	I	Connect to I2C device. <b>Available when XRGB=0 only.</b> Multiplexed MD54-MD55.	When XRGBEN=0, Pull-Up	
Graphics Memory Interface	MD0-MD31	IO	Connect to graphics memory data bus	<=	
	MD32-MD63	IO	Connect to graphics memory data bus. <b>Available when XRGBEN=1 only.</b>	1.XRGBEN=1 MD32-MD63=>OPEN 2.XRGBEN=0 MD32-MD63=>OPEN MD54-MD63=>Pull-Up	See "2.3.4 Graphics Memory Interface"
	MA0-MA13	O	Connect to graphics memory address and bank signals	Unused upper pins =>OPEN	See "5.4 Connection with memory"
	MRAS	O	Connect to graphics memory row address strobe signal	<=	
	MCAS	O	Connect to graphics memory column address strobe signal	<=	
	MWE	O	Connect to graphics memory write enable signal	<=	
	MDQM0-MDQM3	O	Connect to graphics memory data mask signals	<=	
	MDQM4-MDQM7	O	Connect to graphics memory data mask signals. <b>Available when XRGBEN=1 only.</b>	Memory bus width= 32bit (Both XRGBEN=0 and XRGBEN=1) MDQM4-MDQM7=>OPEN	See "2.3.4 Graphics Memory Interface"
	MCLKO	O	Connect to graphics memory clock and MCLKI*4	<=	
	MCLKI	I	Connect to MCLKO*3	<=	
Clock/System	CLKSEL1-0	I	Connect to GND or VDDH according to the input frequency to CLK	<=	See "2.3.5 Clock Input"
	CLK	I	Input a clock according to the setting of CLKSEL1-0	<=	See "2.3.5 Clock Input", "12.2.2 Power on Precaution". Input the clock when power-on.
	XRST	I	Input hardware reset signal	<=	See "12.2.2 Power on Precaution". XRST has to be Low level when power-on.
	S	I	Input PLL reset	<=	See "2.3.5 Clock Input", "12.2.2 Power on Precaution". S has to be Low level when power-on.
	CKM	I	-90<BCLKI<100MHz & Internal Clock*5=100MHz =>VDDH(Use BCLKI as Internal Clock)*6 - BCLKI<90MHz=>GND(Use PLL output)	<=	See "2.3.5 Clock Input"

**Note) This device is warranted under the above listed condition. No warranty made with other combination or treatments.**

Semiconductor devices fail with a known probability. Customer must use safety design ( such as redundant design, fire proof design, over current prevention design, and malfunction prevention design) so that failures will not cause accidents, injury or death.

\*1: :I=Input pin, O=Output pin , O(T)= Output Tri-state pin, IO=Bi-directional pin, Analog O=Analog output, Analog=Analog pin for DAC

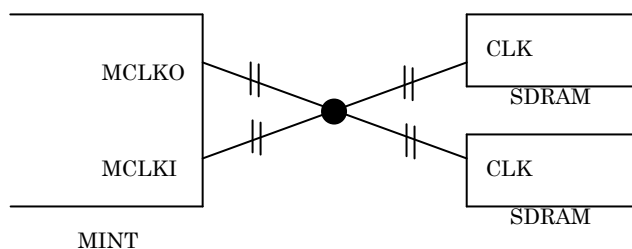
\*2:"<=" mark means treat a pin same as default

\*3:Recommend to be same length MCLKI to A, MCLKO to A, SDRAM CLK to A and take care the AC spec of graphics memory interface.

\*4:All of analog pins are possible to connect to GND when NOT use DAC. But if connect to GND, all of analog pins(includes AVD) have to connect GND.

\*5:The internal clock means "others clock"(memory clock, rendering clock,etc) which is set by COT bit of CCF register.

\*6: In case of CKM=L, BCLKI is used the both internal clock geometry and others module.



## 2.3 Pin Function

### 2.3.1 Host CPU interface

Table 2-1 Host CPU Interface Pins

Pin name	I/O	Description
MODE0-2	Input	Host CPU mode select
RDY_MODE	Input	Normally ready, Not ready select
BS_MODE	Input	BS signal with/without select
XRST	Input	Hardware reset ("L"=Reset, Set to low level when power-on)
D0-31	In/Out	Host CPU bus data
A2-A25	Input	Host CPU bus address (In the V832 mode, A[24] is connected to XMWR.)
BCLKI	Input	Host CPU bus clock
XBS	Input	Bus cycle start signal
XCS	Input	Chip select signal
XRD	Input	Read strobe signal
XWE0	Input	Write strobe for D0 to D7 signal
XWE1	Input	Write strobe for D8 to D15 signal
XWE2	Input	Write strobe for D16 to D23 signal
XWE3	Input	Write strobe for D24 to D31 signal
XRDY	Output Tri-state	Wait request signal (In the SH3 mode, when this signal is "0", it indicates the wait state; in the SH4, V832 and SPARClite modes, when this signal is "1", it indicates the wait state.)
DREQ	Output	DMA request signal (This signal is low-active in both the SH mode and V832 mode.)
DRACK/DMAAK	Input	Acknowledge signal in response to DMA request (DMAAK is used in the V832 mode; this signal is high-active in both the SH mode and V832 mode.)
DTACK/XTC	Input	DMA transfer strobe signal (XTC is used in the V832 mode. In the SH mode, this signal is high-active; in the V832 mode, it is low-active.)
XINT	Output	Interrupt signal issued to host CPU (In the SH mode, and SPARClite this signal is low-active; in the V832 mode, it is high-active)

With regard to BCLKI and XRST, the details, please refer "13.3.2Note at power-on".

MINT can be connected to the Hitachi SH4 (SH7750), SH3 (SH7709) NEC V832 and Fujitsu SPARClite (MB86833) without external circuit. In the SRAM interface mode, MINT can be used with any other CPU as well. The host CPU is specified by the MODE0 to 2 pins.

MODE 2	MODE 1	MODE 0	CPU
L	L	L	SH3
L	L	H	SH4
L	H	L	V832
L	H	H	SPARClite
H	X	X	Reserved

When the bus cycle terminates, a ready signal level can be set. When using the RDY\_MODE signal at “High” level, set two cycles as the CPU software wait of the CPU. (When BS\_MODE = “High” level, set the CPU software wait to three cycles.)

RDY_MODE	Ready signal mode
L	When the bus cycle terminates, sets the XRDY signal to the ‘not ready’ level.
H	When the bus cycle terminates, sets the XRDY signal to the ‘ready’ level.

A CPU with no BS (Bus Start) pin can be used. Setting can be performed in all CPU modes. Connection can be made to a CPU with no BS signal by setting the BS\_MODE signal to “High” level.

When not using the BS signal, fix the BS pin of MINT at “High” level.

When using the BS\_MODE signal as “High” level in the normally ready mode, set the CPU software wait to three cycles.

BS_MODE	BS signal mode
L	Connect to a CPU with the BS signal
H	Connect to a CPU without the BS signal

The data signal is 32 bits (fixed).

The address signal is 32 bits (per one double-word) × 24, and has a 64-Mbyte address field. (16-MByte address space is provided for V832 and SPARClite.)

The external bus operating frequency is up to 100 MHz.

In the SH4, V832, and SPARClite modes, when the XRDY signal is low, it is in the ready state. However, in the SH3 mode, when the XRDY signal is low, it is in the wait state. This signal is a tri-state output that is synchronized with the rising edge of BCLKI.

DMA data transfer is supported using an external DMA controller.

An interrupt signal is generated to the host CPU.

The XRST input must be kept low for at least 300 μs after setting the S (PLL reset) signal to high.

In the V832 mode, MINT signals are connected to the V832 CPU as follows:

MINT Pins	V832 Signals
A24	XMWR
DTACK	XTC
DRACK	DMAAK



## 2.3.2 Video output interface

**Table 2-2 Video Output Interface Pins**

Pin name	I/O	Description
DCLKO	Output	Dot clock signal for display
DCLKI	Input	Dot clock signal input
HSYNC	I/O	Horizontal sync signal output Horizontal sync input <in external sync mode>
VSYNC	I/O	Vertical sync signal output Vertical sync input <in external sync mode>
CSYNC	Output	Composite sync signal output
DISPE	Output	Display enable period signal
GV	Output	Graphics/video switch
R7-0	Output	Digital picture (R) output. These signals are multiplexed MD53-MD46. These pins are available when XRGEN = 0.
G7-0	Output	Digital picture (G) output. These signals are multiplexed MD45-MD38. These pins are available when XRGEN = 0.
B7-0	Output	Digital picture (B) output. These signals are multiplexed MD37-MD32 and MDQM7-6. These pins are available when XRGEN = 0.
XRGBEN	Input	Signal to switch between RGB1-0 output, capture singnals /memory bus (MD 63-MD32,MDQM7,6)
AOUTR	Analog Output	Analog Signal (R) output
AOUTG	Analog Output	Analog Signal (G) output
AOUTB	Analog Output	Analog Signal (B) output
ACOMPR	Analog	Analog (R) Compensation output
ACOMPG	Analog	Analog (G) Compensation output
ACOMPB	Analog	Analog (B) Compensation output
VREF	Analog	Analog Volatage Reference input
VRO	Analog	Analog Reference Current output

It is possible to output digital RGB, when XRGBEN = 0.(Memory bus=32bit)

Additional setting of external circuits can generate composite video signal.

Synchronous to external video signal display can be performed.

Either mode which is synchronous to DCLKI signal or one which is synchronous to dot clock, as for normal display can be selected.

Since HSYNC and VSYNC signals are set to input state after reset, these signals must be pulled up LSI externally.

The GV signal switches graphics and video at chroma key operation. When video is selected, the "Low" level is output.

AOUTR, AOUTG and AOUTB must be terminated at 75 ohm.

1.1-V is input to VREF. A bypass capacitor( with good high-frequency characteristics) must be inserted between VREF and AVS.

ACOMPR, ACOMPG and ACOMPB are tied to analog VDD via 0.1uF ceramic capacitors.

VRO must be pulled down to analog ground by a 2.7 k ohm resister.

When not using DAC, it is possible to connect all of analog pins(AVD, AOUTR,G,B, ACOMPR,G,B, VREF, VRO) to GND.

The 16bit/pixel color mode and 8bit/pixel color mode are converted to digital R:G:B=8:8:8 as the below.

A) 16bit/pixel color mode

	R:G:B=5:5:5 data in graphics memory	Digital R:G:B=8:8:8
→	0	0
→	1-31	Add 111 to lower 3bits Formula= $X*8+7$

B) 8bit/pixel color mode

	R:G:B=6:6:6 data in color pallete	Digital R:G:B=8:8:8
→	0	0
→	1-63	Add 11 to lower 2bits Formula= $X*4+3$

The Y,Cb,Cr mode is converted to R:G:B=8:8:8 directly.

### 2.3.3 Video Capture interface

**Table 2-3 Video Capture Interface Pins**

Pin name	I/O	Description
CCLK	Input	Digital video input clock signal input
VI7-0	Input	ITU656 Digital video data input. These pins are multiplexed MD63-MD56.
SCL	I/O	Serial Clock Line. This pin is multiplexed MD55
SDA	I/O	Serial Data Line. This pin is multiplexed MD54

Inputs ITU-RBT-656 format digital video signal

Digital video data input can be used only when the XRGBEN pin is "0". MD63-MD56 are assigned as the digital video data input pins.

When video capture is not used and the XRGBEN pin is 0, input the "High" level to MD63-MD56.

The SDA and SCL signal are multiplexed with memory controller pins (MD55-54).

Therefore set XRGBEN pin to "0" when use the I<sup>2</sup>C function.

## 2.3.4 Graphics memory interface

Table 2-3 Graphics memory interface pins

Pin name	I/O	Description
MD31-0	I/O	Graphics memory bus data
MD53-32	I/O	Graphics memory bus data or digital R7-0, G7-0, B7-2 output (when XRGBEN pin = 0)
MD55-54	I/O	Graphics memory bus data or I2C signals( When XRGBEN pin = 0)
MD63-56	I/O	Graphics memory bus data or video capture input (when XRGBEN pin =0)
MA0-14	Output	Graphics memory bus data
MRAS	Output	Row address strobe
MCAS	Output	Column address strobe
MWE	Output	Write enable
MDQM5-0	Output	Data mask
MDQM7-6	Output	Data mask or digital B1-0 output(when XRGBEN=0)
MCLK0	Output	Graphics memory clock output
MCLK1	Input	Graphics memory clock input

Connect the interface to the external memory used as memory for image data. The interface can be connected to 64-/128-/256-Mbit SD RAM (16- or 32-bit length data bus) without using any external circuit.

64 bits or 32 bits can be selected for the memory bus data.

Connect MCLK1 to MCLK0.

- When memory bus width is 32 bit and digital RGB output is used ( XRGBEN="0"), MD54-63 pins are set to "high level", and MD32-53 pins and MDQM4-7 pins are set to open.

- When memory bus width is 32 bit and digital RGB output is not used ( XRGBEN="1"), MD32-63 pins and MDQM4-7 pins are set to open.

When XRGBEN is fixed at "1", MD63-MD32 and MDQM7-MDQM6 can be used as graphics memory interface.

When XRGBEN is fixed at "0", these signals can be used as digital RGB output and video capture data input.

## 2.3.5 Clock input

**Table 2-4 Clock Input Pins**

Pin name	I/O	Description
CLK	Input	Clock input signal
S	Input	PLL reset signal ("L"=Reset) Input the signal "L"=>"H" regardless of CKM setting when power-on. Details, refer to "13.3.2Note at power-on"
CKM	Input	Clock mode signal
CLKSEL [1:0]	Input	Clock rate select signal

- Inputs source clock for internal operation clock and display dot clock. Normally, 4 Fsc (= 14.31818 MHz: NTSC) is input. An internal PLL generates the internal operation clock of 100 MHz and the display base clock of 400 MHz.
- For the internal operation clock, use either the output clock of the internal PLL or the bus clock input (BCLKI) from the host CPU. When the host CPU bus speed is 100 MHz, the BCLKI input should be selected. (CKM=H)

CKM	Clock mode
L	Output from internal PLL selected
H	Host CPU bus clock (BCLK1) selected

- In case of use BCLKI as internal clock (CKM=H) and use DCLKI as dot clock, it is possible to set the pins as the follows.

A) In case of NOT use Video capture function

Connect S pin to low level, and input a clock to CLK pin.(The clock has to input to CLK before releasing a hardware reset.)

B) In case of use Video capture function

Don't stop the PLL (Not fixed the S pin to low level).

- When CKM = L, selects input clock frequency when built-in PLL used according to setting of CLKSEL pins

CLKSEL1	CLKSEL0	Input clock frequency	Multiplication rate	Display reference clock
L	L	Inputs 13.5-MHz clock frequency	× 29	391.5 MHz
L	H	Inputs 14.32-MHz clock frequency	× 28	400.96 MHz
H	L	Inputs 17.73-MHz clock frequency	× 22	390.06 MHz
H	H	Reserved		

### 2.3.6 Test pins

Table 2-5 Test Pins

Pin name	I/O	Description
TESTH	Input	Input 3.3-V power.

### 2.3.7 Reset sequence

See “13.3.2 Note at power-on”.

## 3 PROCEDURE OF THE HARDWARE INITIALIZATION

### 3.1 Hardware reset

1. Do the hardware reset. (see section 13.3.2)
2. After the hardware reset, set the CCF(Change of Frequency) register (section 11.2.1).  
In being unstable cycle after the hardware reset, keep 32 bus cycles open.
3. Set the graphics memory interface register, MMR (Memory I/F Mode Register).  
After setting the CCF register, take 200 us to set the MMR register.  
In being unstable memory access cycle, keep 32 bus cycles open.
4. Other registers, except for the CCF register and the MMR register, should be set after setting the CCF register.  
In case of not using memory access, the MMR register could be set in any order after the CCF register is set.

### 3.2 Re-reset

1. Reset XRST signal.
2. See section 3.1 for registers setting after the procedure of re-reset.

### 3.3 Software reset

1. Set the value of the SRST register (see section 11.2.1) for re-reset.
2. It is not necessary to reset the CCF register and the MMR register again.

## 4 HOST INTERFACE

### 4.1 Operation Mode

#### 4.1.1 Host CPU mode

Select the host CPU by setting the MODE0 to MODE2 signals as follows:

Table 4-1 CPU Type Setting

MODE 2	MODE 1	MODE 0	CPU
L	L	L	SH3
L	L	H	SH4
L	H	L	V832
L	H	H	SPARClike
H	X	X	Reserved

#### 4.1.2 Ready signal mode

The MODE2 pin can be used to set the ready signal level when the bus cycle of the host CPU terminates. For the normally not ready mode, set the software wait to 0 or 1 cycles. When using this device in the normally ready mode, set the software wait to 2 cycles. When using this device in the normally not ready mode, set the software wait to one cycle. (When **BS\_MODE = H**, three cycles are needed for the software wait.)

The 'normally not ready mode' is the mode in which the MINT XRDY signal is always in the wait state and Ready is returned only when read/write is ready.

The 'normal ready mode' is the mode in which the MINT XRDY signal is always in the Ready state and it is put into the wait state only when read/write cannot be performed immediately.

Table 4-2 Ready Signal Mode

RDY MODE	Ready signal operation
L	Recognizes XRDY signal as 'not ready level' and terminates bus cycle (normally not ready mode)
H	Recognizes XRDY signal as 'ready level' and terminates bus cycle (normally ready mode)



### 4.1.3 BS signal mode

Connection to a CPU without the BS signal can be made via the **BS\_MODE** signal. This setting can be performed for all CPU modes. To connect to a CPU without the BS signal, set the **BS\_MODE** signal to “High” level.

When not using the BS signal, fix the BS pin of MINT at “High” level.

When using the **BS\_MODE** signal as “High” level, with the normally ready mode established, set the CPU software wait to three cycles.

Table 4-3 BS Signal Mode

BS MODE	Operation of BS signal
L	Connects to CPU with BS signal
H	Connects to CPU without BS signal

### 4.1.4 Endian

MINT operates in little-endian mode. All the register address descriptions in the specifications are byte address in little endian. When using a big-endian CPU, note that the byte-or word-addresses are different from these descriptions.

## 4.2 Access Mode

### 4.2.1 SRAM interface

Data can be transferred to/from MINT using SRAM access protocol. MINT internal registers and graphics memory are all mapped to the physical address area of the host processor.

MINT uses hardware wait based on the XRDY signal, enabling the hardware wait setting of the host CPU. When using the normally not ready mode, set the software wait to "1". When using the normally ready mode, set the software wait to "2". (When using the **BS\_MODE** signal as "High" level, with the normally ready mode established, set the CPU software wait to three cycles.) Switch the ready mode using the **RDY\_MODE** signal.

#### CPU Read

The host processor reads data from internal registers and memory of MINT in double-word (32 bit) units. Valid data is output continuously while XRD and XCS are being asserted at a "Low" level after XRDY has been asserted.

#### CPU Write

The host CPU writes data to internal registers and memory of MINT in byte, word(16 bit) and double-word( 32 bit) units.

### 4.2.2 FIFO interface (fixed transfer destination address)

This interface transfers display lists stored in host memory. Display list information is transferred efficiently using a single address mode DMA transfer. Data can be transferred to FIFO in relation to FIFO buffer area mapped in memory area using SRAM interface or dual address mode.

## 4.3 DMA Transfer

### 4.3.1 Data transfer unit

DMA transfer is performed in double-word (32 bits) units or 8 double-word (32 bytes) units. Byte and word access is not supported.

Note: 8 double-word transfer is supported only in the SH4 mode.

### 4.3.2 Address mode

#### Dual address mode (mode using ACK)

DMA is performed at memory-to-memory transfer between host memory and registers mapped in memory space or graphics memory (destination). Both the host memory address and MINT is used. In the SH4 mode, the 1 double-word transfer (32 bits) and 8 double-word transfer (32 bytes) can be used.

When the CPU transfer destination address is fixed, data can also be transferred to the FIFO interface. However, in this case, even the SH4 mode supports only the 1 double-word transfer.

DREQ and DRACK pins and SRAM interface signals are used. In V832, the DREQ, DMAAK, and XTC pins and SRAM interface signals are used.

Note: The SH3 mode supports the direct address mode; it does not support the indirect address mode.

#### Dual address mode (mode not using ACK)

When not using the ACK signal with the dual address mode established, set bit3 at HostBase+0004h (DNA: Dual address No Ack mode) to 1.

When the ACK is not used, the DREQ signal is in the edge mode and the DREQ signal is negated per transfer and then reasserted it in the next cycle. If processing cannot be performed immediately inside MINT, the DREQ signal remains negated.

The transfer count register (DTC) of MINT is not used, so in order to end DMA transfer, write "1" to the DMA transfer stop register (DTS) from the CPU.

Note 1: In the dual DMA mode (mode without ACK), the destination address can be used only for the FIFO.

In DMA transfer to the graphics memory, etc., use the dual DMA mode.

Note 2: DMA read is not supported.

#### Single address mode (FIFO interface)

Data is transferred between host memory (source) and FIFO (destination). Only the address output from the host memory is used, and the data is transferred to the FIFO. This mode does not support data write to the host memory. When the FIFO is full, the DMA transfer is suspended.

The 1 double-word transfer (32 bits) and the 8 double-word transfer (32 B) can be used.

DREQ, DTACK, and DRACK signal are used.

Note: The single-address mode is supported only in the SH4 mode.

### 4.3.3 Bus mode

MINT supports the DMA transfer cycle steal mode and burst mode according to setting of external DMA mode.

#### **Cycle steal mode (In the V832 mode, the burst mode is called the single transfer mode.)**

In the cycle steal mode, the right to use the bus is obtained or released at every data transfer of 1 unit. The DMA transfer unit can be selected from between the 1 double-word (32 bits) and 8 double-words (32 B).

#### **Burst mode (In the V832 mode, the burst mode is called the demand transfer mode.)**

When DMA transfer is started, the right to use the bus is acquired and the transfer begins. The data transfer unit can be selected from between the 1 double-word (32 bits) and 8 double-words (32 B).

Note: When performing DMA transfer in the dual-address mode, a function for automatically negating DREQ is provided based on the setting of the DBM register.

### 4.3.4 DMA transfer request

#### **Single-address mode**

DMA is started when the MINT issues an external request to DMAC of the host processor.

Set the transfer count in the transfer count register of the MINT and then issue DREQ.

Fix the CPU destination address to the FIFO address.

#### **Dual-address mode**

DMA is started by two procedures: MINT issues an external request to DMAC of the host processor, or the CPU itself is started (auto request mode, etc.). In Ack use mode, set the transfer count in the transfer count register of MINT and then issue DREQ.

Note: In the Ack unused mode and the V832 mode requires no setting of the transfer count register.

### 4.3.5 Ending DMA transfer

- SH3/SH4

When the MINT transfer count register is set to 0, DMA transfer ends and DREQ is negated.

- V832

When the XTC signal from the CPU is low-asserted while the DMAAK signal to S MINT is high-asserted, the end of DMA transfer is recognized and DREQ is negated.

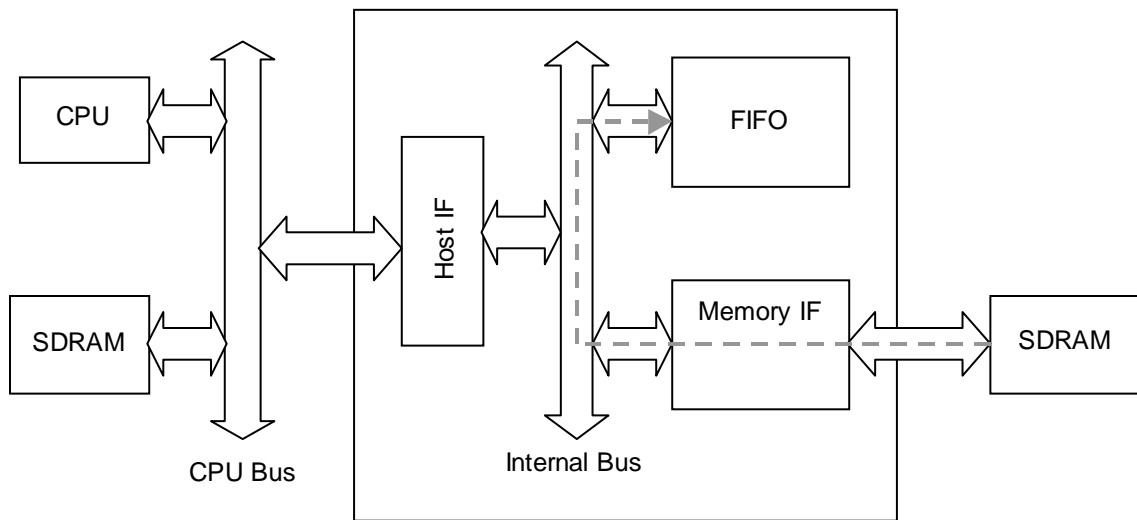
- The end of DMA transfer is detected in two ways: the DMA status register (DST) is polled, and an interrupt to end the drawing command (FD000000<sub>H</sub>) is added to the display list and the interrupt is detected.
- In the dual address mode (mode not using ACK), the DMA transfer count register (DTC) is not used, so the DMA ending cannot be determined. The DREQ signal can be negated to end DMA by writing 1 from the CPU to the DMA transfer stop register (DTS) of MINT at DMA transfer end.

## 4.4 Transfer of Local Display List

This is the mode in which the MINT internal bus is used to transfer the display list stored in the graphics memory to the FIFO interface.

During transfer of the local display list, the host bus can be used for CPU read/write.

How to transfer list: Store the display list in the local memory of MINT, set the transfer source local address (LSA) and the transfer count (LCO), and then issue a request (LREQ). Whether or not the local display list is currently being transferred is checked using the local transfer status register (LSTA).



Transfer Path for Local Display List

## 4.5 Interrupt

MINT issues interrupt requests to the host CPU. Following shows the types of interrupt factor and they can be enabled/disabled by IMASK (Interrupt Mask Register).

- Vertical synchronization detect
- Field synchronization detect
- External synchronization error detect
- Drawing command error
- Drawing command execution end

## 4.6 SH3 Mode

In the SH3 mode, operation is assured under the following conditions:

### Normally not ready mode

- BCLK (CPU bus clock) is 50 MHz or less.
- The XWAIT setup time is 9.0 ns or less.

### Normally ready mode

- Three cycles or more are set for the software wait.

## 4.7 Wait

### Software wait

The software wait is a wait performed on the CPU side; this wait specifies how many cycles of the ready signal (XRDY) sampling timing is ignored.

### Hardware wait

The hardware wait is a wait on the MINT side that occurs when MINT itself cannot read/write data immediately.

## 4.8 Memory Map

The following shows the memory map of MINT to the host CPU memory space. The address is mapped differently in SH3, SH4 and V832.

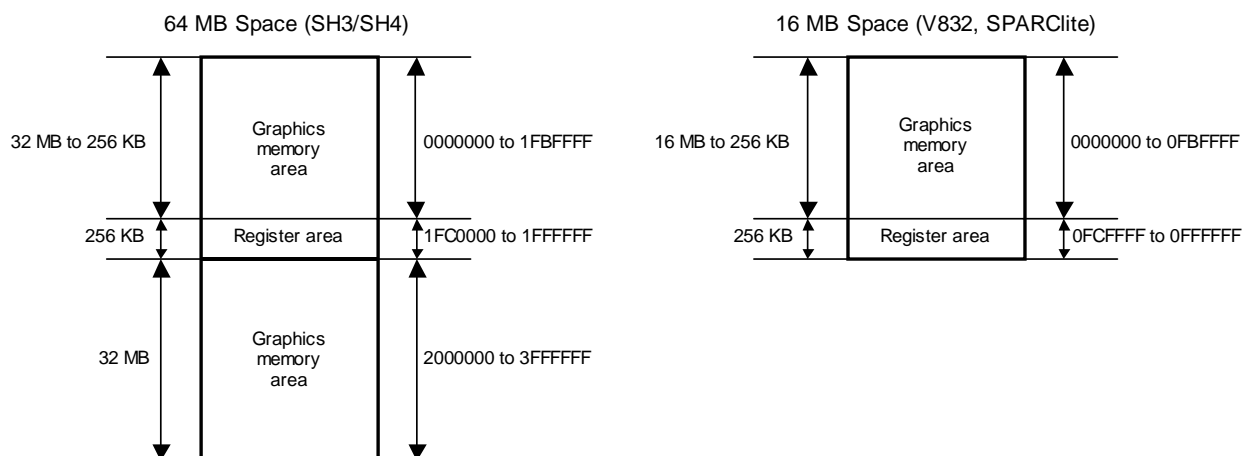


Fig. 4.1 Memory Map

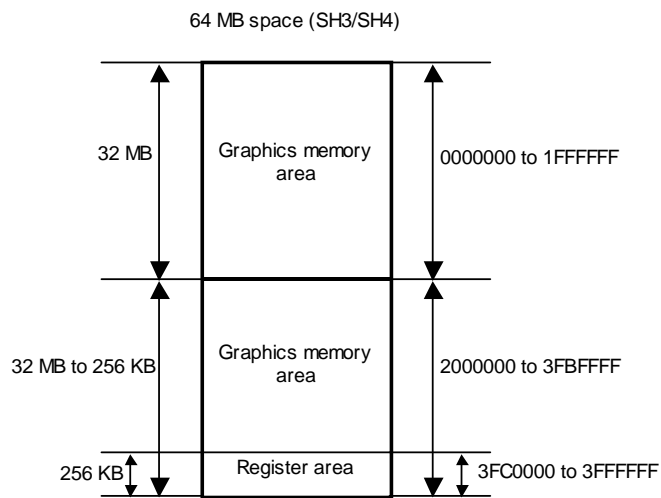
Table 4-4 Address Space in SH3/SH4 Mode

Size	Resource	Base address	(Name)
32 MB to 256 KB		00000000	
64 KB	Host interface registers	01FC0000	(HostBase)
32 KB	Display registers	01FD0000	(DisplayBase)
32 KB	Capture registers	01FD8000	(CaptureBase)
32 KB	Drawing registers	01FF0000	(DrawBase)
32 KB	Geometry engine registers	01FF8000	(GeometryBase)
32 MB	Graphics memory	02000000	

Table 4-5 Address Space in V832, SPARClite Mode

Size	Resource	Base address	(Name)
16 MB to 256 KB	Graphics memory	00000000	
64 KB	Host interface registers	00FC0000	(HostBase)
32 KB	Display registers	00FD0000	(DisplayBase)
32 KB	Capture registers	00FD8000	(CaptureBase)
32 KB	Drawing registers	00FF0000	(DrawBase)
32 KB	Geometry engine registers	00FF8000	(GeometryBase)

When the SH3 or SH4 mode is used, the register area can be moved by writing 1 to bit 0 at HostBase + 005Ch (RSW: Register location Switch). In the initial state, the register space is at the center (1FC0000) of the 64 MB space; access MINT after about 20 bus clocks after writing 1 to RSW.



**Fig. 4.2 Memory Map**

**Table 4-6 Address Mapping in SH3/SH4 Mode**

Size	Resource	Base address	(Name)
64 MB to 256 KB	Graphics memory	00000000	
64 KB	Host interface registers	03FC0000	(HostBase)
32 KB	Display registers	03FD0000	(DisplayBase)
32 KB	Capture registers	03FD8000	(CaptureBase)
32 KB	Drawing registers	03FF0000	(DrawBase)
32 KB	Geometry engine registers	03FF8000	(GeometryBase)



## 5 Graphics Memory

### 5.1 Configuration

The MINT uses local external memory (Graphics memory) for drawing and display management. The configuration of this Graphics memory is described as follows:

#### 5.1.1 Data type

The MINT handles the following types of data. Display list can be stored in the host (main) memory as well. Texture/tile pattern and text pattern can be defined by a display list as well.

##### Drawing Frame

This is a rectangular image data field for 2D/3D drawing. The MINT is able to have plural drawing frames and display a part of these area if it is set to be bigger than display size. The maximum size is 4096x4096 pixel in 32 pixel units. And both indirect color ( 8 bits / pixel) and direct color ( 16 bits / pixel) mode are applicable.

##### Display Frame

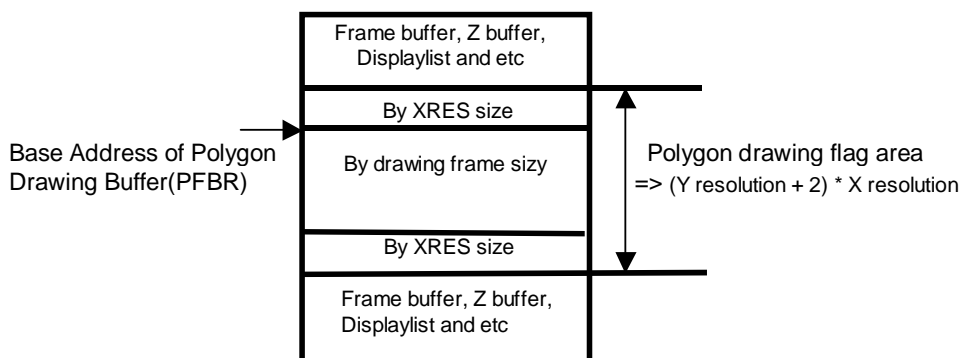
This is a rectangle picture area for display. The MINT is able to set display layer up to 6 layers.

##### Z Buffer

Z buffer is required for eliminating hidden surfaces. In 16 bits modes, 2 bytes and in 8 bits mode, 1 byte are required per 1 pixel. This area has to be cleared before drawing.

##### Polygon Drawing Flag Buffer

This area is used for polygon drawing. It is required 1 bit memory area per 1 pixel and 1 x-axis line area both backward and forward of it. This area has to be cleared before drawing.



##### Displaylist Buffer

The displaylist is a list of drawing commands and parameters.

##### Texture Pattern

This pattern is used for texture mapping. The maximum size is up to 4096 x 4096 pixels.

##### Cursor Pattern

This is used for hardware cursor. The data format is indirect color ( 8 bits / pixel) mode. And the MINT is able to display two cursor of 64 x 64 pixel size.

## 5.1.2 Memory Mapping

A graphics memory is mapped linearly to host CPU address field. Each of these above data is able to be allocated anywhere in the Graphics memory according to the respective register setting. ( However there is some restrictions of an addressing boundary depending on a data type.)

## 5.1.3 Data Format

### Direct Color ( 16 bits / pixel )

This data format is described RGB as each 5 bit. Bit15 is used for alpha bit of layer blending.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A	R					G					B				

### Indirect Color ( 8 bits / pixel )

This data format is a color index code for looking up table (palette).

7	6	5	4	3	2	1	0
Color Code							

### Z Value

It is possible to use Z value as 8 bits or 16 bits. These data format are unsigned integer.

#### 1 ) 16 bits mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Unsigned Integer															

#### 2 ) 8 bits mode

7	6	5	4	3	2	1	0
Unsigned Integer							

### Polygon Drawing Flag

This data format is 1 bit per 1 pixel.

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

### Texture / Tile Pattern

It is possible to use a pattern as direct color mode ( 16 bits / pixel) or indirect color mode ( 8 bits / pixel).

#### 1 ) Direct color mode ( 16 bits / pixel)

This data format is described RGB as each 5 bit. Bit15 is used for alpha bit of stencil or stencil blending. ( Only texture mapping)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A	R					G					B				

#### 2) Indirect color mode ( 8 bits / pixel)

This data format is a color index code for looking up table (palette).

7	6	5	4	3	2	1	0
Color Code							

### Cursor Pattern

This data format is a color index code for looking up table (palette).

7	6	5	4	3	2	1	0
Color Code							

### Video Capture data

This data format is Y:Cb:Cr=4:2:2 and 32 bits per 2 pixel.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Y0								Cb							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Y1								Cr							

### Direct Color ( 32 bits / pixel )

This data format is described RGB as each 8 bit. Bit31 is used for alpha bit of layer blending. But the MINT does not support this color mode drawing. Therefore please draw this layer by CPU writing.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G								B							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
A	Reserved							R							

## 5.2 Frame Management

### 5.2.1 Single Buffer

The entire or partial area of the drawing frame is assigned as a display frame. The display field is scrolled by relocating the position of the display frame. When the display frame crosses the border of the drawing frame, the other side of the drawing frame is displayed, assuming that the drawing frame is rolled over (top and left edges assumed logically connected to bottom and right edges, respectively). To avoid the affect of drawing on display, the drawing data can be transferred to the Graphics Memory in the blanking time period.

### 5.2.2 Double Buffer

Two drawing frames are set. While one frame is displayed, drawing is done at the other frame. Flicker-less animation can be performed by flipping these two frames back and forth. Flipping is done in the blanking time period. There are two flipping modes: automatically at every scan frame period, and by user control. The double buffer is assigned independently for the L2, L3, L4, L5 layers.

## 5.3 Memory Access

### 5.3.1 Memory Access by host CPU

Graphics memory is mapped linearly to host CPU address field. The host CPU can access the Graphics memory like a SRAM.

### 5.3.2 Priority of memory accessing

The priority of Graphics memory accessing is the follows:

1. Refresh
2. Video Capture
3. Display processing
4. Host CPU accessing
5. Drawing accessing

## 5.4 Connection with memory

### 5.4.1 Connection with memory

The memory controller of MINT supports simple connection with SD/FCRAM by setting MMR(Memory Mode Register).

If there is N(=11 to 13) address pins in SD/FCRAM, please connect the SD/FCRAM address(A[n]) pin to the MINT's memory address(MA[n]) pin and SD/FCRAM bank pin to the MINT's next address(MA[N]) pin. Then please set MMR by a number and type of memory.

The follows are the connection table between MINT pin and SD/FCRAM pin.

**64M bit SDRAM(x16 bit)**

MINT pins	SDRAM pins
MA[11:0]	A[11:0]
MA12	BA0
MA13	BA1

**64M bit SDRAM(x32 bit)**

MINT pins	SDRAM pins
MA[10:0]	A[10:0]
MA11	BA0
MA12	BA1

**128M bit SDRAM(x16 bit)**

MINT pins	SDRAM pins
MA[11:0]	A[11:0]
MA12	BA0
MA13	BA1

**128M bit SDRAM(x32 bit)**

MINT pins	SDRAM pins
MA[11:0]	A[11:0]
MA12	BA0
MA13	BA1

**256M bit SDRAM(x16 bit)**

MINT pins	SDRAM pins
MA[12:0]	A[12:0]
MA13	BA0
MA14	BA1

**16M bit FCRAM(x16 bit)**

MINT pins	FCRAM pins
MA[10:0]	A[10:0]
MA11	BA

## 6 DISPLAY CONTROLLER

### 6.1 Overview

#### Display control

Window display can be performed for six layers. Window scrolling, etc., can also be performed.

#### Backward compatibility

Backward compatibility with previous products is supported in the four-layer display mode or in the left/right split display mode.

#### Video timing generator

The video display timing is generated according to the display resolution (from  $320 \times 240$  to  $1024 \times 768$ ).

#### Color look-up

There are two sets of color look-up tables by palette RAM for the indirect color mode (8 bits/pixel).

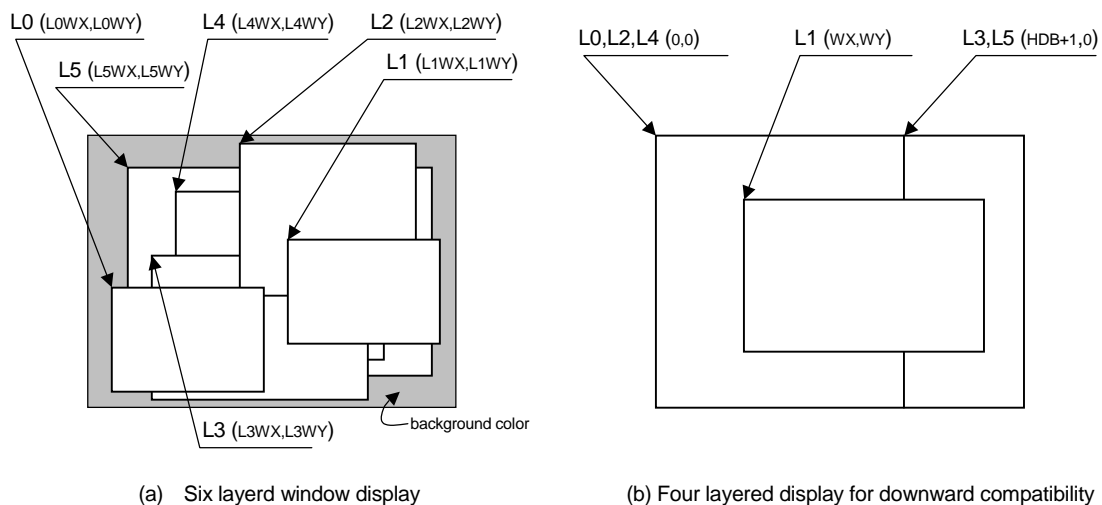
#### Cursor

Two sets of hardware cursor patterns (8 bits/pixel,  $64 \times 64$  pixels each) can be used.

### 6.2 Display Function

#### 6.2.1 Layer configuration

Six-layer window display is performed. Layer overlay sequence can be set in any order. A four-layer display mode and left/right split display mode are also provided, supporting backward compatibility with previous products.



## Configuration of Display Layers

The correspondence between the display layers for this product and for previous products is shown below.

Layer correspondence		Coordinates of starting point		Width/height	
		Window mode	Compatibility mode	Window mode	Compatibility mode
L0	C	(L0WX, L0WY)	(0, 0)	(L0WW, L0WH + 1)	(HDP + 1, VDP + 1)
L1	W	(L1WX, L1WY)	(WX, WY)	(L1WW, L1WH + 1)	(WW, WH + 1)
L2	ML	(L2WX, L2WY)	(0, 0)	(L2WW, L2WH + 1)	(HDB + 1, VDP + 1)
L3	MR	(L3WX, L3WY)	(HDB, 0)	(L3WW, L3WH + 1)	(HDP – HDB, VDP + 1)
L4	BL	(L4WX, L4WY)	(0, 0)	(L4WW, L4WH + 1)	(HDB + 1, VDP + 1)
L5	BR	(L5WX, L5WY)	(HDB, 0)	(L5WW, L5WH + 1)	(HDP – HDB, VDP + 1)

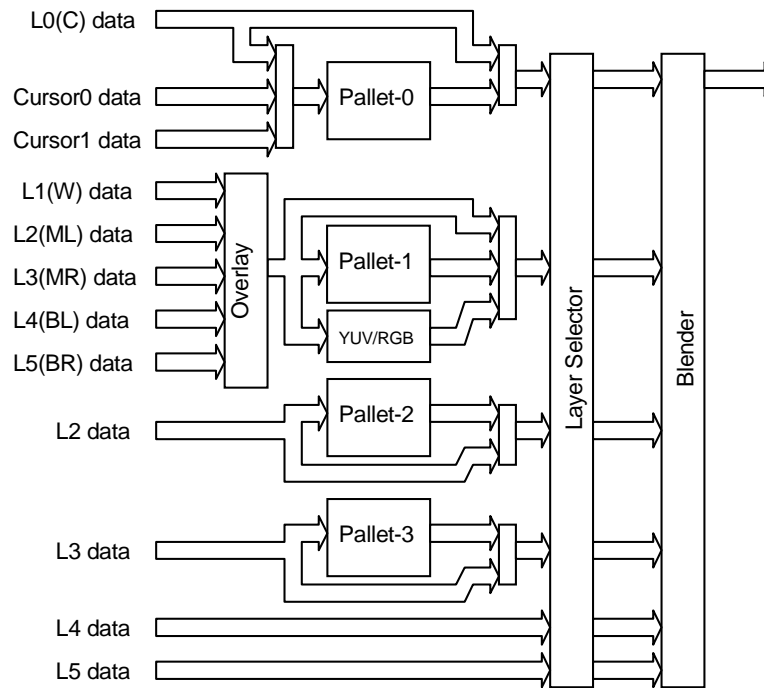
C, W, ML, MR, BL, and BR above mean layers for previous products. The window mode or the compatibility mode can be selected for each layer. It is possible to use new functions through minor program changes by allowing the coexistence of display modes instead of separating them completely.

However, if high resolutions are displayed, the count of layers that can be displayed simultaneously and pixel data may be restricted according to the graphics memory ability to supply data.

## 6.2.2 Overlay

### (1) Overview

Image data for the six layers (L0 to L5) is processed as shown below.



The fundamental flow is: Palette → Layer selection → Blending. The palettes convert 8-bit color codes to the RGB format. The layer selector exchanges the layer overlay sequence arbitrarily. The blender performs blending using the blend coefficient defined for each layer or overlays in accordance with the transparent-color definition.

The L0 layer corresponds to the C layer for previous products and shares the palettes with the cursor. As a result, the L0 layer and cursor are overlaid before blend operation.

The L1 layer corresponds to the W layer for previous products. To implement backward compatibility with previous products, the L1 layer and lower layers are overlaid before blend operation.

The L2 to L5 layers have two paths; in one path, these layers are input to the blender separately and in the other, these layers and the L1 layer are overlaid and then are input to the blender. When performing processing using the extended mode, select the former; when performing the same processing as previous products, select the latter. It is possible to specify which one to select for each layer.



## (2) Overlay mode

Image layer overlay is performed in two modes: simple priority mode, and blend mode.

In the simple priority mode, processing is performed according to the transparent color defined for each layer. When the color is a transparent color, the value of the lower layer is used as the image value for the next stage; when the color is not a transparent color, the value of the layer is used as the image value for the next stage.

$$\begin{aligned} D_{\text{view}} &= D_{\text{new}} \text{ (when } D_{\text{new}} \text{ does not match transparent color)} \\ &= D_{\text{lower}} \text{ (when } D_{\text{new}} \text{ matches transparent color)} \end{aligned}$$

When the L1 layer is in the YCbCr mode, transparent color checking is not performed for the L1 layer; processing is always performed assuming that transparent color is not used.

In the blend mode, the blend ratio “r” defined for each layer is specified using 8-bit tolerance, and the following operation is performed:

$$D_{\text{view}} = D_{\text{new}} * r + D_{\text{lower}} * (1 - r)$$

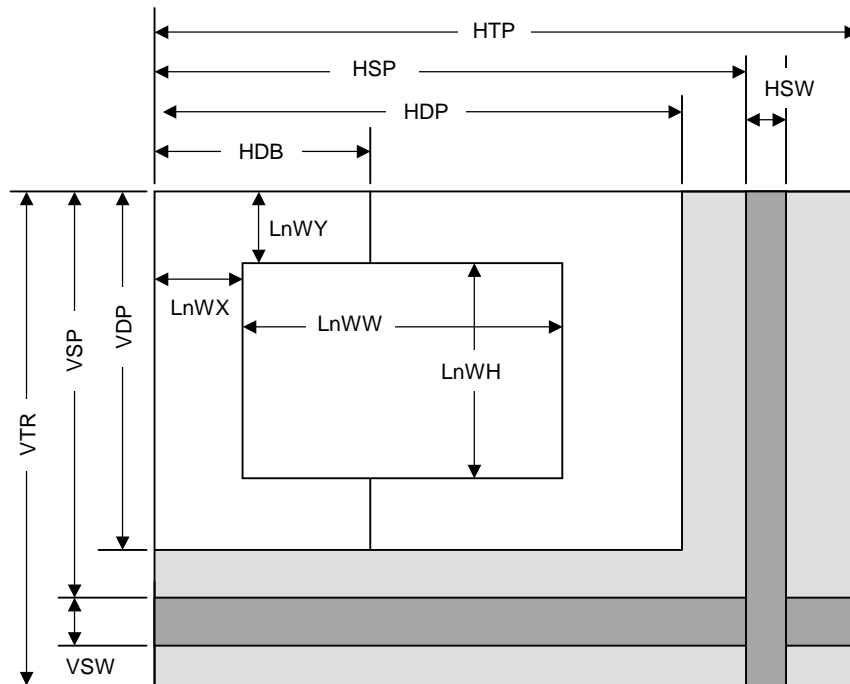
Blending is enabled for each layer by mode setting and a specific bit of the pixel is set to “1”. For 8 bits/pixel, the MSB of RAM data enables blending; for 16 bits/pixel, the MSB of data of the relevant layer enables blending; for 24 bits/pixel, the MSB of the word enables blending.

## (3) Blend coefficient layer

In the normal blend mode, the blend coefficient is fixed for each layer. However, in the blend coefficient layer mode, the L5 layer can be used as the blend coefficient layer. In this mode, the blend coefficient can be specified for each pixel, providing gradation, for example. When using this mode, set the L5 layer(L5M and L5EM register) to 8 bits/pixel, window display mode and extend overlay mode.

### 6.2.3 Display parameters

The display area is defined according to the following parameters. Each parameter is set independently at the respective register.



**Fig. 5.1 Display Parameters**

Note: The actual parameter settings are little different from the above. The details, please refer “11.3.1 Interlaced mode”.

HTP	Horizontal Total Pixels
HSP	Horizontal Synchronize pulse Position
HSW	Horizontal Synchronize pulse Width
HDP	Horizontal Display Period
HDB	Horizontal Display Boundary
VTR	Vertical Total Raster
VSP	Vertical Synchronize pulse Position
VSW	Vertical Synchronize pulse Width
VDP	Vertical Display Period
LnWX	Layer n Window position X
LnWY	Layer n Window position Y
LnWW	Layer n Window Width
LnWH	Layer n Window Height

When not splitting the window, set HDP to HDB and display only the left side of the window. The settings must meet the following relationship:

$$0 < \text{HDB} \leq \text{HDP} < \text{HSP} < \text{HSP} + \text{HSW} + 1 < \text{HTP}$$

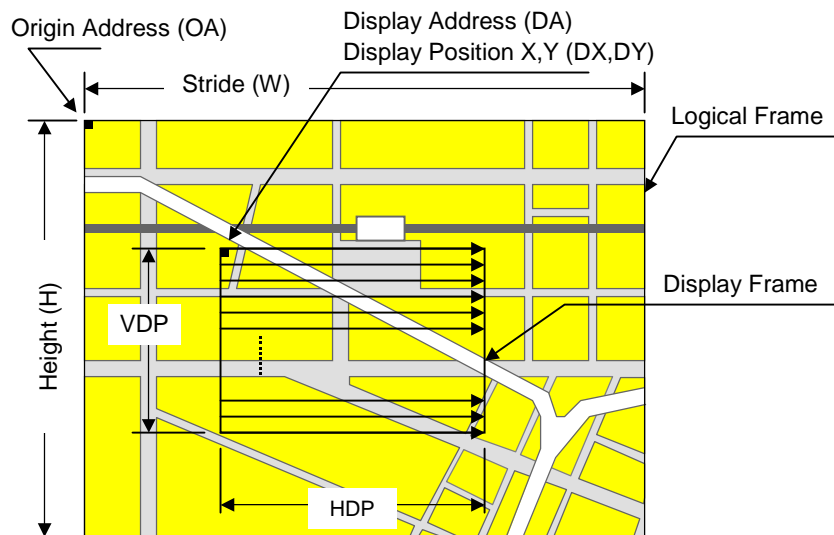
$$0 < \text{VDP} < \text{VSP} < \text{VSP} + \text{VSW} + 1 < \text{VTR}$$

## 6.2.4 Display position control

The graphic image data to be displayed is located in the logical 2D coordinates space (logical graphics space) in the Graphics Memory. There are six logical graphics spaces as follows:

- L0 layer
- L1 layer
- L2 layer
- L3 layer
- L4 layer
- L5 layer

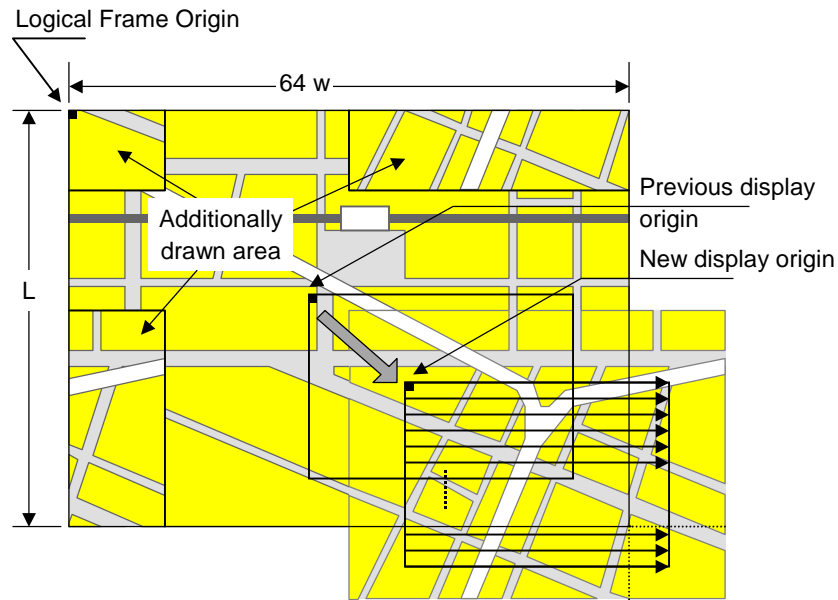
The relation between the logical graphics space and display position is defined as follows:



**Fig. 5.2 Display Position Parameters**

OA	Origin Address	Origin address of logical graphics space. Memory address of top left edge pixel in logical frame origin
W	Stride	Width of logical graphics space. Defined in 64-byte unit
H	Height	Height of logical graphics space. Total raster (pixel) count of field
DA	Display Address	Display origin address. Top left position address of display frame origin
DX DY	Display Position	Display origin coordinates. Coordinates in logical frame space of display frame origin

MINT scans the logical graphics space as if the entire space is rolled over in both the horizontal and vertical directions. Using this function, if the display frame crosses the border of the logical graphics space, the part outside the border is covered with the other side of the logical graphics space, which is assumed to be connected cyclically as shown below:



**Fig. 5.3 Wrap Around of Display Frame**

The expression of the X and Y coordinates in the frame and their corresponding linear addresses (in bytes) is shown below.

$$A(x,y) = x \times \text{bpp}/8 + 64wy \quad (\text{bpp} = 8 \text{ or } 16)$$

The origin of the displayed coordinates has to be within the frame. To be more specific, the parameters are subject to the following constraints:

$$0 \leq DX < w \times 64 \times 8/\text{bpp} \quad (\text{bpp} = 8 \text{ or } 16)$$

$$0 \leq DY < H$$

DX, DY, and DA have to indicate the same point within the frame. In short, the following relationship must be satisfied.

$$DA = OA + DX \times \text{bpp}/8 + 64w \times DY \quad (\text{bpp} = 8 \text{ or } 16)$$

## 6.3 Display Color

Color data is displayed in the following modes:

### Indirect color (8 bits/pixel)

In this mode, the index of the palette RAM is displayed. Data is converted to image data consisting of 6 bits for R, G, and B via the palette RAM and is then displayed.

### Direct color (16 bits/pixel)

Each level of R, G, and B is represented using 5 bits.

### Direct color (24 bits/pixel)

Each level of R, G, and B is represented using 8 bits.

### YCbCr color (16 bits/pixel)

In this mode, image data is displayed with YCbCr = 4:2:2. Data is converted to image data consisting of 8 bits for R, G, and B using the operation circuit and is then displayed.

The display colors for each layer are shown below.

Layer	Compatibility mode	Extended mode
L0	Direct color (16, 24), Indirect color (P0)	Direct color (16, 24), Indirect color (P0)
L1	Direct color (16, 24), Indirect color (P1), YCbCr	Direct color (16, 24), Indirect color (P1), YCbCr
L2	Direct color (16, 24), Indirect color (P1)	Direct color (16, 24), Indirect color (P2)
L3	Direct color (16, 24), Indirect color (P1)	Direct color (16, 24), Indirect color (P3)
L4	Direct color (16, 24), Indirect color (P1)	Direct color (16, 24)
L5	Direct color (16, 24), Indirect color (P1)	Direct color (16, 24)

“Pn” stands for the corresponding palette RAM. Four palettes are used as follows:

### Palette 0 (P0)

This palette corresponds to the C-layer palette for previous products. This palette is used for the L0 layer. This palette can also be used for the cursor.

### Palette 1 (P1)

This palette corresponds to the M/B layer palette for previous products. In the compatibility mode, this palette is common to layers L1 to 5. In the extended mode, this palette is dedicated to the L1 layer.

### Palette 2 (P2)

This palette is dedicated to the L2 layer. This palette can be used only for the extended mode.

### Palette 3 (P3)

This palette is dedicated to the L2 layer. This palette can be used only for the extended mode.

## **6.4 Cursor**

### **6.4.1 Cursor display function**

MINT can display two hardware cursors. Each cursor is specified as  $64 \times 64$  pixels, and the cursor pattern is set in the Graphics Memory. The indirect color mode (8 bits/pixel) is used and the L0 layer palette is used. However, transparent color control (handling of transparent color code and code 0) is independent of L0 layer. Blending with lower layer is not performed.

### **6.4.2 Cursor control**

The display priority for hardware cursors is programmable. The cursor can be displayed either on upper or lower the L0 layer using this feature. A separate setting can be made for each hardware cursor. If part of a hardware cursor crosses the display frame border, the part outside the border is not shown.

Usually, cursor 0 is preferred to cursor 1. However, with cursor 1 displayed upper the L0 layer and cursor 0 displayed lower the L0 layer, the cursor 1 display is preferred to the cursor 0.

## 6.5 Display Scan Control

### 6.5.1 Applicable display

The following table shows typical display resolutions and their synchronous signal frequencies. The pixel clock frequency is determined by setting the division rate of the display reference clock. The display reference clock is either the internal PLL (400.9 MHz at input frequency of 14.318 MHz), or the clock supplied to the DCLKI input pin. The following table gives the clock division rate used when the internal PLL is the display reference clock:

**Table 4-1 Resolution and Display Frequency**

Resolution	Division rate of reference clock	Pixel frequency	Horizontal total pixel count	Horizontal frequency	Vertical total raster count	Vertical frequency
320 × 240	1/60	6.7 MHz	424	15.76 kHz	263	59.9 Hz
400 × 240	1/48	8.4 MHz	530	15.76 kHz	263	59.9 Hz
480 × 240	1/40	10.0 MHz	636	15.76 kHz	263	59.9 Hz
640 × 480	1/16	25.1 MHz	800	31.5 kHz	525	59.7 Hz
854 × 480	1/12	33.4 MHz	1062	31.3 kHz	525	59.9 Hz
800 × 600	1/10	40.1 MHz	1056	38.0 kHz	633	60.0 Hz
1024 × 768	1/6	66.8 MHz	1389	48.1 kHz	806	59.9 Hz

Pixel frequency =  $14.318 \text{ MHz} \times 28 \times \text{reference clock division rate (when internal PLL selected)}$   
= DCLKI input frequency × reference clock division rate (when DCLKI selected)

Horizontal frequency = Pixel frequency/Horizontal total pixel count

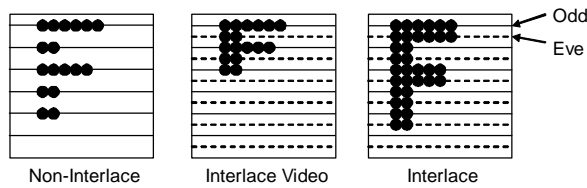
Vertical frequency = Horizontal frequency/Vertical total raster count

## 6.5.2 Interlace display

MINT can perform both a non-interlace display and an interlace display.

When the DCM register synchronization mode is set to interlace video (11), images in memory are output in odd and even rasters alternately to each field, and one frame (odd + even fields) forms one screen.

When the DCM register synchronization mode is set to interlace (10), images in memory are output in raster order. The same image data is output to odd fields and even fields. Consequently, the count of rasters on the screen is half of that of interlace video. However, unlike the non-interlace mode, there is a distinction between odd and even fields depending on the phase relationship between the horizontal and vertical synchronous signals.



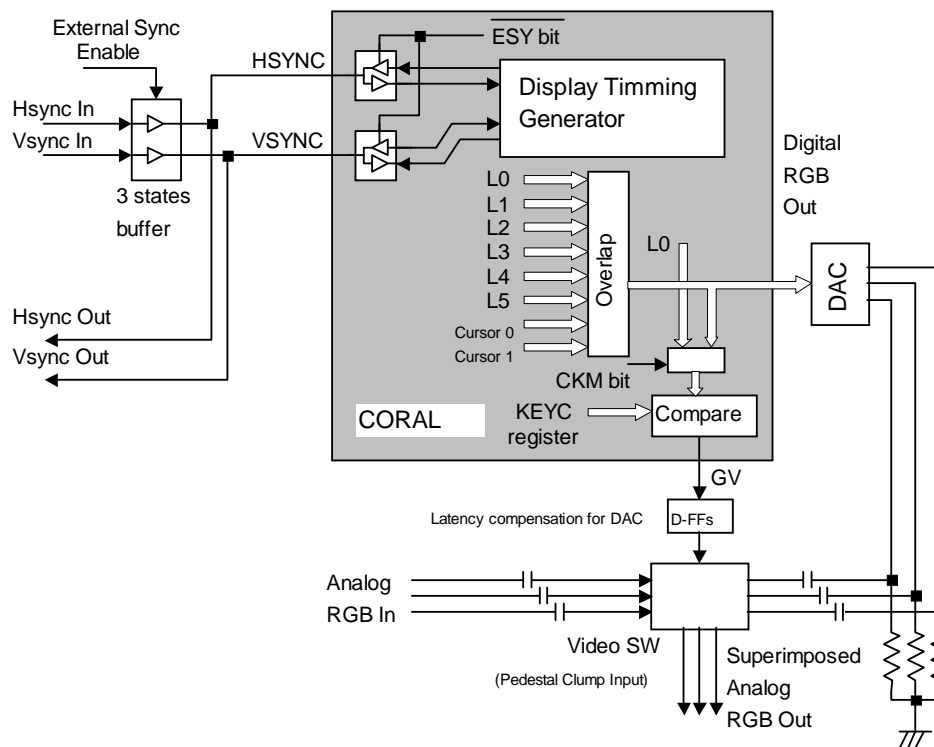
**Fig. 5.4 Display Difference between Synchronization Modes**



## 6.6 The external synchronous signal

The display scan can be performed by synchronizing horizontal/vertical synchronous signal from the external.

In selecting the external synchronization mode, MINT is sampling the HSYNC signal and displays the synchronizing the external video signal. Either the internal PLL clock or the DCLKI input signal could be selected for the sampling clock. Also, the superimposed analog output is performed by the chroma key process. The following diagram shows an example of the external synchronization circuit.



**An example of the external synchronization circuit**

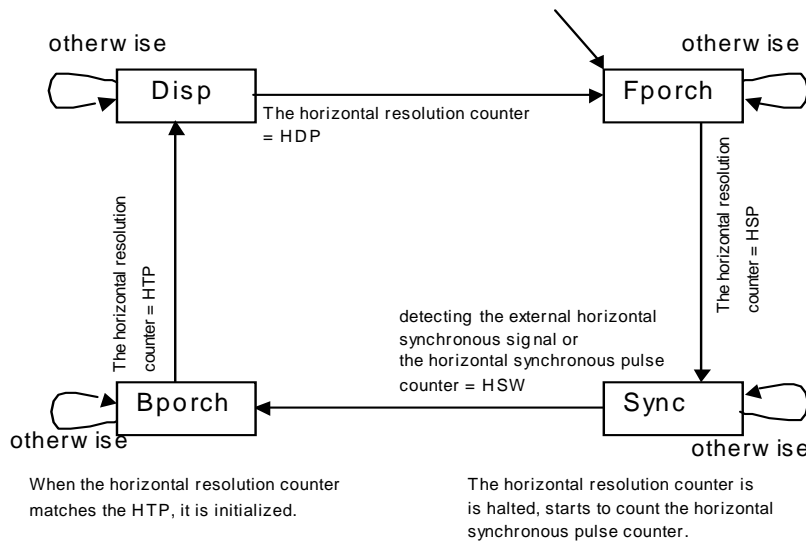
The external synchronization mode is performed by setting the ESY bit of the DCM register. In setting the external synchronization mode, HSYNC, VSYNC, and EO pin of MINT is changed to the input mode. After that it needs to be provided the synchronous signal by using the 3 state buffer from the external. When turning off the external synchronization mode, MINT internal ESY bit needs to be switched OFF after disconnecting the synchronous input signal from the external.

The buffer of the external synchronization signal must not be switched ON when the synchronous output signal of MINT is ON. Follow the previous instruction to prevent simultaneous ON from occurring.

In using the external synchronous signal with the display clock based on the internal PLL, MINT extends the clock period and fits the clock phase with the horizontal synchronous signal phase after inputting the horizontal synchronous pulse. The following caution is necessary. In case of connecting the high speed transmit signal, such as LVDS, with the digital RGB output, PLL with a built-in the high speed serial transmission is temporally unstable due to this connection. Therefore,

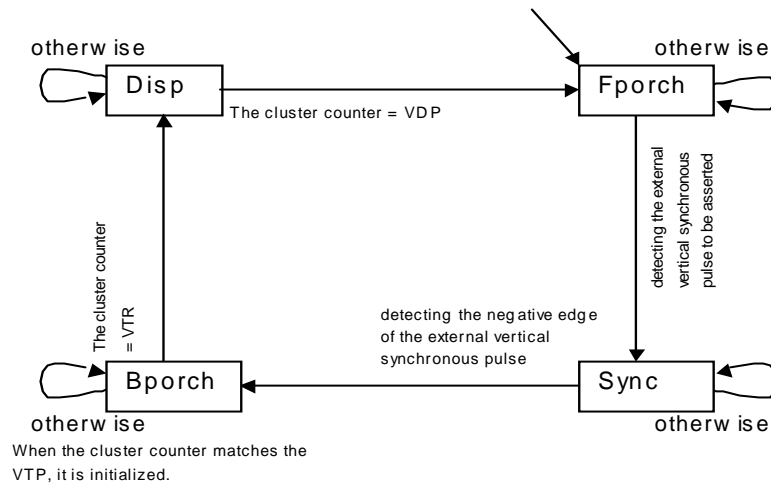
the external synchronous signal based on the internal PLL must not be used with high speed synchronous transmit signal.

The synchronization of the horizontal direction is controlled by the following state diagram.



The finite state diagram is controlled by the horizontal resolution counter. The period of outputting the signal is assigned the Disp state. When the value of the horizontal resolution counter matches that of the HDP register, it ends to output the signal and the current state is transmitted from Disp state to Fporch state (front porch). In the Fporch state, when the value of the vertical resolution register matches that of the HSP register, the current state is transmitted to the Sync state. In this state, it waits for the horizontal synchronous signal from the external. MINT detects the negative edge of the horizontal synchronous pulse from the external and synchronizes it. In detecting the horizontal synchronous signal from the external, the current state is transmitted to the Bporch state (back porch). The horizontal resolution register does not count in the Sync state, instead the horizontal synchronous counter is incremented from zero. When the value of this counter matches the setting value of the HSW register, the current state is transmitted to the Bporch state without detecting the horizontal synchronous signal from the external. When the value of the horizontal resolution counter matches that of the HTP register in the Bporch state, the horizontal resolution counter is reset, and also the current state is transmitted to the Disp state and it begins to display the next cluster.

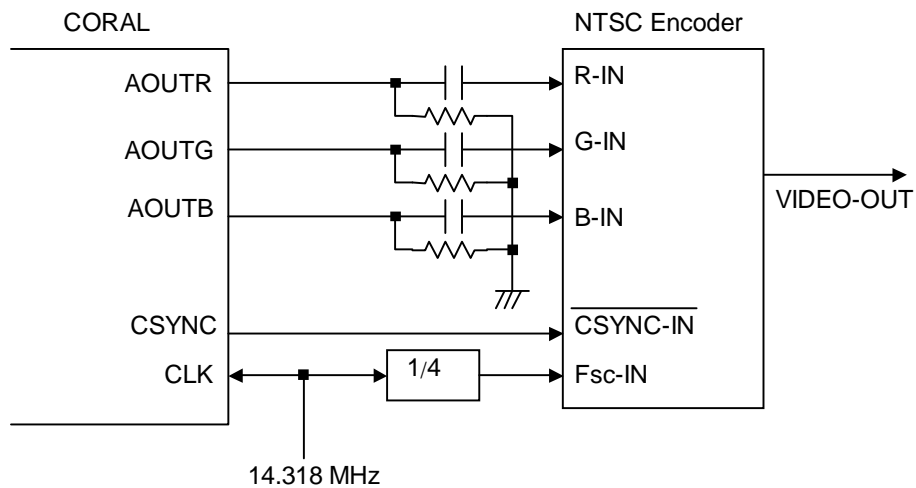
The synchronization of vertical direction is controlled by the following state diagram.



The state diagram of the vertical direction is controlled by the value of the cluster counter. The period of outputting the signal is assigned the Disp state. When the value of the cluster counter matches the value of the VDP register, it ends to output the signal and the current state is transmitted from the Disp state to the Fporch state. In the Fporch state, it waits the external synchronous pulse to be asserted. In detecting the external synchronous pulse to be asserted, the current state is transmitted to the Sync state. In the Sync state, it waits for the negative edge of the external synchronous signal. In detecting the negative edge, the current state is transmitted to the Bporch state. When the value of the cluster counter matches the values of the VTR register, the cluster counter is reset, and also the current state is transmitted to the Disp state and it starts to display the next field.

## 6.7 Video Interface, NTSC/PAL Output

In outputting NTSC/PAL signal, NTSC/PAL encoder must be connected externally as shown below:



**Fig. 5.4 Example of NTSC Encoder Connection**

Note) The neither CSYNC and VSYNC pins are impossible to output the 2.5H width signal.

## 7 Video Capture

### 7.1 Format

#### 7.1.1 Input Data Format

Input a digital video stream in the ITU RBT-656 format. NTSC and PAL signals (horizontal pixel =720) are both supported.

#### 7.1.2 Video Signal Capture

When the VIE bit of the video capture mode register (VCM) is 1, MINT is enabled to capture video stream data from the 8-bit VI pin in synchronization with the CCLK clock. Only a digital video stream conforming to ITU-RBT656 can be processed. For this reason, a Y,Cb,Cr 4:2:2 format to which timing reference codes are added is used. The video stream is captured according to the timing reference codes; MINT automatically supports both NTSC and PAL. However, to detect error codes, set NTSC/PAL in the VS bit of VCM. If NTSC is not set, reference the number of data in the capture data count register (CDCN). If PAL is not set, reference the number of data in the capture data counter register (CDCP). If the reference data does not match the stream data, bit 4 to bit 0 of the video capture status register (VCS) will be values other than 0000.

#### 7.1.3 Non-interlace Transformation

Captured video graphics can be displayed in non-interlaced format. Two modes (BOB and WEAVE) can be selected at non-interlace transformation.

##### - BOB Mode

In odd fields, the even-field rasters generated by average interpolation are added to produce one frame. In even fields, the odd-field rasters generated by average interpolation are added to produce one frame.

The BOB mode is selected by enabling vertical interpolation with the VI bit of the video capture mode register (VCM) and setting the L1IM bit of the L1-layer mode register (L1M) to 0.

##### - WEAVE Mode

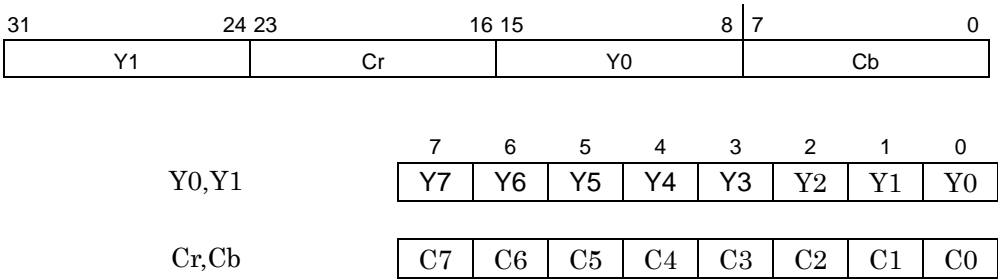
Odd and even fields are merged in the video capture buffer to produce one frame. Vertical resolutions in the WEAVE mode are higher than those in the BOB mode but raster dislocation appears at moving places.

The WEAVE mode is selected by disabling vertical interpolation with the VI bit of VCM and setting the L1IM bit of L1-layer mode register(L1M) to 1.

7.2 Video Buffer

7.2.1 Data Format

Captured graphics are stored in memory in the 16-bit/pixel YcbCr format. Video data is transformed to the RGB format when it is displayed in the L1-layer.



7.2.2 Synchronous Control

Video graphics data is written to scan-independent memory for display. Memory for video capture is controlled by the ring buffer method. When graphics data for one frame is ready in memory, the frame is displayed.

If the video capture frame rate is different from the display frame rate, a frame is omitted or the same frame is displayed continuously.

7.2.3 Area Allocation

Allocate an area of about 2.2 frames to the video capture buffer. The size of this area is equivalent to the size that considers the margin equivalent to the double buffer of the frame. Set the starting address and upper-limit address of the area in the CBOA/CBLA registers. Here, specify the raster start position as the upper-limit address.

To allocate n rasters as the video capture buffer, set the upper-limit value as follows:

CBLA = CBOA + 64n X CBS

If CBLA does not match the head of a raster, video capture data is written beyond the upper limit by only 1 raster (max.). Note that if other meaningful data is held in the area, the user-intended operation is hindered by overwriting.

For reduced display, allocate the buffer area of the reduced frame size.

### 7.2.4 Window Display

The L1 layer is used to display the captured video graphics. A part or the whole of the captured graphics can be displayed as the full screen or as a window.

To capture and display video graphics, set the L1 layer to the capture synchronous mode (L1CS = 1). In the capture synchronous mode, the L1 layer displays the latest frame in the video capture buffer. The display addresses used in the normal mode are ignored.

The stride of the L1 layer must match that of the video capture buffer. If they do not match, the displayed graphics have oblique distortion.

Match the display size of the L1 layer with the reduced graphics size of the video capture. Setting the display size of the L1 layer larger than the capture image size causes display of invalid data.

The L1 layer supports selection of the RGB display format and YcbCr display format. To capture video graphics, select the YcbCr display format (WYC = 1).

### 7.2.5 Interlace Display

The graphics captured in the video capture buffer in the WEAVE mode can be displayed in interlace. Interlace display setting is the same as WEAVE mode setting. Select 'Interlace & video display' for display scan.

Flicker appears in moving video graphics. To prevent flicker, set the OO (Odd Only) bit of the capture buffer mode register (CBM) to "1".

## 7.3 Scaling

### 7.3.1 Video Reduction Function

When the CM bits of the video capture mode register (VCM) are 11, MINT reduces the video screen size. The reduction can be set independently in the vertical and horizontal scales. The reduction is set per line in the vertical direction and in 2-pixel units in the horizontal direction. The scale setting value is defined by an input/output value. It is a 16-bit fixed fraction where the integer is represented by 5 bits and the fraction is represented by 11 bits. Valid setting values are from 0800<sub>H</sub> to FFFF<sub>H</sub>. Set the vertical direction at bit 31 to bit 16 of the capture scale register (CSC) and the horizontal direction at bits 15 to bit 00. The initial value for this register is 08000800<sub>H</sub> (once). An example of the expressions for setting a reduction in the vertical and horizontal directions is shown below.

Reduction in vertical direction	576 → 490 lines	$576/490 = 1.176$
	$1.176 \times 2048 = 2408$	→ 0968 <sub>H</sub>
Reduction in horizontal direction	720 → 648 pixels	$720/648 = 1.111$
	$1.111 \times 2048 = 2275$	→ 08E3 <sub>H</sub>

Therefore, 096808E3<sub>H</sub> is set in CSC.

The capture horizontal pixel register (CHP) and capture vertical pixel register (CVP) are used to limit the number of pixels processed during scaling. They are not used to set scaling values. Clamp processing is performed on the video streaming data outside the values set in CHP and CVP. Usually, the defaults for these registers are used.

### 7.3.2 Vertical Interpolation

When the VI bit of the video capture mode register (VCM) is “0”, data in the same field is used to interpolate the interlace screen vertically. The interlace screen is doubled in the vertical direction. When the VI bit is “1”, the interlace screen is not interpolated vertically.

## 7.4 Error Handling

### 7.4.1 Error Detection Function

If an expected control code is not detected in the input video stream, an error occurs. If an error occurs, the status is returned to the register.





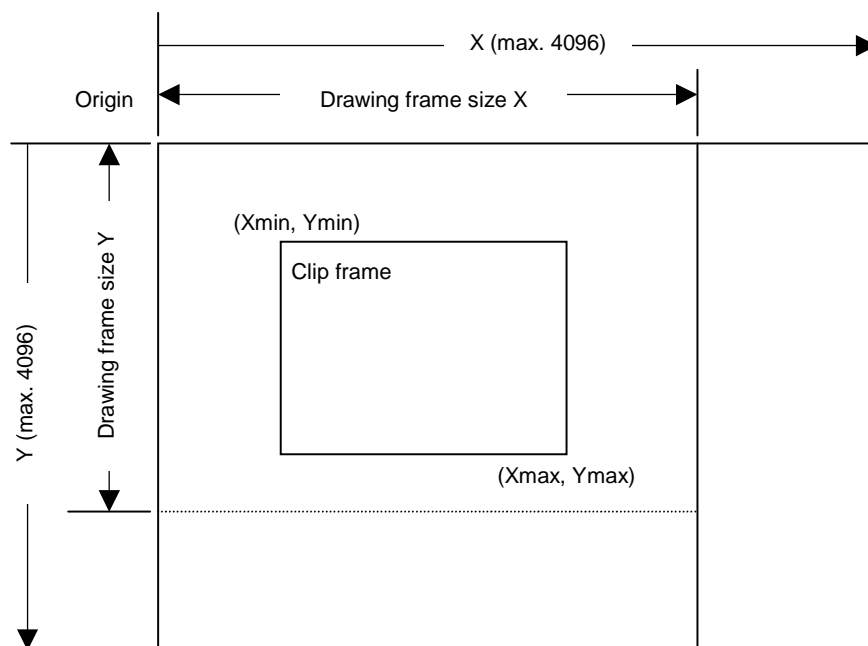
## 8 DRAWING PROCESSING

### 8.1 Coordinate System

#### 8.1.1 Drawing coordinates

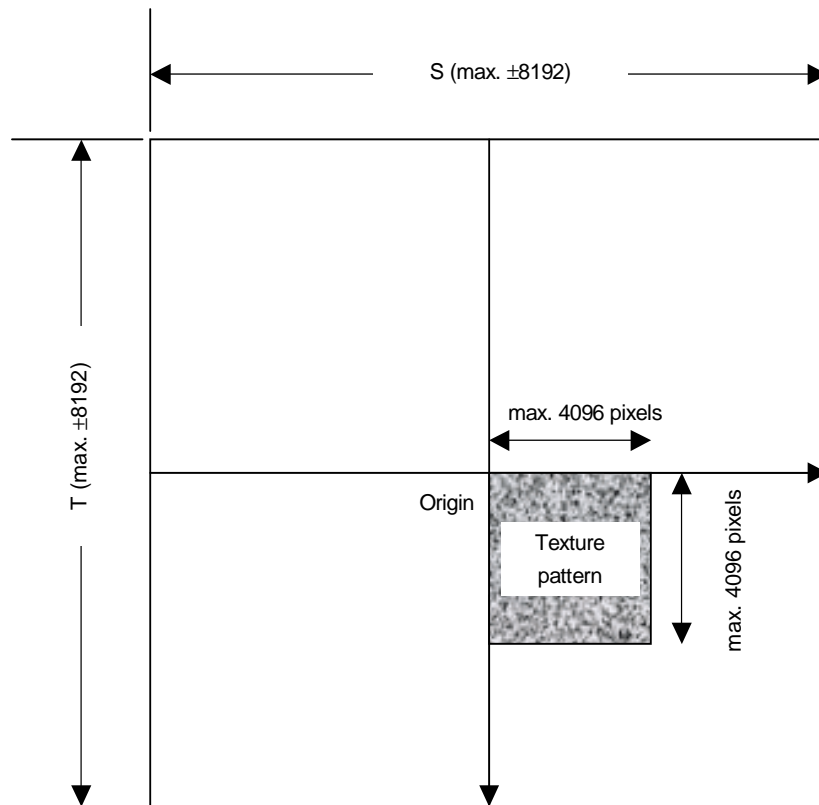
MINT GDC draws data in the drawing frame in the graphics memory that finally uses the drawing coordinates (device coordinates).

Drawing frame is treated as 2D coordinates with the origin at the top left as shown in the figure below. The maximum coordinates is  $4096 \times 4096$ . Each drawing frame is located in the Graphics Memory by setting the address of the origin and resolution of X direction (size). Although the size of Y direction does not need to be set, Y coordinates which are max. at drawing must not be overlapped with other area. In addition, at drawing, specifying the clip frame (top left and bottom right coordinates) can prevent the drawing of images outside the clip frame.



### 8.1.2 Texture coordinates

Texture coordinate is a 2D coordinate system represented as S and T (S: horizontal, T: vertical). Any integer in a range of  $-8192$  to  $+8191$  can be used as the S and T coordinates. The texture coordinates is correlated to the 2D coordinates of a vertex. One texture pattern can be applied to up to  $4096 \times 4096$  pixels. The pattern size is set in the register. When the S and T coordinates exceed the maximum pattern size, the repeat, clamp or border color option is selected.



### 8.1.3 Frame buffer

For drawing, the following area must be assigned to the Graphics Memory. The frame size (count of pixels on X direction) is common for these areas.

#### Drawing frame

The results of drawing are stored in the graphical image data area. Both the direct and indirect color mode are applicable.

#### Z buffer (Optional function)

Z buffer is required for eliminating hidden surfaces. In 16 bits mode, 2 bytes and in 8 bits mode, 1 byte are required per 1 pixel.

#### Polygon drawing flag buffer

This area is used for polygon drawing. 1 bit is required per 1 pixel.

## 8.2 Figure Drawing

### 8.2.1 Drawing primitives

MINT GDC is supported the rendering command that is compatible with the MB86290A. The following types of figure drawing primitives are compatible with the MB86290A.

Point

Line

Triangle

2DLine with XY setup

2DTriangle with XY setup

Polygon

### 8.2.2 Polygon drawing function

An irregular polygon (including concave shape) is drawn by hardware in the following manner:

1. Execute PolygonBegin command.

Initialize polygon drawing hardware.

2. Draw vertices.

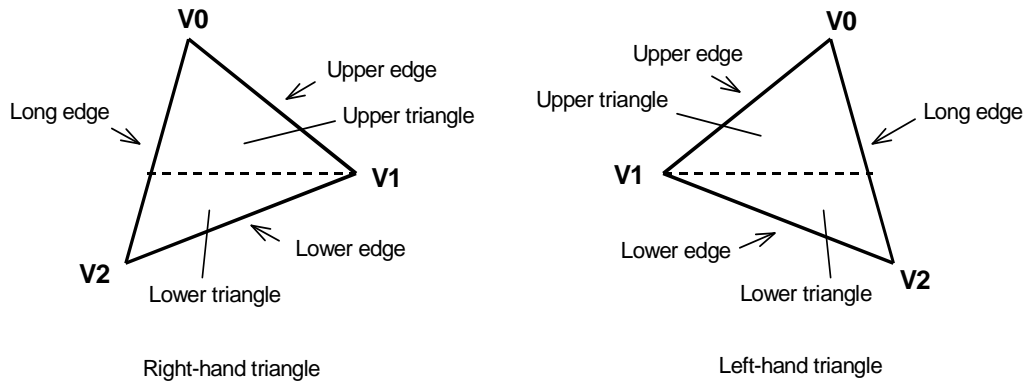
Draw outline of polygon and plot all vertices to polygon draw flag buffer using 2D Triangle with XY setup.

3. Execute PolygonEnd command.

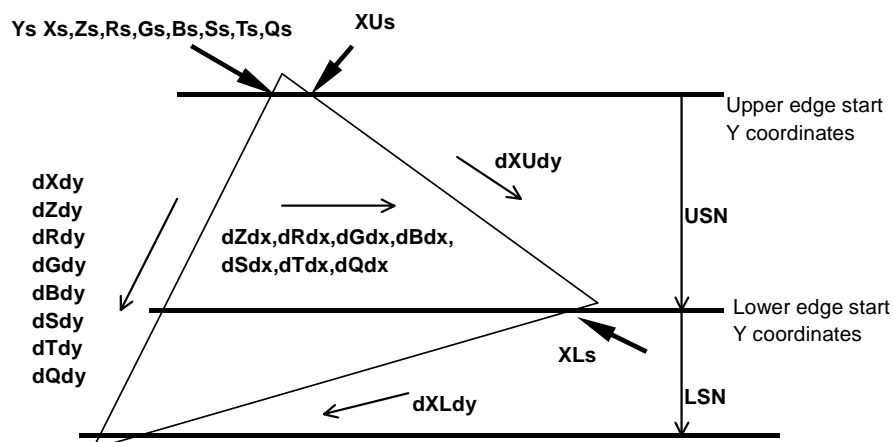
Copy shape in polygon draw flag buffer to drawing frame and fill shape with color or specified tiling pattern.

### 8.2.3 Drawing parameters

The triangles (Right triangle and Left triangle) are distinguished according to the locations of three vertices as follows (not used for 2D Triangle with XY setup):



The following parameters are required for drawing triangles (for 2D Triangle with XY setup, X and Y coordinates of each vertex are specified).



Note: Be careful about the positional relationship between coordinates Xs, XUs, and XLs.  
For example, in the above diagram, when a right-hand triangle is drawn using the parameter that shows the coordinates positional relationship Xs (upper edge start Y coordinates) > XUs or Xs (lower edge start Y coordinates) > XLs, the appropriate picture may not be drawn.

Ys	Y coordinates start position of long edge in drawing triangle
Xs	X coordinates start position of long edge corresponding to Ys
XUs	X coordinates start position of upper edge
XLs	X coordinates start position of lower edge
Zs	Z coordinates start position of long edge corresponding to Ys
Rs	R color value of long edge corresponding to Ys
Gs	G color value of long edge corresponding to Ys
Bs	B color value of long edge corresponding to Ys
Ss	S coordinate of textures of long edge corresponding to Ys
Ts	T coordinate of textures of long edge corresponding to Ys
Qs	Q perspective correction value of texture of long edge corresponding to Ys
dXdY	X DDA value of long edge direction
dXUdy	X DDA value of upper edge direction
dXLdy	X DDA value of lower edge direction
dZdy	Z DDA value of long edge direction
dRdy	R DDA value of long edge direction
dGdy	G DDA value of long edge direction
dBdy	B DDA value of long edge direction
dSdy	S DDA value of long edge direction
dTdy	T DDA value of long edge direction
dQdy	Q DDA value of long edge direction
USN	Count of spans of upper triangle
LSN	Count of spans of lower triangle
dZdx	Z DDA value of horizontal direction
dRdx	R DDA value of horizontal direction
dGdx	G DDA value of horizontal direction
dBdx	B DDA value of horizontal direction
dSdx	S DDA value of horizontal direction
dTdx	T DDA value of horizontal direction
dQdx	Q DDA value of horizontal direction

#### 8.2.4 Anti-aliasing function

MINT GDC performs anti-aliasing to make jaggies less noticeable and smooth on line edges. To use this function at the edges of primitives, redraw the primitive edges with anti-alias lines.

- (The edge of line is blended with a frame buffer color at that time. Ideally please draw sequentially from father object.)

## 8.3 Bit Map Processing

### 8.3.1 BLT

A rectangular shape in pixel units can be transferred. There are following types of transfer:

1. Transfer from host CPU to Drawing frame memory
2. Transfer between Graphics Memories including Drawing frame

Concerning 1 and 2 above, 2-term logic operation is performed between source and destination data and its result can be stored.

Setting a transparent color enables a drawing of a specific pixel with transmission.

If part of the source and destination of the BLT field are physically overlapped in the display frame, the start address (from which vertex the BLT field to be transferred) must be set correctly.

### 8.3.2 Pattern data format

MINT GDC can handle three bit map data formats: indirect color mode (8 bits/pixel), direct color mode (16 bits/pixel), and binary bit map (1 bit/pixel).

The binary bit map is used for character/font patterns, where foreground color is used for bitmap = 1 pixel, and background color (background color can be set to be transparent by setting) is applied for bitmap = 0 pixels.

## 8.4 Texture Mapping

### 8.4.1 Texture size

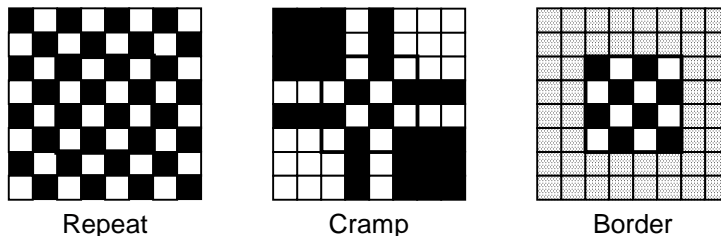
MINT GDC reads texel corresponding to the specified texture coordinates (S, T), and draws that data at the correlated pixel position of the polygon. For the S and T coordinates, the selectable texture data size is any value in the range from 4 to 4096 pixels represented as an exponent of 2.

### 8.4.2 Texture color

Drawing of 8-/16-bit direct color is supported for the texture pattern. For drawing 8-bit direct color, only point sampling can be specified for texture interpolation; only de-curl can be specified for the blend mode.

### 8.4.3 Texture Wrapping

If a negative or larger than the specified texture pattern size is specified as the texture coordinates (S, T), according to the setting, one of these options (repeat, cramp or border) is selected for the 'out-of-range' texture mapping. The mapping image for each case is shown below:



#### Repeat

This just simply masks the upper bits of the applied (S, T) coordinates. When the texture pattern size is  $64 \times 64$  pixels, the lower 6 bits of the integer part of (S, T) coordinates are used for S and T coordinates.

#### Cramp

When the applied (S, T) coordinates is either negative or larger than the specified texture pattern size, cramp the (S, T) coordinate as follows instead of texture:

$S < 0$	$S = 0$
$S > \text{Texture X size} - 1$	$S = \text{Texture X size} - 1$

#### Border

When the applied (S, T) coordinate is either negative or larger than the specified texture pattern size, the outside of the specified texture pattern is rendered in the 'border' color.

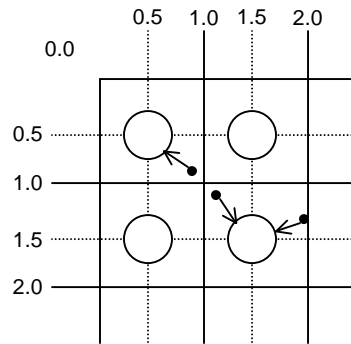


## 8.4.4 Filtering

MINT GDC supports two texture filtering modes: point filtering, and bi-linear filtering.

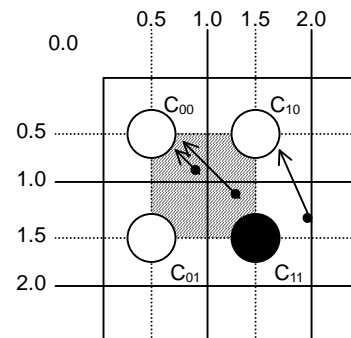
### Point filtering

This mode uses the texture pixel specified by the (S, T) coordinates as they are for drawing. The nearest pixel in the texture pattern is chosen according to the calculated (S, T) coordinates.



### Bi-linear filtering

The four nearest pixels specified with (S, T) coordinate are blended according to the distance from specified point and used in drawing.



## 8.4.5 Texture blending

MINT GDC supports the following three blend modes for texture mapping:

### Decal

This mode displays the selected texture pixel color regardless of the polygon color.

### Modulate

This mode multiplies the native polygon color ( $C_P$ ) and selected texture pixel color ( $C_T$ ) and the result is used for drawing. Rendering color is calculated as follows ( $C_O$ ):

$$C_O = C_T \times C_P$$

### Stencil

This mode selects the display color from the texture color with MSB as a flag.

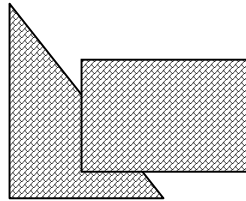
MSB = 1: Texture color

MSB = 0: Polygon color

## 8.5 Rendering

### 8.5.1 Tiling

Tiling reads the pixel color from the correlated tiling pattern and maps it onto the polygon. The tiling determines the pixel on the pattern read by pixel coordinates to be drawn, irrespective of position and size of primitive. Since the tiling pattern is stored in the texture memory, this function and texture mapping cannot be used at the same time. Also, the tiling pattern size is limited to within  $64 \times 64$  pixels. (at 16-bit color)



Example of Tiling

### 8.5.2 Alpha blending

Alpha blending blends the drawn in frame buffer to-be-drawn pixel or pixel already according to the alpha value set in the alpha register. This function cannot be used simultaneously with logic operation drawing. It can be used only when the direct color mode (16 bits/pixel) is used. The blended color  $C$  is calculated as shown below when the color of the pixel to be drawn is  $C_P$ , the color of frame buffer is  $C_F$ , and the alpha value is  $A$ :

$$C = C_P \times A + (1-A) \times C_F$$

The alpha value is specified as 8-bit data. 00h means alpha value 0% and FFh means alpha value 100%. When the texture mapping function is enabled, the following blending modes can be selected:

#### Normal

Blends post texture mapping color with frame buffer color

#### Stencil

Uses MSB of texel color for ON/OFF control:

MSB = 1: Texel color

MSB = 0: Frame buffer color

#### Stencil alpha

Uses MSB of texel color for  $\alpha$ /OFF control:

MSB = 1: Alpha blend texel color and current frame buffer color

MSB = 0: Frame buffer color

Note: MSB of frame buffer is drawn MSB of texel in both stencil and stencil alpha mode.

Therefore in case MSB of texel is MSB=0, a color of frame buffer is frame buffer, but MSB of frame buffer is set to 0.

### 8.5.3 Logic operation

This mode executes a logic operation between the pixel to be drawn and the one already drawn in frame buffer and its result is drawn. Alpha blending cannot be used when this function is specified.

Type	ID	Operation	Type	ID	Operation
CLEAR	0000	0	AND	0001	S & D
COPY	0011	S	OR	0111	S   D
NOP	0101	D	NAND	1110	!(S & D)
SET	1111	1	NOR	1000	!(S   D)
COPY INVERTED	1100	!S	XOR	0110	S xor D
INVERT	1010	!D	EQUIV	1001	!(S xor D)
AND REVERSE	0010	S & !D	AND INVERTED	0100	!S & D
OR REVERSE	1011	S   !D	OR INVERTED	1101	!S   D

### 8.5.4 Hidden plane management (Optional function)

MINT GDC supports the Z buffer for hidden plane management.

This function compares the Z value of a new pixel to be drawn and the existing Z value in the Z buffer. Display/not display is switched according to the Z-compare mode setting. Define the Z-buffer access options in the ZWRITEMASK mode.

The Z compare operation type is determined by the Z compare mode.

Either 16 or 8 bits can be selected for the Z-value.

<b>ZWRITEMASK</b>	1	Compare Z values, no Z value write overwrite
	0	Compare Z values, Z value write

Z Compare mode	Code	Condition
NEVER	000	Never draw
ALWAYS	001	Always draw
LESS	010	Draw if pixel Z value < current Z buffer value
LEQUAL	011	Draw if pixel Z value ≤ current Z buffer value
EQUAL	100	Draw if pixel Z value = current Z buffer value
GEQUAL	101	Draw if pixel Z value ≥ current Z buffer value
GREATER	110	Draw if pixel Z value > current Z buffer value
NOTEQUAL	111	Draw if pixel Z value != current Z buffer value

## 8.6 Drawing Attributes

### 8.6.1 Line drawing attributes

In drawing lines, the following attributes apply:

**Line Drawing Attributes**

Drawing Attribute	Description
Line width	Line width selectable in range of 1 to 32 pixels
Broken line	Specify broken line pattern in 32-bit data
Anti-alias	Line edge smoothed when anti-aliasing enabled

### 8.6.2 Triangle drawing attributes

In drawing triangles, the following attributes apply (these attributes are disabled in 2DTriangle with XY setup). Texture mapping and tiling have separated texture attributes:

**Triangle Drawing Attributes**

Drawing Attribute	Description
Shading	Gouraud shading or flat shading selectable
Alpha blending	Set alpha blending enable/disable per polygon
Alpha blending coefficient	Set color blending ratio of alpha blending

### 8.6.3 Texture attributes

In texture mapping, the following attributes apply:

**Texture Attributes**

Drawing Attribute	Description
Texture mode	Select either texture mapping or tiling
Texture filter	Select either point sampling or bi-linear filtering
Texture coordinates correction	Select either linear or perspective correction
Texture wrap	Select either repeat or clamp of texture pattern
Texture blend mode	Select either decal or modulate

### 8.6.4 BLT attributes

In BLT drawing, the following attributes apply:

**BLT Attributes**

Drawing Attribute	Description
Logic operation mode	Specify two source logic operation mode
Transparency mode	Set transparent copy mode and transparent color
Alpha map mode	Blend a color according to alpha map

### 8.6.5 Character pattern drawing attributes

**Character Pattern Drawing**

Drawing Attribute	Description
Character pattern enlarge/shrink	Vertical and Horizontal $\times 2$ , Horizontal $\times 2$ , Vertical and Horizontal $\times 1/2$ , Horizontal $\times 1/2$
Character pattern color	Set character color and background color
Transparency/non-transparency	Set background color to transparency/non-transparency

## 9 DISPLAY LIST

### 9.1 Overview

Display list is a set of display list commands, parameters and pattern data. All display list commands stored in a display list are executed consequently.

The display list is transferred to the display list FIFO by the following method:

Transfer from graphics memory to display FIFO by register setting

Display list Command-1
Data 1-1
Data 1-2
Data 1-3
Display list Command-2
Data 2-1
Data 2-2
Data 2-3
...

**Display List**

### 9.1.1 Header format

The format of the display list header is shown below.

## Format List

[illegible]

### Description of Each Field

Type	Display list type
Command	Command
Count	Count of data excluding header
Address	Address value used at data transfer
Vertex	Vertex number
Flag	Attribute flag peculiar to display list command

### Vertex Number Specified in Vertex Code

Vertex	Vertex number (Line)	Vertex number (Triangle)
00	V0	V0
01	V1	V1
10	Setting prohibited	V2
11	Setting prohibited	Setting prohibited

## 9.2 Rendering Command

### 9.2.1 Command list

The following table lists MINT GDC rendering commands and their command codes.

Type	Command	Description
Nop	—	No operation
Interrupt	—	Interrupt request to host CPU
Sync	—	Synchronization with events
SetRegister	—	Sets data to register
SetVertex2i	Normal	Sets data to 2D Triangle with XY setup vertex register
	PolygonBegin	Initializes border rectangle calculation of multiple vertices random shape
Draw	PolygonEnd	Clears polygon flag after drawing polygon
	Flush_FB/Z	Flushes drawing pipelines
DrawPixel	Pixel	Draws point
DrawPixelZ	PixelZ	Draws point with Z
DrawLine	Xvector	Draws line (principal axis X)
	Yvector	Draws line (principal axis Y)
	AntiXvector	Draws line with anti-alias option (principal axis X)
	AntiYvector	Draws line with anti-alias option (principal axis Y)
DrawLine2i	ZeroVector	Draws 2D Line with XY setup (with vertex 0 as starting point)
DrawLine2iP	OneVector	Draws 2D Line with XY setup (with vertex 1 as starting point)
DrawTrap	TrapRight	Draws right triangle
	TrapLeft	Draws left triangle
DrawVertex2i	TriangleFan	Draws 2D Triangle with XY setup
DrawVertex2iP	FlagTriangleFan	Draws 2D Triangle with XY setup for multiple vertices random shape
DrawRectP	BltFill	Draws rectangle with single color
	ClearPolyFlag	Clears polygon flag buffer
DrawBitmapP	BltDraw	Draws Blt (16-bit)
	Bitmap	Draws binary bit map (character)
DrawBitmapLargeP	BltDraw	Draws Blt (32-bit)
BltCopyP BltCopy-AlternateP	TopLeft	Blt transfer from top left coordinates
	TopRight	Blt transfer from top right coordinates
	BottomLeft	Blt transfer from bottom left coordinates
	BottomRight	Blt transfer from bottom right coordinates
LoadTextureP	LoadTexture	Loads texture pattern
	LoadTILE	Loads tile pattern
BltTextureP	LoadTexture	Loads texture pattern from local memory
	LoadTILE	Loads tile pattern from local memory
BltCopyAlt-AlphaBlendP	—	Alpha blending is supported (see the alpha map). BltCopyAlternateP



**Type Code Table**

Type	Code
DrawPixel	0000_0000
DrawPixelZ	0000_0001
DrawLine	0000_0010
DrawLine2i	0000_0011
DrawLine2iP	0000_0100
DrawTrap	0000_0101
DrawVertex2i	0000_0110
DrawVertex2iP	0000_0111
DrawRectP	0000_1001
DrawBitmapP	0000_1011
BitCopyP	0000_1101
BitCopyAlternateP	0000_1111
LoadTextureP	0001_0001
BltTextureP	0001_0011
BltCopyAltAlphaBlendP	0001_1111
SetVertex2i	0111_0000
SetVertex2iP	0111_0001
Draw	1111_0000
SetRegister	1111_0001
Sync	1111_1100
Interrupt	1111_1101
Nop	1111_1111

**Command Code Table (1)**

Command	Code
Pixel	000_00000
PixelZ	000_00001
Xvector	001_00000
Yvector	001_00001
XvectorNoEnd	001_00010
YvectorNoEnd	001_00011
XvectorBlpClear	001_00100
YvectorBlpClear	001_00101
XvectorNoEndBlpClear	001_00110
YvectorNoEndBlpClear	001_00111
AntiXvector	001_01000
AntiYvector	001_01001
AntiXvectorNoEnd	001_01010
AntiYvectorNoEnd	001_01011
AntiXvectorBlpClear	001_01100
AntiYvectorBlpClear	001_01101
AntiXvectorNoEndBlpClear	001_01110
AntiYvectorNoEndBlpClear	001_01111
ZeroVector	001_10000
Onevector	001_10001
ZeroVectorNoEnd	001_10010
OnevectorNoEnd	001_10011
ZeroVectorBlpClear	001_10100
OnevectorBlpClear	001_10101
ZeroVectorNoEndBlpClear	001_10110
OnevectorNoEndBlpClear	001_10111
AntiZeroVector	001_11000
AntiOnevector	001_11001
AntiZeroVectorNoEnd	001_11010
AntiOnevectorNoEnd	001_11011
AntiZeroVectorBlpClear	001_11100
AntiOnevectorBlpClear	001_11101
AntiZeroVectorNoEndBlpClear	001_11110
AntiOnevectorNoEndBlpClear	001_11111

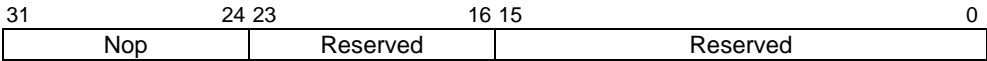
**Command Code Table (2)**

Command	Code
BlitFill	010_00001
BlitDraw	010_00010
Bitmap	010_00011
TopLeft	010_00100
TopRight	010_00101
BottomLeft	010_00110
BottomRight	010_00111
LoadTexture	010_01000
LoadTILE	010_01001
TrapRight	011_00000
TrapLeft	011_00001
TriangleFan	011_00010
FlagTriangleFan	011_00011
Flush_FB	110_00001
Reserved	110_00010
PolygonBegin	111_00000
PolygonEnd	111_00001
ClearPolyFlag	111_00010
Normal	111_11111

9.2.2 Details of rendering commands

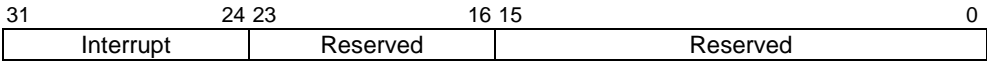
All parameters belonging to their command are stored in relevant registers. The definition of each parameter is explained in the section of each command.

Nop (Format1)



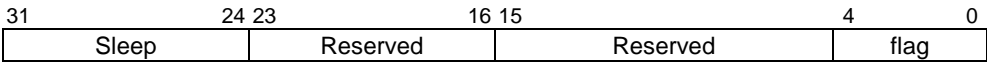
No operation

Interrupt (Format1)



The **Interrupt** command generates interrupt request to host CPU.

Sync (Format9)



The **Sync** command suspends all subsequent display list processing until event set in flag detected.

Flag:

Bit number	4	3	2	1	0
Bit field name	Reserved	Reserved	Reserved	Reserved	VBANK

- Bit 0
- VBANK

VBANK Synchronization

0No operation

1Wait for VSYNC detection

### SetRegister (Format2)

31	24 23	16 15	0
SetRegister	Count	Address	
		(Val 0)	
		(Val 1)	
		...	
		(Val n)	

The **SetRegister** command sets data to sequential registers.

Count: Data word count (in double-word unit)

Address: Register address

Set the value of the address for **SetRegister** given in the register list.

When transferring two or more data, set the starting register address.

### SetVertex2i (Format8)

31	24 23	16 15	4 3 2 1 0
SetVertex2i	Command	Reserved	flag vertex
	Xdc		
	Ydc		

The **SetVertex2i** command sets vertices data for 2D Line with XY setup or 2D Triangle with XY setup to registers.

Commands:

Normal Sets vertex data (X, Y).

PolygonBegin Starts calculation of circumscribed rectangle for random shape to be drawn. Calculate vertices of rectangle including all vertices of random shape defined between **PolygonBegin** and **PolygonEnd**.

Flag: Not used

### SetVertex2iP (Format8)

31	24 23	16 15	4 3 2 1 0
SetVertex2i	Command	Reserved	flag vertex
	Ydc	Xdc	

The **SetVertex2iP** command sets vertices data for 2D Line with XY setup or 2D Triangle with XY setup to registers.

Only the integer (packed format) can be used to specify these vertices.

Commands:

Normal Sets vertices data.

PolygonBegin Starts calculation of circumscribed rectangle of random shape to be drawn. Calculate vertices of rectangle including all vertices of random shape defined between **PolygonBegin** and **PolygonEnd**.

Flag: Not used

## Draw (Format5)

31	24 23	16 15	0
Draw	Command	Reserved	

The **Draw** command executes drawing command. All parameters required for drawing command execution must be set at their appropriate registers.

Commands:

PolygonEnd	Draws polygon end. Fills random shape with color according to flags generated by <b>FlagTriangleFan</b> command and information of circumscribed rectangle generated by <b>PolygonBegin</b> command.
Flush_FB	Flushes drawing data in the drawing pipeline into the graphics memory. Place this command at the end of the display list.
Flush_Z	Flushes Z value data in the drawing pipeline into the graphics memory. When using the Z buffer, place this command together with the <b>Flush_FB</b> command at the end of the display list.

## DrawPixel (Format5)

31	24 23	16 15	0
DeawPixel	Command	Reserved	
PXs			
PYs			

The **DrawPixel** command draws pixel.

Command:

Pixel	Draws pixel without Z value.
-------	------------------------------

## DrawPixelZ (Format5)

31	24 23	16 15	0
DeawPixel	Command	Reserved	
PXs			
PYs			
PZs			

The **DrawPixelZ** command draws pixel with Z value.

Command:

PixelZ	Draws pixel with Z value.
--------	---------------------------

## DrawLine (Format5)

31	24 23	16 15	0
DrawLine	Command	Reserved	
LPN			
LXs			
LXde			
LYs			
LYde			

The **DrawLine** command draws line. It starts drawing after setting all parameters at line draw registers.

### Commands:

Xvector	Draws line (principal axis X).
Yvector	Draws line (principal axis Y).
XvectorNoEnd	Draws line (principal axis X, and without end point drawing).
YvectorNoEnd	Draws line (principal axis Y, and without end point drawing).
XvectorBlpClear	Draws line (principal axis X, and prior to drawing, broken line pattern reference position cleared).
YvectorBlpClear	Draws line (principal axis Y, and prior to drawing, broken line pattern reference position cleared).
XvectorNoEndBlpClear	Draws line (principal axis X, without end point drawing and prior to drawing, broken line pattern reference position cleared).
YvectorNoEndBlpClear	Draws line (principal axis Y, without end point drawing and prior to drawing, broken line pattern reference position cleared).
AntiXvector	Draws anti-alias line (principal axis X).
AntiYvector	Draws anti-alias line (principal axis Y).
AntiXvectorNoEnd	Draws anti-alias line (principal axis X, and without end point drawing).
AntiYvectorNoEnd	Draws anti-alias line (principal axis Y, and without end point drawing).
AntiXvectorBlpClear	Draws anti-alias line (principal axis X and prior to drawing, broken line pattern reference position cleared).
AntiYvectorBlpClear	Draws anti-alias line (principal axis Y and prior to drawing, broken line pattern reference position cleared).
AntiXvectorNoEndBlpClear	Draws anti-alias line (principal axis X, without end point drawing and prior to drawing, broken line pattern reference position cleared).
AntiYvectorNoEndBlpClear	Draws anti-alias line (principal axis Y, without end point drawing and prior to drawing, broken line pattern reference position cleared).

## DrawLine2i (Format7)

31	24 23	16 15	0
DrawLine2i	Command	Reserved	vertex
	LFXs	0	
	LFYs	0	

The **DrawLine2i** command draws 2DLine with XY setup. It starts drawing after setting parameters at the 2DLine with XY setup drawing registers. Integer data can only be used for coordinates.

Commands:

ZeroVector	Draws line from vertex 0 to vertex 1.
OneVector	Draws line from vertex 1 to vertex 0.
ZeroVectorNoEnd	Draws line from vertex 0 to vertex 1 (without drawing end point).
OneVectorNoEnd	Draws line from vertex 1 to vertex 0 (without drawing end point).
ZeroVectorBlpClear	Draws line from vertex 0 to vertex 1 (principal axis X, and prior to drawing, broken line pattern reference position cleared).
OneVectorBlpClear	Draws line from vertex 1 to vertex 0 (principal axis Y, and prior to drawing, broken line pattern reference position cleared).
ZeroVectorNoEndBlpClear	Draws line from vertex 0 to vertex 1 (principal axis X, without end point drawing and prior to drawing, broken line pattern reference position cleared).
OneVectorNoEndBlpClear	Draws line from vertex 1 to vertex 0 (principal axis Y, without end point drawing and prior to drawing, broken line pattern reference position cleared).
AntiZeroVector	Draws anti-alias line from vertex 0 to vertex 1.
AntiOneVector	Draws anti-alias line from vertex 1 to vertex 0.
AntiZeroVectorNoEnd	Draws anti-alias line from vertex 0 to vertex 1 (without end point).
AntiOneVectorNoEnd	Draws anti-alias line from vertex 1 to vertex 0 (without end point).
AntiZeroVectorBlpClear	Draws anti-alias line from vertex 0 to vertex 1 (principal axis X and prior to drawing, broken line pattern reference position cleared).
AntiOneVectorBlpClear	Draws anti-alias line from vertex 1 to vertex 0 (principal axis Y and prior to drawing, broken line pattern reference position cleared).
AntiZeroVectorNoEndBlpClear	Draws anti-alias line from vertex 0 to vertex 1 (principal axis X, without end point drawing and prior to drawing, broken line pattern reference position cleared).
AntiOneVectorNoEndBlpClear	Draws anti-alias line from vertex 1 to vertex 0 (principal axis Y, without end point drawing and prior to drawing, broken line pattern reference position cleared).



## DrawLine2iP (Format7)

31	24 23	16 15	0
DrawLine2iP	Command	Reserved	vertex
LFYs		LFXs	

The **DrawLine2iP** command draws high-speed 2DLine. It starts drawing after setting parameters at high-speed 2DLine drawing registers. Only packed integer data can be used for coordinates.

Commands:

ZeroVector	Draws line from vertex 0 to vertex 1.
OneVector	Draws line from vertex 1 to vertex 0.
ZeroVectorNoEnd	Draws line from vertex 0 to vertex 1 (without drawing end point).
OneVectorNoEnd	Draws line from vertex 1 to vertex 0 (without drawing end point).
ZeroVectorBlpClear	Draws line from vertex 0 to vertex 1 (principal axis X, and prior to drawing, broken line pattern reference position cleared).
OneVectorBlpClear	Draws line from vertex 1 to vertex 0 (principal axis Y, and prior to drawing, broken line pattern reference position cleared).
ZeroVectorNoEndBlpClear	Draws line from vertex 0 to vertex 1 (principal axis X, without end point drawing and prior to drawing, broken line pattern reference position cleared).
OneVectorNoEndBlpClear	Draws line from vertex 1 to vertex 0 (principal axis Y, without end point drawing and prior to drawing, broken line pattern reference position cleared).
AntiZeroVector	Draws anti-alias line from vertex 0 to vertex 1.
AntiOneVector	Draws anti-alias line from vertex 1 to vertex 0.
AntiZeroVectorNoEnd	Draws anti-alias line from vertex 0 to vertex 1 (without end point).
AntiOneVectorNoEnd	Draws anti-alias line from vertex 1 to vertex 0 (without end point).
AntiZeroVectorBlpClear	Draws anti-alias line from vertex 0 to vertex 1 (principal axis X and prior to drawing, broken line pattern reference position cleared).
AntiOneVectorBlpClear	Draws anti-alias line from vertex 1 to vertex 0 (principal axis Y and prior to drawing, broken line pattern reference position cleared).
AntiZeroVectorNoEndBlpClear	Draws anti-alias line from vertex 0 to vertex 1 (principal axis X, without end point drawing and prior to drawing, broken line pattern reference position cleared).
AntiOneVectorNoEndBlpClear	Draws anti-alias line from vertex 1 to vertex 0 (principal axis Y, without end point drawing and prior to drawing, broken line pattern reference position cleared).

### DrawTrap (Format5)

31	24 23	16 15	0
DrawTrap	Command	Reserved	
Ys		0	
Xs			
DXdy			
XUs			
DXUdy			
XLs			
DXLdy			
USN		0	
LSN		0	

The **DrawTrap** command draws Triangle. It starts drawing after setting parameters at the Triangle Drawing registers (coordinates).

Commands:

TrapRight	Draws right triangle.
TrapLeft	Draws left triangle.

### DrawVertex2i (Format7)

31	24 23	16 15	0
DrawVertex2i	Command	Reserved	vertex
Xdc		0	
Ydc		0	

The **DrawVertex2i** command draws 2D Triangle with XY setup

It starts triangle drawing after setting parameters at 2D Triangle Drawing registers.

Commands:

TriangleFan	Draws 2D Triangle with XY setup.
FlagTriangleFan	Draws 2D Triangle with XY setup for polygon drawing in the flag buffer.

### DrawVertex2iP (Format7)

31	24 23	16 15	0
DrawVertex2iP	Command	Reserved	vertex
Ydc		Xdc	

The **DrawVertex2iP** command draws 2D Triangle with XY setup.

It starts drawing after setting parameters at 2D Triangle with XY setup Drawing registers

Only the packed integer format can be used for vertex coordinates.

Commands:

TriangleFan	Draw 2D Triangle with XY setup.
FlagTriangleFan	Draws 2D Triangle with XY setup for polygon drawing in the flag buffer.

### DrawRectP (Format5)

31	24 23	16 15	0
DrawRectP	Command	Reserved	
Rys		RXs	
RsizeY		RsizeX	

The **DrawRectP** command fills rectangle. The rectangle is filled with the current color after setting parameters at the rectangle registers. Please set XRES(X resolution) to in 8 byte units when using this command.

Commands:

BlitFill	Fills rectangle with current color (single).
ClearPolyFlag	Fills <b>polygon drawing</b> flag buffer area with 0. The size of drawing frame is defined in RsizeX,Y. Must set RXs[3:0] and RsizeX[3:0] as 0000. (16pixel aligned) Drawing clipping is not work for this command.

### DrawBitmapP (Format6)

31	24 23	16 15	0
DrawBitmapP	Command	Count	
Rys		RXs	
RsizeY		RsizeX	
(Pattern 0)			
(Pattern 1)			
...			
(Pattern n)			

The **DrawBitmapP** command draws rectangle patterns. Please set XRES(X resolution) to in 8 byte units when using this command.

Commands:

BlitDraw	Draws rectangle of 8 bits/pixel or 16 bits/pixel.
DrawBitmap	Draws binary bitmap character pattern. Bit 0 is drawn in transparent or background color, and bit 1 is drawn in foreground color.

### DrawBitmapLargeP (Format11)

31	24 23	16 15	0
DrawBitmapLargeP	Command	Reserved	
Count			
Rys		RXs	
RsizeY		RsizeX	
(Pattern 0)			
(Pattern 1)			
...			
(Pattern n)			

The **DrawBitmapP** command draws rectangle patterns.

The parameter(count field) could be used up to 32-bit(\*1) unlike DrawBitmapP.

(\*1: The data format of counter field is signed long. Thus actually it is possible to use up to 31-bit.)

Please set XRES(X resolution) to in 8 byte units when using this command.

Commands:

BlitDraw	Draws rectangle of 8 bits/pixel or 16 bits/pixel.
----------	---

### BltCopyP (Format5)

31	24 23	16 15	0
BltCopyP	Command	Reserved	
SRYs		SRXs	
DRYs		DRXs	
BRsizeY		BRsizeX	

The **BltCopyP** command copies rectangle pattern within drawing frame. Please set XRES(X resolution) to in 8 byte units when using this command.

Commands:

TopLeft	Starts BitBlt transfer from top left coordinates.
TopRight	Starts BitBlt transfer from top right coordinates.
BottomLeft	Starts BitBlt transfer from bottom left coordinates.
BottomRight	Starts BitBlt transfer from bottom right coordinates.

### BltCopyAlternateP (Format5)

31	24 23	16 15	0
BltCopyAlternateP	Command	Reserved	
SADDR			
SStride			
SRYs		SRXs	
DADDR			
DStride			
DRYs		DRXs	
BRsizeY		BRsizeX	

The **BltCopyAlternateP** command copies rectangle between two separate drawing frames.

Please set XRES(X resolution) to in 8 byte units when using this command. And please set SStride and DStride to in 8 byte units.

Command:

TopLeft	Starts BitBlt transfer from top left coordinates. Drawing clipping is not wok for this command.
---------	--

### BltCopyAltAlphaBlendP (Format5)

31	24 23	16 15	0
BltCopyAlternateP	Command	Reserved	
SADDR			
SStride			
SRYs		SRXs	
BlendStride			
BlendRYs		BlendRXs	
DRYs		DRXs	
BRsizeY		BRsizeX	

The **BltCopyAltAlphaBlendP** command performs alpha blending for the source (specified using SADDR, SStride, SRXs, SRXy) and the alpha map (specified using ABR (alpha base address), BlendStride, BlendRXs, BlendRYs) and then copies the result of the alpha blending to the destination (specified using FBR (frame buffer base address), XRES (X resolution), DRXs, and DRYs).

Please set XRES(X resolution) to in 8 byte units when using this command. And please set SStride and BlendStride to in 8 byte units.

Command:

reserved

Set 0000\_0000 to maintain future compatibility.

## 10 I2C Function

connected to SDA line via open-drain I/O cell. And this interface also has SCL input (SCLI) and SCL out Two bi-directional buses, serial data line (SDA) and serial clock line (SCL), carry information at I2C-bus. Scarlet I2C interface has SDA input (SDAI) and SDA output (SDAO) for SDA and is put (SCLO) for SCL line and is connected to SCL line via open-drain I/O cell. The wired theory is used when the interface is connected to SDA line and SCL line.

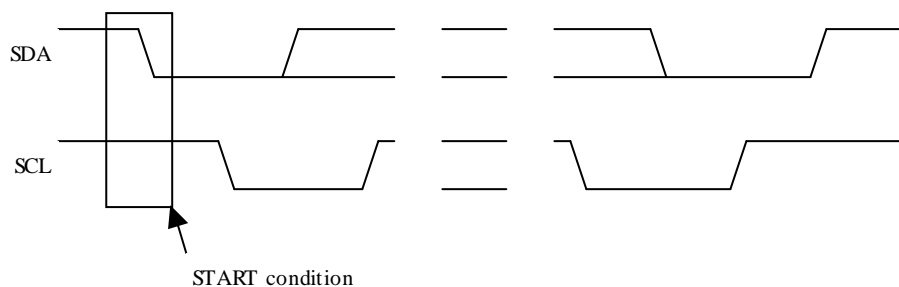
### 10.1 START condition

If "1" is written to MSS bit while the bus is free, this module will become a master mode and will generate START condition simultaneously. In a master mode, even if a bus is in a use state (BB=1), START condition can be generated again by writing "1" to SCC bit.

There are two conditions to generate START condition.

- "1" writing to MSS bit in the state where the bus is not used (MSS=0 & BB=0 & INT=0 & AL=0)
- "1" writing to SCC bit in the interruption state in a master mode (MSS=1 & BB=1 & INT=1 & AL=0)

If "1" writing is performed to MSS bit in an idol state, AL bit will be set to "1". "1" writing to MSS bit other than the above is disregarded.

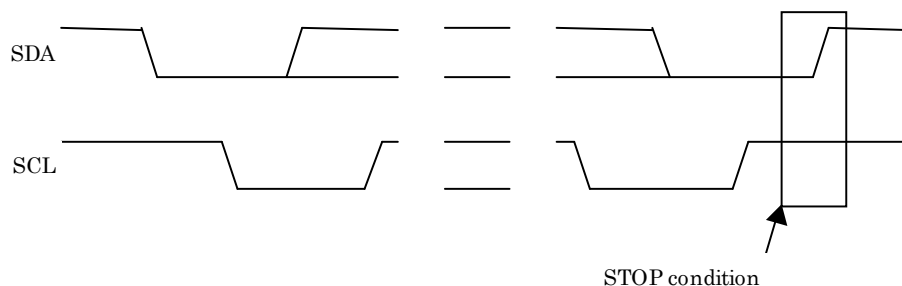


### 10.2 STOP condition

If "0" is written to MSS bit in a master mode (MSS=1), this module will generate STOP condition and will become a slave mode.

There is a condition to generate STOP condition.

- "0" writing to MSS bit in the interruption state in a master mode (MSS=1 & BB=1 & INT=1 & AL=0)
- "0" writing to MSS bit other than the above is disregarded.

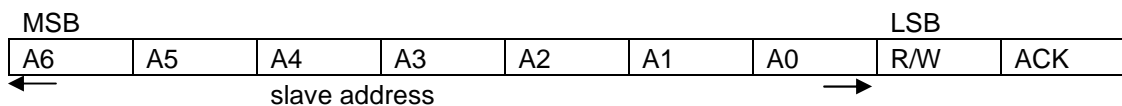


## 10.3 Addressing

In a master mode, it is set to BB="1" and TRX="0" after generation of START condition, and the contents of DAR register are output from MSB. When this module receives acknowledge after transmission of address data, the bit-0 of transmitting data (bit-0 of DRA register after transmission) is reversed and it is stored in TRX bit.

### - Transfer format of slave address

A transfer format of slave address is shown below:



### - Map of slave address

A map of slave address is shown below:

slave address	R/W	Description
0 0 0 0    0 0 0	0	General call address
0 0 0 0    0 0 0	1	START byte
0 0 0 0    0 0 1	X	CBUS address
0 0 0 0    0 1 0	X	Reserved
0 0 0 0    0 1 1	X	Reserved
0 0 0 0    1 X X	X	Reserved
0 0 0 1    X X X ⋮ 1 1 1 0    X X X	X	Available slave address
1 1 1 1    0 X X	X	10-bit slave addressing*1
1 1 1 1    1 X X	X	Reserved

\*1 This module does not support 10-bit slave address.

## 10.4 Synchronization of SCL

When two or more I2C devices turn into a master device almost simultaneously and drive SCL line, each device senses the state of SCL line and adjusts the drive timing of SCL line automatically in accordance with the timing of the latest device.

## 10.5 Arbitration

When other masters have transmitted data simultaneously at the time of master transmission, arbitration takes place. When its own transmitting data is "1" and the data on SDA line is "0", the master considers that the arbitration was lost and sets "1" to AL. And if the master is going to generate START condition while the bus is in use by other master, it will consider that arbitration was lost and will set "1" to AL.

When the START condition which other masters generated is detected by the time the master actually generated START condition, even when it checked the bus is in nonuse state and wrote in MSS="1", it considers that the arbitration was lost and sets "1" to AL.

When AL bit is set to "1", a master will set MSS="0" and TRX="0" and it will be a slave receiving mode.

When the arbitration is lost (it has no royalty of a bus), a master stops a drive of SDA.

However, a drive of SCL is not stopped until 1 byte transfer is completed and interruption is cleared.

## 10.6 Acknowledge

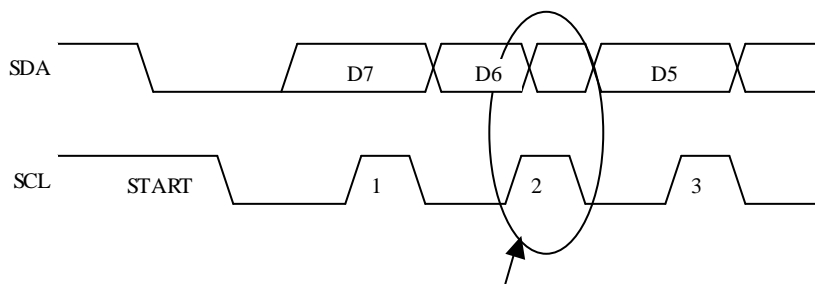
Acknowledge is transmitted from a reception side to a transmission side. At the time of data reception, acknowledge is stored in LRB bit by ACK bit.

When the acknowledge from a master reception side is not received at the time of slave transmission, it sets TRX="0" and becomes slave receiving mode. Thereby, a master can generate STOP condition when a slave opens SCL.

## 10.7 Bus error

When the following conditions are satisfied, it is judged as a bus error, and this interface will be in a stop state.

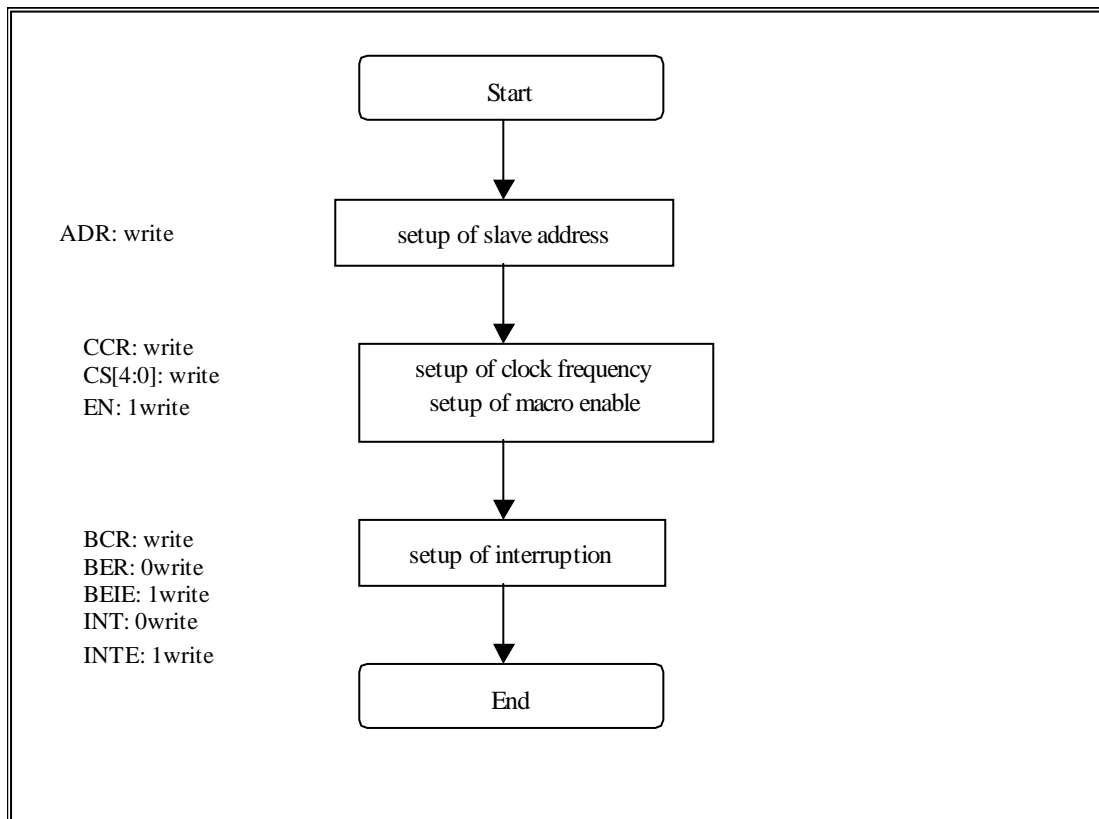
- Detection of the basic regulation violation on I2C-bus under data transfer (including ACK bit)
- Detection of STOP condition in a master mode
- Detection of the basic regulation violation on I2C-bus at the time of bus idol



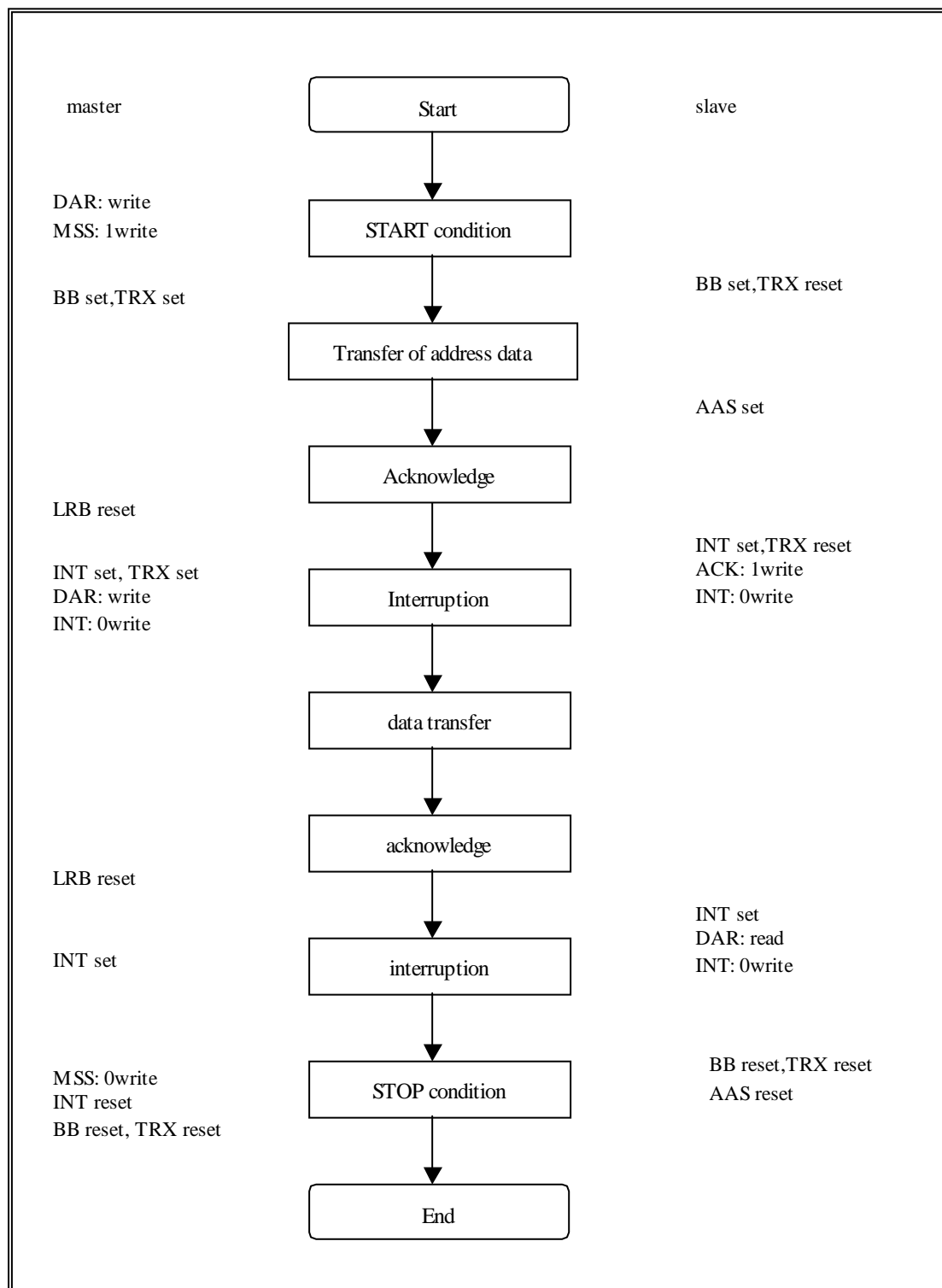
SDA changed under data transmission (SCL=H). It becomes bus error.



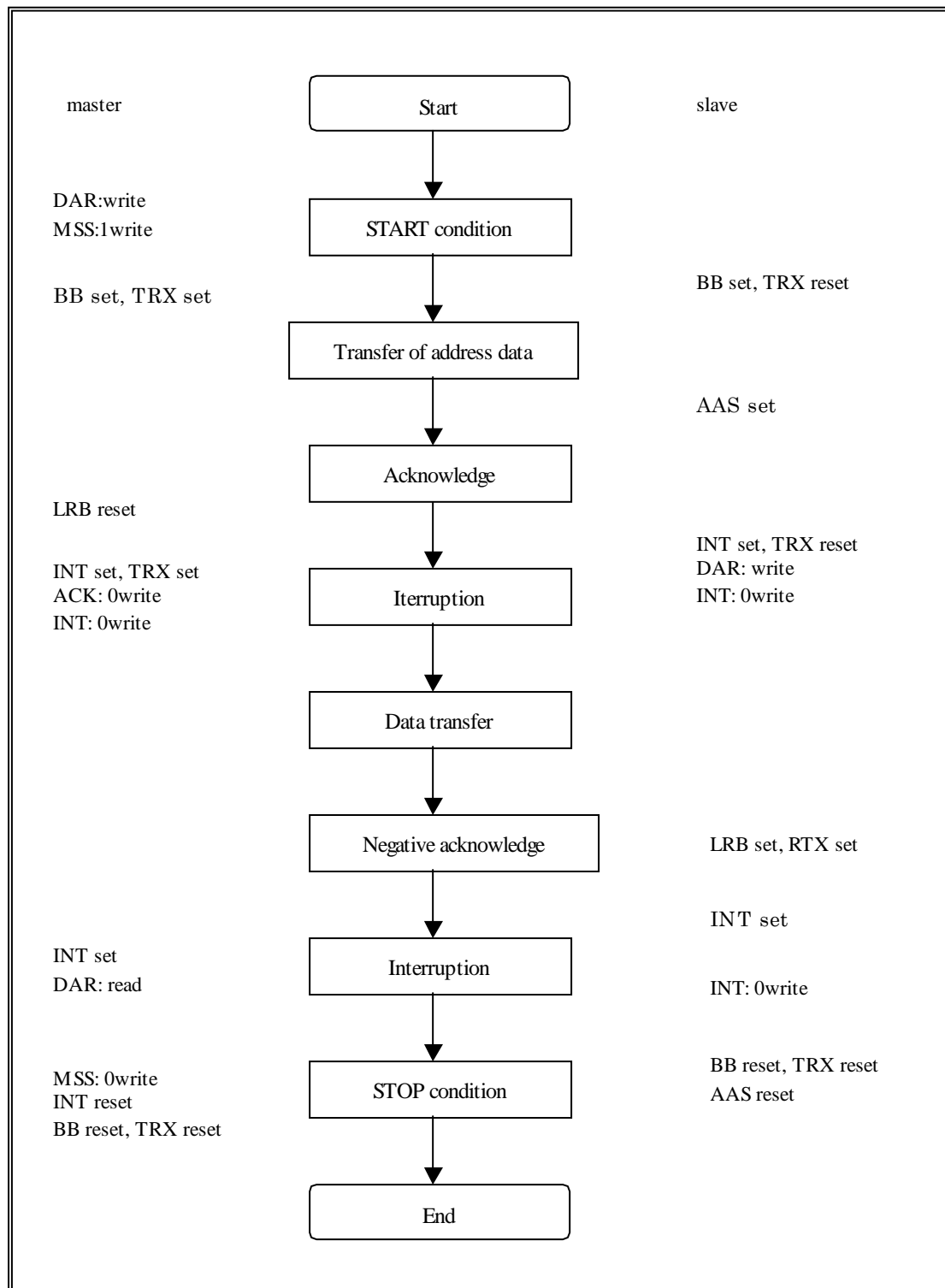
## 10.8 Initialize



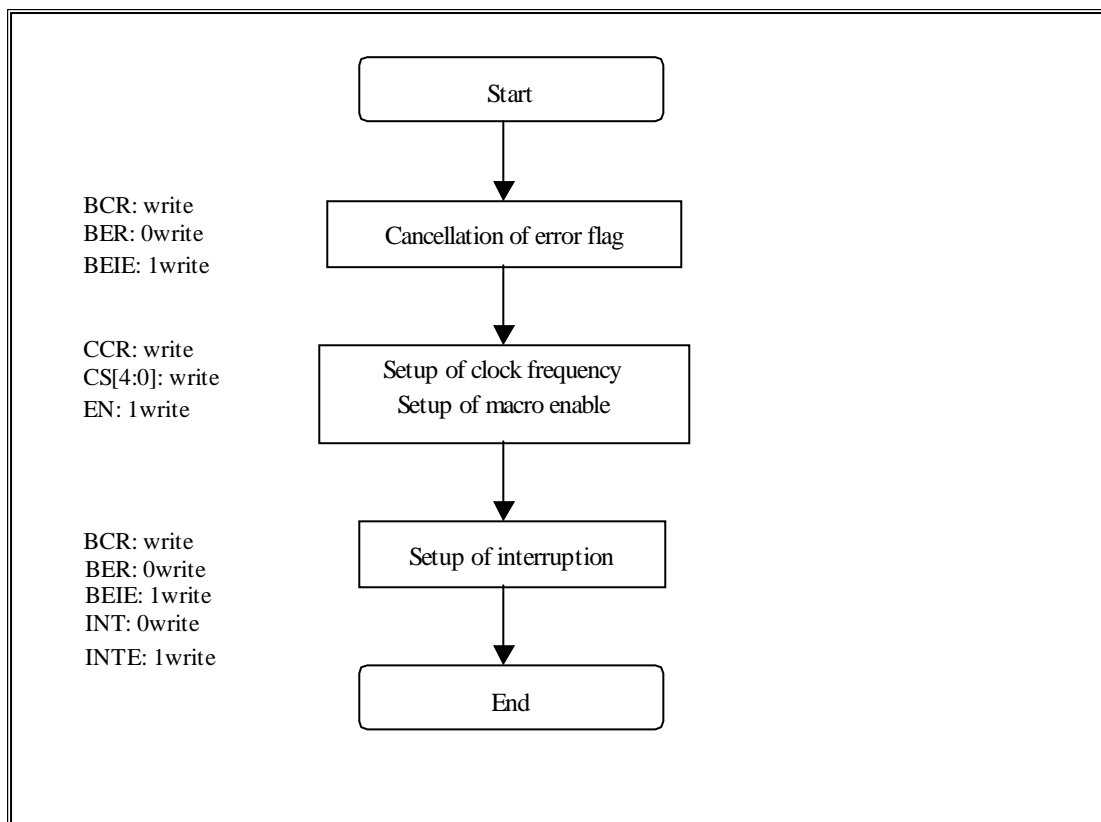
## 10.91-byte transfer from master to slave



## 10.10 1-byte transfer from slave to master



## 10.11 Recovery from bus error



## 10.12 Notes

### 10.12.1 About a 10-bit slave address

This module does not support the 10-bit slave address. Therefore, please do not specify the slave address of from 78H to 7bH to this module. If it is specified by mistake, a normal transfer cannot be performed although acknowledge bit is returned at the time of 1 byte reception.

### 10.12.2 About competition of SCC, MSS, and INT bit

Competition of the following byte transfer, generation of START condition, and generation of STOP condition happens by the simultaneous writing of SCC, MSS, and INT bit. At this time the priority is as follows.

- 1) The following byte transfer and generation of STOP condition  
If "0" is written to INT bit and "0" is written to MSS bit, priority will be given to "0" writing to MSS bit and STOP condition will be generated.
- 2) The following byte transfer and generation of START condition  
If "0" is written to INT bit and "1" is written to SCC bit, priority will be given to "1" writing to SCC bit and START condition will be generated.
- 3) Generation of START condition and generation of STOP condition  
The simultaneous writing of "1" in SCC bit and "0" to MSS bit is prohibition.

### 10.12.3 About setup of S serial transfer clock

When the delay of the positive edge of SCL terminal is large or when the clock is extended by the slave device, it may become smaller than setting value (calculation value) because of generation of overhead.

# 11 REGISTER

## 11.1 Register List

### 11.1.1 Host interface register list

Base = HostBase

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
000	DTC																															
										DTC																						
004									DST								DRM								DSU							
																DST														DNA	DAM	DBM
008																	LTS								DTS							
																									LTS							
010	LSTA																															
018	DRQ																															
020	IST																															
024	IMASK																															
02C																									SRST							
038	CCF																															
													CCE	COT																		
040	LSA																															
									LSA																							
044	LCO																															
									LCO																							
048																									LREQ							
05C																									RSW							
0f0	CID																															

[illegible]

11.1.2 Graphics memory interface register list

Base = HostBase

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FFFC	MMR																															
		TWR			ID			TRRD		TRC		TRP		TRAS		TRCD		LOWD		RTS		SAW		ASW						CL		



### 11.1.3 Display controller register list

Base = DisplayBase

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
000	DCE (Display Controller Enable)																DCM (Display Control Mode)															
	DEN													L45E	L23E	L1E	L0E	CKS	DCS							EEQ				SF	ESY	SYNC
100	DCEE (Display Controller Extend Enable)																DCEM (Display Control Extend Mode)															
	DEN												L5E	L4E	L3E	L2E	L1E	L0E	CKS	DCS						EEQ	EDE	EOF	EOD	SF	ESY	SYNC
004																																
008																																
00C																																
010																																
014																																
018																																
01C																																
020	L0M (L0 Mode)																															
	L0C																															
024																																
028																																
02C																																
110	L0EM (L0 Extend Mode)																															
	L0EC																															L0WP
114																																
118																																
030	L1M (L1 Mode)																															
	L1C	L1YC	L1CS	L1IM																												
034																																
120	L1EM (L1 Extend Mode)																															
	L1EC																															
040	L2M (L2 Mode)																															
	L2C	L2FLP																														
044																																
048																																
04C																																
050																																
054																																
130	L2EM (L2 Extend Mode)																															
	L2EC																														L2OM	L2WP
134																																
138																																

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
058	L3M (L3 Mode)																															
	L3C	L3FLP							L3S (L3 Width)										L3H (L3 Height)													
05C							L3OA0 (L3 Origin Address 0)																									
060							L3DA0 (L3 Display Address 0)																									
064							L3OA1 (L3 Origin Address 1)																									
068							L3DA1 (L3 Display Address 1)																									
06C					L3DY (L3 Display Y)												L3DX (L3 Display X)															
140	L3EM (L3 Extend Mode)																															
	L3EC								L3PB																						L3OM	L3WP
144					L3WY (L3 Window Y)														L3WX (L3 Window X)													
148					L3WH (L3 Window Height)														L3WW (L3 Window Width)													
070	L4M (L4 Mode)																															
	L4C	L4FLP							L4S (L4 Width)										L4H (L4 Height)													
074							L4OA0 (L4 Origin Address 0)																									
078							L4DA0 (L4 Display Address 0)																									
07C							L4OA1 (L4 Origin Address 1)																									
080							L4DA1 (L4 Display Address 1)																									
084					L4DY (L4 Display Y)												L4DX (L4 Display X)															
150	L4EM (L4 Extend Mode)																															
	L4EC																														L4OM	L4WP
154					L4WY (L4 Window Y)														L4WX (L4 Window X)													
158					L4WH (L4 Window Height)														L4WW (L4 Window Width)													
088	L5M (L5 Mode)																															
	L5C	L5FLP							L5S (L5 Width)										L5H (L5 Height)													
08C							L5OA0 (L5 Origin Address 0)																									
090							L5DA0 (L5 Display Address 0)																									
094							L5OA1 (L5 Origin Address 1)																									
098							L5DA1 (L5 Display Address 1)																									
09C					L5DY (L5 Display Y)												L5X (L5 Display X)															
160	L5EM (L5 Extend Mode)																															
	L5EC																														L5OM	L5WP
164					L5WY (L5 Window Y)														L5WX (L5 Window X)													
168					L5WH (L5 Window Height)														L5WW (L5 Window Width)													

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0A0									CPM								CUTC (Cursor Transparent Control)															
											CUE1	CUE0				CUO1	CUO0														CUZ1	CUTC
0A4																																
0A8																																
0AC																																
0B0																																
180	DLS (Display Layer Select)																															
184																																
0B4	L0BLD (L0 Blend)																															
188	L1BLD (L1 Blend)																															
18C	L2BLD (L2 Blend)																															
190	L3BLD (L3 Blend)																															
194	L4BLD (L4 Blend)																															
198	L5BLD (L5 Blend)																															

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0BC																	L0TC (L0 Transparent Control)															
																	L0ZT	L0TC (L0 Transparent Color)														
0C0	L2TR (L2 Transparent Control)																L3TR (L3 Transparent Control)															
	L2ZT	L2TC (L2 Transparent Color)														L3ZT	L3TR (L3 Transparent Color)															
1A0	L0TEC (L0 Extend Transparency Control)																															
	L0EZT																L0ETC (L0 Extend Transparent Color)															
1A4	L1TEC (L1 Transparent Extend Control)																															
	L1EZT																L1ETC (L1 Extend Transparent Color)															
1A8	L2TEC (L2 Transparent Extend Control)																															
	L2EZT																L2ETC (L2 Extend Transparent Color)															
1AC	L3TEC (L3 Transparent Extend Control)																															
	L3EZT																L3ETC (L3 Extend Transparent Color)															
1B0	L4ETC (L4 Extend Transparent Control)																															
	L4EZT																L4ETC (L4 Extend Transparent Color)															
1B4	L5ETC (L5 Extend Transparent Control)																															
	L5EZT																L5ETC (L5 Extend Transparent Color)															

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
400	L0PAL0																															
	A													R					G								B					
404	L0PAL1																															
:	:																															
7FC	L0PAL255																															
800	L1PAL0																															
	A													R					G								B					
804	L1PAL1																															
:	:																															
BFC	L1PAL255																															
1000	L2PAL0																															
	A													R					G								B					
1004	L2PAL1																															
:	:																															
13FC	L2PAL255																															
1400	L3PAL0																															
	A													R					G								B					
1404	L3PAL1																															
:	:																															
17FC	L3PAL255																															

## 11.1.4 Video Capture register list

Base = CaptureBase

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
000	VCM (Video Capture Mode)																																								
	VIE							CM				>																			VS										
004	CSC(Capture SCAle)																																								
	VSCI				VSCF								HSCI				HSCF																								
008	VCS(Video Capture Status)																																								
																															CE										
010	CBM(Capture Buffer Mode)																																								
	00												CBW																												
014	CBOA(Capture Bauffer Origin Address)																																								
													CBOA																												
018	CBLA(Capture Buffer Limit Address)																																								
													CBLA																												
01C												CIVSTR														CIHSTR															
020												CIVEND														CIHEND															
028	CHP(Capture Horizontal Pixel)																																								
																																CHP									
02C	CVP(Capture Vertical Pixel)																																								
													CVPP																			CVPN									
040	CLPF(Capture Low Pass Filter)																																								
								CVLPF								CHLPF																									
4000	CDCN(Capture Data Count for NTSC)																																								
													BDCN																		VDCN										
4004	CDCP(Capture Data Count for PAL)																																								
													BDCP																		VDCP										

## 11.1.5 Drawing engine register list

The parenthesized value in the Offset field denotes the absolute address used by the **SetRegister** command.

Base = DrawBase

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
000 (000)	Ys																																			
	s	s	s	s	Int												Frac																			
004 (001)	Xs																																			
	s	s	s	s	Int												Frac																			
008 (002)	dXdY																																			
	s	s	s	s	Int												Frac																			
00C (003)	XUs																																			
	s	s	s	s	Int												Frac																			
010 (004)	dXUdy																																			
	s	s	s	s	Int												Frac																			
014 (005)	XLs																																			
	s	s	s	s	Int												Frac																			
018 (006)	dXLdy																																			
	s	s	s	s	Int												Frac																			
01C (007)	USN																																			
	0	0	0	0	Int												0																			
020 (008)	LSN																																			
	0	0	0	0	Int												0																			
040 (010)	Rs																																			
	0	0	0	0	0	0	0	0	0	Int												Frac														
044 (011)	dRdx																																			
	s	s	s	s	s	s	s	s	s	Int												Frac														
048 (012)	dRdy																																			
	s	s	s	s	s	s	s	s	s	Int												Frac														
04C (013)	Gs																																			
	0	0	0	0	0	0	0	0	0	Int												Frac														
050 (014)	dGdx																																			
	s	s	s	s	s	s	s	s	s	Int												Frac														
054 (015)	dGdy																																			
	s	s	s	s	s	s	s	s	s	Int												Frac														
058 (016)	Bs																																			
	0	0	0	0	0	0	0	0	0	Int												Frac														
05C (017)	dBdx																																			
	s	s	s	s	s	s	s	s	s	Int												Frac														
060 (018)	dBdy																																			
	s	s	s	s	s	s	s	s	s	Int												Frac														

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
080 (020)	Zs																																			
	0	Int																Frac																		
084 (021)	dZdx																																			
	s	Int																Frac																		
088 (022)	dZdy																																			
	s	Int																Frac																		
0C0 (030)	Ss																																			
	s	s	s	Int																Frac																
0C4 (031)	dSdx																																			
	s	s	s	Int																Frac																
0C8 (032)	dSdy																																			
	s	s	s	Int																Frac																
0CC (033)	Ts																																			
	s	s	s	Int																Frac																
0D0 (034)	dTdx																																			
	s	s	s	Int																Frac																
0D4 (035)	dTdy																																			
	s	s	s	Int																Frac																
0D8 (036)	Qs																																			
	0	0	0	0	0	0	0	0	≦	Frac																										
0DC (037)	dQdx																																			
	s	s	s	s	s	s	s	s	≦	Frac																										
0E0 (038)	dQdy																																			
	s	s	s	s	s	s	s	s	≦	Frac																										
140 (050)	LPN																																			
	0	0	0	0	Int																0															
144 (051)	LXs																																			
	s	s	s	s	s	Int																Frac														
148 (052)	LXde																																			
	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	≦	Frac						
14C (053)	LYs																																			
	s	s	s	s	s	Int																Frac														
150 (054)	LYde																																			
	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	≦	Frac					
154 (055)	LZs																																			
	s	Int																Frac																		
158 (056)	LZde																																			
	s	Int																Frac																		



Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
180	PXdc																															
(060)	0	0	0	0	Int												0															
184	PYdc																															
(061)	0	0	0	0	Int												0															
188	PZdc																															
(062)	0	Int												0																		
200	RXs																															
(080)	0	0	0	0	Int												0															
204	RYs																															
(081)	0	0	0	0	Int												0															
208	RsizeX																															
(082)	0	0	0	0	Int												0															
20C	RsizeY																															
(083)	0	0	0	0	Int												0															
240	SADDR																															
(090)	0	0	0	0	0	0	0	0	Address																							
244	SStride																															
(091)	0	0	0	0	Int												0															
248	SRXs																															
(092)	0	0	0	0	Int												0															
24C	SRYs																															
(093)	0	0	0	0	Int												0															
250	DADDR																															
(094)	0	0	0	0	0	0	0	0	Address																							
254	DStride																															
(095)	0	0	0	0	Int												0															
258	DRXs																															
(096)	0	0	0	0	Int												0															
25C	DRYs																															
(097)	0	0	0	0	Int												0															
260	BsizeX																															
(098)	0	0	0	0	Int												0															
264	BsizeY																															
(099)	0	0	0	0	Int												0															
280	TColor																															
(09A)	0																Color															
28C	PNBPI																															
(0A3)																															PN	

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
3E0	BLPO																																	
(0F8)																												BCR						
400	CTR																																	
(100)								ED	EE	OE		FCNT					NF	FF	FE			SS				DS				PS				
404	IFSR																																	
(-)																													NF	FF	FE			
408	IFCNT																																	
(-)																											FCNT							
40C	SST																																	
(-)																															SS			
410	DS																																	
(-)																															DS			
414	PST																																	
(-)																															PS			
418	EST																																	
(-)																													FD	CE				
420	MDR0																																	
(108)											ZP				CF								GY	CX					BSV	BSH				
424	MDR1/MDR1S/MDR1B																																	
(109)				LW							BP	BL								LOG		BM		ZW		ZCL		ZC						
428	MDR2/MDR2S/MDR2TL																																	
(10a)			TT																LOG		BM		ZW		ZCL		ZC	AS	SM					
42C	MDR3																																	
(10b)							BA			TAB				TBL						TWS	TWT				TF		TC							
430	MDR4																																	
(10c)																				LOG		BM								TE				
43C	MDR7																																	
(10f)																										LTH	EZ	GG		PGH	PTH	PZH		

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
440	FBR																																							
(110)								FBASE																																
444	XRES																																							
(111)																						XRES																		
448	ZBR																																							
(112)								ZBASE																																
44C	TBR																																							
(113)								TBASE																																
450	PFBR																																							
(114)								PFBASE																																
454	CXMIN																																							
(115)																						CLIPXMIN																		
458	CXMAX																																							
(116)																						CLIPXMAX																		
45C	CYMIN																																							
(117)																						CLIPYMIN																		
460	CYMAX																																							
(118)																						CLIPYMAX																		
464	TXS																																							
(119)				TXSN															TXSM																					
468	TIS																																							
(11a)									TISN																				TISM											
46C	TOA																																							
(11b)																						XBO																		
470	SHO																																							
(11C)									SHOFFS																															
474	ABR																																							
(11D)								ABASE																																
480	FC																																							
(120)									FGC8/16																															
484	BC																																							
(121)									BGC8/16																															
488	ALF																																							
(122)																													A											
48C	BLP																																							
(123)																																								
494	TBC																																							
(125)									BC8/16																															

Offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
540	LX0dc																															
(150)	0	0	0	0	Int												0															
544	LY0dc																															
(151)	0	0	0	0	Int												0															
548	LX1dc																															
(152)	0	0	0	0	Int												0															
54C	LY1dc																															
(153)	0	0	0	0	Int												0															
580	X0dc																															
(160)	0	0	0	0	Int												0															
584	Y0dc																															
(161)	0	0	0	0	Int												0															
588	X1dc																															
(162)	0	0	0	0	Int												0															
58C	Y1dc																															
(163)	0	0	0	0	Int												0															
590	X2dc																															
(164)	0	0	0	0	Int												0															
594	Y2dc																															
(165)	0	0	0	0	Int												0															

### 11.1.6 I2C register list

I2C Interface offset address head = 1FCC000 (SH3/SH4), 0FCC000 (V83x,SPARClite)

Byte address	Data					
	31	24	23	16	15	8 7 0
000h	Reserved					BSR
004h	Reserved					BCR
008h	Reserved					CCR
00Ch	Reserved					ADR
010h	Reserved					DAR
014h	Access prohibition					
018h	Access prohibition					
01Ch	Access prohibition					

## 11.2 Explanation of Register

Terms appeared in this chapter are explained below:

1. Register address  
Indicates address of register
2. Bit number  
Indicates bit number
3. Bit field name  
Indicates name of each bit field included in register
4. R/W  
Indicates access attribute (read/write) of each field  
Each symbol shown in this section denotes the following:

R0     "0" always read at read.    Write access is Don't care.

W0     Only "0" can be written.

R       Read enabled

W       Write enabled

RX     Read enabled (read values undefined)

RW     Read and write enabled

RW0   Read and write 0 enabled

5. Initial value  
Indicates initial value of immediately before the reset of each bit field.
6. Handling of reserved bits  
"0" is recommended for the write value so that compatibility can be maintained with future products.

## 11.2.1 Host interface registers

### DTC (DMA Transfer Count)

Register address	HostBaseAddress + 00 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved								DTC																							
R/W	R0								RW																							
Initial value	0								Don't care																							

DTC is a readable/writable 32-bit register which sets the transfer count in either one long-word (32 bits) or 32 bytes units. When “1h” is set transfer is performed once. However, when “0h” is set, it indicates the maximum transfer count and 16M (16,777,216) data are transferred. During DMA transfer, the remaining transfer count is shown, therefore, the register value cannot be overwritten until DMA transfer is completed.

Note: This register need not be set in a mode in which Dual DMA ACK is not used, or the V832 mode.

### DSU (DMA Set Up)

Register address	HostBaseAddress + 04H							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved					DAM	DBM	DW
R/W	R0					RW	RW	RW
Initial value	0					0	0	0

#### Bit 0 DW (DMA Word)

Specifies DMA transfer count

- 0: 1-double word (32 bits) per DMA transfer
- 1: 8-double words (32 bytes) per DMA transfer (only SH4)

#### Bit 1 DBM (DMA Bus request Mode)

Selects DREQ mode used in DMA transfer in dual-address mode

- 0: DREQ is not negated during DMA transfer irrespective of cycle steal or burst mode.
- 1: DREQ is negated irrespective of cycle steal or burst mode when MINT cannot receive data (that is, when Ready cannot be returned immediately). When MINT is ready to receive data, DREQ is reasserted (When DMA transfer is performed in the single-address mode, DREQ is controlled automatically).

#### Bit 2 DAM (DMA Address Mode)

Selects DMA address mode in issuing external request

- 0: Dual address mode
- 1: Single address mode (SH4 only)

#### Bit 3 DNA (Dual address No Ack mode)

This bit is selected when using the dual-address-mode DMA that does not use the ACK signal.

- 0: Uses dual-address-mode DMA that uses ordinary ACK signal
- 1: Uses dual-address-mode DMA that does not use ACK signal  
Detection of the DREQ edge is supported; DREQ is negated per transfer. When data cannot be received irrespective of the Bit1 setting, DREQ continues being negated.

#### DRM (DMA Request Mask)

Register address	HostBaseAddress + 05H							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							DRM
R/W	R0							RW
Initial value	0							0

This register enables the DMA request. Setting “1” to this register to temporarily stop the DMA request from the MINT. The external request is enabled by setting “0” to this register.

#### DST (DMA Status)

Register address	HostBaseAddress + 06H							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							DST
R/W	R0							R
Initial value	0							0

This register indicates the DMA transfer status. DST is set to “1” during DMA transfer. This state is cleared to “0” when the DMA transfer is completed.

#### DTS (DMA Transfer Stop)

Register address	HostBaseAddress + 08H							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							DTS
R/W	R0							RW
Initial value	0							0

This register suspends DMA transfer.

An ongoing DMA transfer is suspended by setting DTS to “1”.

In the dual-address without ACK mode, to end the DMA transfer, write “1” to this register after CPU DMA transfer.

#### LTS (display Transfer Stop)

Register address	HostBaseAddress + 09H							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							LTS
R/W	R0							RW
Initial value	0							0

This register suspends DisplayList transfer.

Ongoing DisplayList transfer is suspended by setting LTS to “1”.

#### LSTA (displayList transfer Status)

Register address	HostBaseAddress + 10H							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							LSTA
R/W	R0							R
Initial value	0							0

This register indicates the DisplayList transfer status from Graphics Memory. LSTA is set to “1” while DisplayList transfer is in progress. This status is cleared to 0 when DisplayList transfer is completed



### DRQ (DMA ReQquest)

Register address	HostBaseAddress + 18 <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							DRQ
R/W	R0							RW1
Initial value	0							0

This register starts sending external DMA request.

DMA transfer using the external request handshake is triggered by setting DRQ to “1”. The external DREQ signal cannot be issued when DMA is masked by the DRM register. This register cannot be written “0”. When DMA transfer is completed, this status is cleared to “0”.

### IST (Interrupt Status)

Register address	HostBaseAddress + 20 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved																Resv	Reserved								IST		IST				
R/W	R0																R0W0	R0								RW0		RW0				
Initial value	0																0	0								0		0				

This register indicates the current interrupt status. It shows that an interrupt request is issued when “1” is set to this register. The interrupt status is cleared by writing “0” to this register.

- Bit 0            CERR (Command Error Flag)  
Indicates drawing command execution error interrupt
- Bit 1            CEND (Command END)  
Indicates drawing command end interrupt
- Bit 2            VSYNC (Vertical Sync.)  
Indicates vertical interrupt synchronization
- Bit 3            FSYNC (Frame Sync.)  
Indicates frame synchronization interrupt
- Bit 4            SYNCERR (Sync. Error)  
Indicates external synchronization error interrupt
- Bit 17 and 16   Reserved  
This field is provided for testing.  
Normally, the read value is “0”, but note that it may be “1” when a drawing command error (Bit 0) has occurred.

## IMASK (Interrupt MASK)

Register address	HostBaseAddress + 24 <sub>H</sub>																																
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name	Reserved																Resv	Reserved														IMASK	IMASK
R/W	R0																R0W0	R0														RW	RW
Initial value	0																0	0														0	0

This register masks interrupt requests. Even when the interrupt request is issued for the bit to which “0” is written, interrupt signal is not asserted for CPU.

- Bit 0      CERRM (Command Error Interrupt Mask)  
Masks drawing command execution error interrupt
- Bit 1      CENDM (Command Interrupt Mask)  
Masks drawing command end interrupt
- Bit 2      VSYNCM (Vertical Sync. Interrupt Mask)  
Masks vertical synchronization interrupt
- Bit 3      FSYNCH (Frame Sync. Interrupt Mask)  
Masks frame synchronization interrupt
- Bit 4      SYNCERRM (Sync Error Mask)  
Masks external synchronization error interrupt

## SRST (Software ReSeT)

Register address	HostBaseAddress + 2C <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							SRST
R/W	R0							W1
Initial value	0							0

This register controls software reset. When “1” is set to this register, a software reset is performed.

### LSA (displayList Source Address)

Register address	HostBaseAddress + 40 <sub>H</sub>																																
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name	Reserved								LSA																								
R/W	R0								RW																								R0
Initial value	0								Don't care																								0

This register sets the DisplayList transfer source address. When DisplayList is transferred from Graphics Memory, set the transfer start address of DisplayList stored in Graphics Memory. Since the lower two bits of this register are always treated as “0”, DisplayList must be 4-byte aligned. The values set at this register do not change during or after transfer.

### LCO (displayList Count)

Register address	HostBaseAddress + 44 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved								LCO																							
R/W	R0								RW																							
Initial value	0								Don't care																							

This register sets the DisplayList transfer count. Set the display list transfer count by the long word. When “1h” is set, 1-word data is transferred. When “0” is set, it is considered to be the maximum count and 16M (16,777,216) words of data are transferred. The values set at this register do not change during or after transfer.

### LREQ (displayList transfer REQuest)

Register address	HostBaseAddress + 48 <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							LREQ
R/W	R0							RW1
Initial value	0							0

This register triggers DisplayList transfer from the Graphics Memory. Transfer is started by setting LREQ to “1”. The DisplayList is transferred from the Graphics Memory to the internal display list FIFO. Access to the display list FIFO by the CPU or DMA is disabled during transfer.

### RSW (Register location Switch)

Register address	HostBaseAddress + 5C <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved							RSW
R/W	R0							RW
Initial value	0							0

In SH3 or SH4 mode, set this register when moving the register area from the center (1FC0000) to the end of the MINT area (3FC0000). This move can be performed when “1” is written to this register.

Set this register at the first access after reset. Access MINT after about 20 bus clocks after setting the register.

### CCF (Change of Clock Frequency)

Register address	HostBaseAddress + 38 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved												CGE	COT	Reserved																	
R/W	RW0												RW	RW	RW0																	
Initial value	0												00	00	0																	

This register changes the operating frequency.

Bit 19 and 18 CGE (Clock select for Geometry Engine)

Selects the clock for the geometry engine

11 Reserved

10 Reserved

01 Reserved

00 100 MHz

Bit 17 and 16 COT (Clock select for the others except-geometry engine)

Selects the clock for other than the geometry engine

11 Reserved

10 Reserved

01 Reserved

00 100 MHz

#### Notes:

1. Write "0" to the bit field other than the above ([31:20], [15:00]).
2. Mint supports only 100MHz. Do not set other value.

## 11.2.2 Graphics memory interface registers

### MMR (Memory I/F Mode Register)

Register address	HostBaseAddress + FFFC <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	*1		RWR		Reserved		*1		*1		TRRD		TRC		TRP		TRAS		TRCD		LOWD		RTS		RAW		ASW		CL			
R/W	RW	RW	R		R1 W0		R		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW			
Initial value	0	0	Don't care		1		0		00		0000		00		000		00		00		00		000		000		0		000			

\*1: Reserved

This register sets the mode of the graphics memory interface. A value must be written to this register after a reset. (When default setting is performed, a value must also be written to this register.) Only write once to this register; do not change the written value during operation.

This register is not initialized at a software reset.

Bit 2 to 0 CL (CAS Latency)

Sets the CAS latency. Write the same value as this field, to the mode register for SDRAM

011 CL3

010 CL2

Other than the above Setting disabled

Bit 3 ASW (Attached SDRAM bit Width)

Sets the bit width of the data bus (memory bus width mode)

1 64 bit

0 32 bit

Bit 6 to 4 SAW (SDRAM Address Width)

Sets the bit width of the SDRAM address

001 15 bit BANK 2 bit ROW 13 bit COL 9 bit SDRAM

111 14 bit BANK 2 bit ROW 12 bit COL 9 bit SDRAM

110 14 bit BANK 2 bit ROW 12 bit COL 8 bit SDRAM

101 13 bit BANK 2 bit ROW 11 bit COL 8 bit SDRAM

100 12 bit BANK 1 bit ROW 11 bit COL 8 bit FCRAM

000 14 bit BANK 2 bit ROW 12 bit COL 8 bit SDRAM

Other than the above Setting disabled

Bit 9 to 7 RTS (Refresh Timing Setting)

Sets the refresh interval

000 Refresh is performed every 384 internal clocks.

111 Refresh is performed every 1552 internal clocks.

001 to 110 Refresh is performed every '64 × n' internal clocks in the 64 to 384 range.

Bit 11 and 10	LOWD	
		Sets the count of clocks secured for the period from the instant the ending data is output to the instant the write command is issued.
	10	2 clocks
	00	2 clocks
	Other than the above	Setting disabled
Bit 13 and 12	TRCD	
		Sets the wait time secured from the bank active to CAS. The clock count is used to express the wait time.
	11	3 clocks
	10	2 clocks
	01	1 clock
	00	0 clock
Bit 16 to 14	TRAS	
		Sets the minimum time for 1 bank active. The clock count is used to express the minimum time.
	111	7 clocks
	110	6 clocks
	101	5 clocks
	100	4 clocks
	011	3 clocks
	010	2 clocks
	Other than the above	Setting disabled
Bit 18 and 17	TRP	
		Sets the wait time secured from the pre-charge to the bank active. The clock count is used to express the wait time.
	11	3 clocks
	10	2 clocks
	01	1 clock
Bit 22 to 19	TRC	
		This field sets the wait time secured from the refresh to the bank active. The clock count is used to express the wait time.
	1010	10 clocks
	1001	9 clocks
	1000	8 clocks
	0111	7 clocks
	0110	6 clocks
	0101	5 clocks
	0100	4 clocks

	0011	3 clocks
	Other than the above	Setting disabled
Bit 24 and 23	TRRD	
	Sets the wait time secured from the bank active to the next bank active. The clock count is used to express the wait time.	
	11	3 clocks
	10	2 clocks
Bit 26	Reserved	
	Always write "0" at write.	
	"1" is always read at read.	
Bit 30	TWR	
	Sets the write recovery time (the time from the write command to the read or to the pre-charge command).	
	1	2 clocks
	0	1 clock



### 11.2.3 Display control register

#### DCM (Display Control Mode) / DCEM (Display Control Extend Mode)

Register address	DisplayBaseAddress + 00 <sub>H</sub> (DisplayBaseAddress + 100 <sub>H</sub> )															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	CKS	Reserved	SC						EEQ	ODE	Reserved	Reserved	SF	ESY	SYNC	
R/W	RW	RW0	RW						RW	RW	RX	RX	RW	RW	RW	
Initial value	0	0	01110 (DCM)						0	X			0	1	00	
			11101 (DCEM)													

This register controls the display count mode. It is not initialized by a software reset. This register is mapped to two addresses. The difference between the two registers is the format of the frequency division rate setting (SC).

Bit 1 to 0      SYNC (Synchronize)  
Set synchronization mode  
X0    Non-interlace mode  
10    Interlace mode  
11    Interlace video mode

Bit 2            ESY (External Synchronize)  
Sets external synchronization mode  
0:    External synchronization disabled  
1:    External synchronization enabled

Bit 3            SF (Synchronize signal format)  
Sets format of synchronization (VSYNC, HSYNC) signals  
0:    Negative logic  
1:    Positive logic

Bit 7            EEQ (Enable Equalizing pulse)  
Sets CCYNC signal mode  
0:    Does not insert equalizing pulse into CCYNC signal  
1:    Inserts equalizing pulse into CCYNC signal

Bit 13 to 8    SC (Scaling)  
Divides display reference clock by the preset ratio to generate dot clock

Offset = 0				Offset = 100 <sub>H</sub>			
x00000	Frequency not divided			000000	Frequency not divided		
x00001	Frequency division rate = 1/4			000001	Frequency division rate = 1/2		
x00010	Frequency division rate = 1/6			000010	Frequency division rate = 1/3		
X00011	Frequency division rate = 1/8			000011	Frequency division rate = 1/4		
:				:			
x11111	Frequency division rate = 1/64			111111	Frequency division rate = 1/64		

When  $n$  is set, with Offset = 0, the frequency division rate is  $1/(2n + 2)$ .

When  $m$  is set, with Offset = 100h, the frequency division rate is  $1/(m + 1)$ .

Basically, these are setting parameters with the same function ( $2n + 2 = m + 1$ ).

Because of this,  $m = 2n + 1$  is established. When  $n$  is set to the SC field with Offset = 0,  $2n + 1$  is reflected with Offset = 100h.

Also, when PLL is selected as the reference clock, frequency division rates 1/1 to 1/5 are non-functional even when set; other frequency division rates are assigned.

- Bit 15      CKS (Clock Source)
- Selects reference clock
- 0:    Internal PLL output clock
- 1:    DCLKI input

## DCE (Display Controller Enable)

Register address	DisplayBaseAddress + 02 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	DEN	Reserved											L45E	L23E	L1E	L0E
R/W	RW	R0											RW	RW	RW	RW
Initial value	0	0											0	0	0	0

This register controls enabling the video signal output and display of each layer. Layer enabling is specified in four-layer units to maintain backward compatibility with previous products.

- Bit 0**      **L0E (L0 layer Enable)**  
Enables display of the L0 layer. The L0 layer corresponds to the C layer for previous products.  
0:      Does not display L0 layer  
1:      Displays L0 layer
- Bit 1**      **L1E (L1 layer Enable)**  
Enables display of the L1 layer. The L1 layer corresponds to the W layer for previous products.  
0:      Does not display L1 layer  
1:      Displays L1 layer
- Bit 2**      **L23E (L2 & L3 layer Enable)**  
Enables simultaneous display of the L2 and L3 layers. These layers correspond to the M layer for previous products.  
0:      Does not display L2 and L3 layer  
1:      Displays L2 and L3 layer
- Bit 3**      **L45E (L4 & L5 layer Enable)**  
Enables simultaneous display of the L4 and L5 layers. These layers correspond to the B layer for previous products.  
0:      Does not display L4 and L5 layer  
1:      Displays L4 and L5 layer
- Bit 15**     **DEN (Display Enable)**  
Enables display  
0:      Does not output display signal  
1:      Outputs display signal

## DCEE (Display Controller Extend Enable)

Register address	DisplayBaseAddress + 102 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	DEN	Reserved									L5E	L4E	L3E	L2E	L1E	L0E
R/W	RW	R0									RW	RW	RW	RW	RW	RW
Initial value	0	0									0	0	0	0	0	0

This register controls enabling the video signal output and display of each layer. This register has the same function as DCE.

- Bit 0      L0E (L0 layer Enable)  
Enables L0 layer display  
0:      Does not display L0 layer  
1:      Displays L0 layer
- Bit 1      L1E (L1 layer Enable)  
Enables L1 layer display  
0:      Does not display L1 layer  
1:      Displays L1 layer
- Bit 2      L2E (L2 layer Enable)  
Enables L2 layer display  
0:      Does not display L2 layer  
1:      Displays L2 layer
- Bit 3      L3E (L3 layer Enable)  
Enables L3 layer display  
0:      Does not display L3 layer  
1:      Displays L3 layer
- Bit 4      L4E (L4 layer Enable)  
Enables L4 layer display  
0:      Does not display L4 layer  
1:      Displays L4 layer
- Bit 5      L5E (L5 layer Enable)  
Enables L5 layer display  
0:      Does not display L5 layer  
1:      Displays L5 layer
- Bit 15     DEN (Display Enable)  
Enables display  
0:      Does not output display signal  
1:      Outputs display signal

#### HTP (Horizontal Total Pixels)

Register address	DisplayBaseAddress + 06 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				HTP											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal total pixel count. Setting value + 1 is the total pixel count.

#### HDP (Horizontal Display Period)

Register address	DisplayBaseAddress + 08 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				HDP											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the total horizontal display period in unit of pixel clocks. Setting value + 1 is the pixel count for the display period.

#### HDB (Horizontal Display Boundary)

Register address	DisplayBaseAddress + 0A <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				HDB											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the display period of the left part of the window in unit of pixel clocks. Setting value + 1 is the pixel count for the display period of the left part of the window. When the window is not divided into right and left before display, set the same value as HDP.

#### HSP (Horizontal Synchronize pulse Position)

Register address	DisplayBaseAddress + 0C <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				HSP											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the pulse position of the horizontal synchronization signal in unit of pixel clocks. When the clock count since the start of the display period reaches setting value + 1, the horizontal synchronization signal is asserted.

#### HSW (Horizontal Synchronize pulse Width)

Register address	DisplayBaseAddress + 0E <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	HSW							
R/W	RW							
Initial value	Don't care							

This register controls the pulse width of the horizontal synchronization signal in unit of pixel clocks. Setting value + 1 is the pulse width clock count.

### VSW (Vertical Synchronize pulse Width)

Register address	DisplayBaseAddress + 0F <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved		VSW					
R/W	R0		RW					
Initial value	0		Don't care					

This register controls the pulse width of vertical synchronization signal in unit of raster. Setting value + 1 is the pulse width raster count.

### VTR (Vertical Total Rasters)

Register address	DisplayBaseAddress + 12 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				VTR											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical total raster count. Setting value + 1 is the total raster count. For the interlace display, Setting value + 1.5 is the total raster count for 1 field; 2 × setting value + 3 is the total raster count for 1 frame (see **Section 8.3.2**).

### VSP (Vertical Synchronize pulse Position)

Register address	DisplayBaseAddress + 14 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				VSP											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the pulse position of vertical synchronization signal in unit of raster. The vertical synchronization pulse is asserted starting at the setting value + 1st raster relative to the display start raster.

### VDP (Vertical Display Period)

Register address	DisplayBaseAddress + 16 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				VDP											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical display period in unit of raster. Setting value + 1 is the count of raster to be displayed.

## L0M (L0 layer Mode)

Register address	DisplayBaseAddress + 20 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L0C		Reserved		Reserved					LOW								Reserved					CH									
R/W	RW		R0		R0					RW								R0					RW									
Initial value	0		0		0					Don't care								0					Don't care									

Bit 11 to 0      L0H (L0 layer Height)  
Specifies the height of the logic frame of the L0 layer in pixel units. Setting value + 1 is the height

Bit 23 to 16      LOW (L0 layer memory Width)  
Sets the memory width (stride) of the logic frame of the L0 layer in 64-byte units

Bit 31            L0C (L0 layer Color mode)  
Sets the color mode for L0 layer  
0      Indirect color (8 bits/pixel) mode  
1      Direct color (16 bits/pixel) mode

## L0EM (L0-layer Extended Mode)

Register address	DisplayBaseAddress + 110 <sub>H</sub>																											
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	-----	4	3	2	1	0
Bit field name	L0EC		Reserved					L0PB				Reserved										LOWP						
R/W	RW		R0					RW				R0										RW						
Initial value			0									0										0						

Bit 0            L0 WP (L0 layer Window Position enable)  
Selects the display position of L0 layer  
0      Compatibility mode display (C layer supported)  
1      Window display

Bit 23 to 20      L0PB (L0 layer Palette Base)  
Shows the value added to the index when subtracting palette of L0 layer. 16 times of setting value is added.

Bit 31 and 30      L0EC (L0 layer Extended Color mode)  
Sets extended color mode for L0 layer  
00      Mode determined by L0C  
01      Direct color (24 bits/pixel) mode  
1x      Reserved

### L0OA (L0 layer Origin Address)

Register address	DisplayBaseAddress + 24 <sub>H</sub>																																	
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved						L0OA																											
R/W	R0						RW																										R0	
Initial value	0						Don't care																										0000	

This register sets the origin address of the logic frame of the L0 layer. Since lower 4 bits are fixed at "0", address 16-byte-aligned.

### L0DA (L0-layer Display Address)

Register address	DisplayBaseAddress + 28 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L0DA																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the display origin address of the L0 layer. For the direct color mode (16 bits/pixel), the lower 1 bit is "0", and this address is treated as being aligned in 2 bytes.

### L0DX (L0-layer Display position X)

Register address	DisplayBaseAddress + 2C <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L0DX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (X coordinates) of the L0 layer on the basis of the origin of the logic frame in pixels.

### L0DY (L0-layer Display position Y)

Register address	DisplayBaseAddress + 2E <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L0DY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (Y coordinates) of the L0 layer on the basis of the origin of the logic frame in pixels.



#### LOWX (L0 layer Window position X)

Register address	DisplayBaseAddress + 114 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				LOWX											
R/W	R0				RW											
Initial value	0															

This register sets the X coordinates of the display position of the L0 layer window.

#### LOWY (L0 layer Window position Y)

Register address	DisplayBaseAddress + 116 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				LOWY											
R/W	R0				RW											
Initial value	0															

This register sets the Y coordinates of the display position of the L0 layer window.

#### LOWW (L0 layer Window Width)

Register address	DisplayBaseAddress + 118 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				LOWW											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal direction display size (width) of the L0 layer window. Do not specify "0".

#### LOWH (L0 layer Window Height)

Register address	DisplayBaseAddress + 11A <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				LOWH											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical direction display size (height) of the L0 layer window. Setting value + 1 is the height.

## L1M (L1-layer Mode)

Register address	DisplayBaseAddress + 30 <sub>H</sub>																														
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	---			5	4	3	2	1	0
Bit field name	L1C	L1YC	L1CS	L1IM	Reserved				L1W								Reserved														
R/W	RW	RW	RW	RW	R0				RW								R0														
Initial value	0	0	0	0	0				Don't Care								0														

- Bit 23 to 16    L1W (L1 layer memory Width)  
Sets the memory width (stride) of the logic frame of the L layer in unit of 64 bytes
- Bit 28    L1IM (L1 layer Interlace Mode)  
Sets video capture mode when L1CS in capture mode  
0:    Normal mode  
1:    For non-interlace display, displays captured video graphics in WEAVE mode  
For interlace and video display, buffers are managed in frame units (pair of odd field and even field).
- Bit 29    L1CS (L1 layer Capture Synchronize)  
Sets whether the layer is used as normal display layer or as video capture  
0:    Normal mode  
1:    Capture mode
- Bit 30    L1YC (L1 layer YC mode)  
Sets color format of L1 layer  
The YC mode must be set for video capture.  
0:    RGB mode  
1:    YC mode
- Bit 31    L1C (L1 layer Color mode)  
Sets color mode for L1 layer  
0:    Indirect color (8 bits/pixel) mode  
1:    Direct color (16 bits/pixel) mode

### L1EM (L1 layer Extended Mode)

Register address	DisplayBaseAddress + 120 <sub>H</sub>																													
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	---			4	3	2	1	0
Bit field name	L1EC				Reserved				L1PB				Reserved																	
R/W	RW				R0				RW				R0																	
Initial value	0				0				0				0																	

Bit 23 to 20 L1PB (L1 layer Palette Base)

Shows the value added to the index when subtracting palette of L1 layer. 16 times of setting value is added.

Bit 31 to 30 L1EC (L1 layer Extended Color mode)

Sets extended color mode for L1 layer

00 Mode determined by L0C

01 Direct color (24 bits/pixel) mode

1x Reserved

### L1DA (L1 layer Display Address)

Register address	DisplayBaseAddress + 34 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L0DA																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the display origin address of the L1 layer. For the direct color mode (16 bits/pixel), the lower 1 bit is "0", and this register is treated as being aligned in 2 bytes. Wraparound processing is not performed for the L1 layer, so the frame origin linear address and display position (X coordinates, and Y coordinates) are not specified.

### L1WX (L1 layer Window position X)

Register address	DisplayBaseAddress + 124 <sub>H</sub> (DisplayBaseAddress + 18 <sub>H</sub> )															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L1WX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the X coordinates of the display position of the L1 layer window. This register is placed in two address spaces. The parenthesized address is the register address to maintain compatibility with previous products. The same applies to L1WY, L1WW, and L1WH.

### L1WY (L1 layer Window position Y)

Register address	DisplayBaseAddress + 126 <sub>H</sub> (DisplayBaseAddress + 1A <sub>H</sub> )															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L1WY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the Y coordinates of the display position of the L1 layer window.

### L1WW (L1 layer Window Width)

Register address	DisplayBaseAddress + 128 <sub>H</sub> (DisplayBaseAddress + 1C <sub>H</sub> )															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L1WW											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal direction display size (width) of the L1 layer window. Do not specify "0".

### L1WH (L1 layer Window Height)

Register address	DisplayBaseAddress + 12A <sub>H</sub> ((DisplayBaseAddress + 1E <sub>H</sub> ))															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L1WH											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical direction display size (height) of the L1 layer window. Setting value + 1 is the height.

## L2M (L2 layer Mode)

Register address	DisplayBaseAddress + 40 <sub>H</sub>																															
Bit number	31	30	29	28	27	--	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name	L2C		L2FLP		Reserved			L2W								Reserved				L2H												
R/W	RW		RW		R0			RW								R0				RW												
Initial value	0		00		0			Don't care								0				Don't care												

Bit 11 to 0      L2H (L2 layer Height)  
Specifies the height of the logic frame of the L2 layer in pixel units. Setting value + 1 is the height

Bit 23 to 16      L2W (L2 layer memory Width)  
Sets the memory width (stride) of the logic frame of the L2 layer in 64-byte units

Bit 30 and 29      L2FLP (L2 layer Flip mode)  
Sets flipping mode for L2 layer  
00      Displays frame 0  
01      Displays frame 1  
10      Switches frame 0 and 1 alternately for display  
11      Reserved

Bit 31      L2C (L2 layer Color mode)  
Sets the color mode for L2 layer  
0      Indirect color (8 bits/pixel) mode  
1      Direct color (16 bits/pixel) mode

## L2EM (L2 layer Extended Mode)

Register address	DisplayBaseAddress + 130 <sub>H</sub>																														
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	-----				4	3	2	1	0
Bit field name	L2EC		Reserved						L2PB		Reserved										L2OM		LOWP								
R/W	RW		R0						RW		R0										RW		RW								
Initial value	00		0						0		0												0								

- Bit 0            L2 WP (L2 layer Window Position enable)  
Selects the display position of L2 layer  
0      Compatibility mode display (ML layer supported)  
1      Window display
- Bit 1            L2OM (L2 layer Overlay Mode)  
Selects the overlay mode for L2 layer  
0      Compatibility mode  
1      Extended mode
- Bit 23 to 20    L2PB (L2 layer Palette Base)  
Shows the value added to the index when subtracting palette of L2 layer. 16 times of setting value is added.
- Bit 31 and 30   L2EC (L2 layer Extended Color mode)  
Sets extended color mode for L2 layer  
00      Mode determined by L2C  
01      Direct color (24 bits/pixel) mode  
1x      Reserved

### L2OA0 (L2 layer Origin Address 0)

Register address	DisplayBaseAddress + 44 <sub>H</sub>																																	
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved						L2OA0																											
R/W	R0						RW																										R0	
Initial value	0						Don't care																										0000	

This register sets the origin address of the logic frame of the L2 layer in frame 0. Since lower 4 bits are fixed to "0", this address is 16-byte aligned.

### L2DA0 (L2 layer Display Address 0)

Register address	DisplayBaseAddress + 48 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L2DA0																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the origin address of the L2 layer in frame 0. For the direct color mode (16 bits/pixel), the lower 1 bit is "0" and this address is 2-byte aligned.

### L2OA1 (L2 layer Origin Address 1)

Register address	DisplayBaseAddress + 4C <sub>H</sub>																																	
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved						L2OA1																											
R/W	R0						RW																										R0	
Initial value	0						Don't care																										0000	

This register sets the origin address of the logic frame of the L2 layer in frame 1. Since lower 4-bits are fixed to "0", this address is 16-byte aligned.

### L2DA1 (L2 layer Display Address 1)

Register address	DisplayBaseAddress + 50 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L2DA1																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the origin address of the L2 layer in frame 1. For the direct color mode (16 bits/pixel), the lower 1 bit is "0" and this address is 2-byte aligned.

### L2DX (L2 layer Display position X)

Register address	DisplayBaseAddress + 54 <sub>H</sub>																														
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0															
Bit field name	Reserved						L2DX																								
R/W	R0						RW																								
Initial value	0						Don't care																								

This register sets the display starting position (X coordinates) of the L2 layer on the basis of the origin of the logic frame in pixels.

### L2DY (L2 layer Display position Y)

Register address	DisplayBaseAddress + 56 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L2DY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (Y coordinates) of the L2 layer on the basis of the origin of the logic frame in pixels.

### L2WX (L2 layer Window position X)

Register address	DisplayBaseAddress + 134 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L2WX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the X coordinates of the display position of the L2 layer window.

### L2WY (L2 layer Window position Y)

Register address	DisplayBaseAddress + 136 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L2WY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the Y coordinates of the display position of the L2 layer window.

### L2WW (L2 layer Window Width)

Register address	DisplayBaseAddress + 138 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L2WW											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal direction display size (width) of the L2 layer window. Do not specify "0".

### L2WH (L2 layer Window Height)

Register address	DisplayBaseAddress + 13A <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L2WH											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical direction display size (height) of the L2 layer window. Setting value + 1 is the height.



### L3M (L3 layer Mode)

Register address	DisplayBaseAddress + 58 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L3C	L3FLP		Reserved						L3W				Reserved				L3H														
R/W	RW	R0		R0						RW				R0				RW														
Initial value	0	0		0						Don't care				0				Don't care														

- Bit 11 to 0      L3H (L3 layer Height)  
Specifies the height of the logic frame of the L3 layer in pixel units. Setting value + 1 is the height
- Bit 23 to 16      L3W (L3 layer memory Width)  
Sets the memory width (stride) of the logic frame of the L3 layer in 64-byte units
- Bit 30 and 29      L3FLP (L3 layer Flip mode)  
Sets flipping mode for L3 layer
- 00    Displays frame 0
  - 01    Displays frame 1
  - 10    Switches frame 0 and 1 alternately for display
  - 11    Reserved
- Bit 31      L3C (L3 layer Color mode)  
Sets the color mode for L3 layer
- 0    Indirect color (8 bits/pixel) mode
  - 1    Direct color (16 bits/pixel) mode

### L3EM (L3 layer Extended Mode)

Register address	DisplayBaseAddress + 140 <sub>H</sub>																													
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	---			4	3	2	1	0
Bit field name	L3EC		Reserved						L3PB		Reserved										L3OM		L3WP							
R/W	RW		R0						RW		R0										RW		RW							
Initial value	00		0						0		0										0		0							

- Bit 0                      L3 WP (L3 layer Window Position enable)  
Selects the display position of L3 layer  
0      Compatibility mode display (MR layer supported)  
1      Window display
- Bit 1                      L3OM (L3 layer Overlay Mode)  
Selects the overlay mode for L3 layer  
0      Compatibility mode  
1      Extended mode
- Bit 23 to 20              L3PB (L3 layer Palette Base)  
Shows the value added to the index when subtracting palette of L3 layer.    16 times of setting value is added.
- Bit 31 and 30              L3EC (L3 layer Extended Color mode)  
Sets extended color mode for L3 layer  
00      Mode determined by L3C  
01      Direct color (24 bits/pixel) mode  
1x      Reserved

### L3OA0 (L3 layer Origin Address 0)

Register address	DisplayBaseAddress + 5C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L3OA0																									
R/W	R0						RW																								R0	
Initial value	0						Don't care																								0000	

This register sets the origin address of the logic frame of the L3 layer in frame 0. Since lower 4 bits are fixed to "0", this address is 16-byte aligned.

### L3DA0 (L3 layer Display Address 0)

Register address	DisplayBaseAddress + 60 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L3DA0																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the origin address of the L3 layer in frame 0. For the direct color mode (16 bits/pixel), the lower 1 bit is "0" and this address is 2-byte aligned.

### L3OA1 (L3 layer Origin Address 1)

Register address	DisplayBaseAddress + 64 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L3OA1																									
R/W	R0						RW																				R0					
Initial value	0						Don't care																				0000					

This register sets the origin address of the logic frame of the L3 layer in frame 1. Since lower 4-bits are fixed to "0", this address is 16-byte aligned.

### L3DA1 (L3 layer Display Address 1)

Register address	DisplayBaseAddress + 68 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L3DA1																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the origin address of the L3 layer in frame 1. For the direct color mode (16 bits/pixel), the lower 1 bit is "0" and this address is 2-byte aligned.

### L3DX (L3 layer Display position X)

Register address	DisplayBaseAddress + 6C <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L3DX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (X coordinates) of the L3 layer on the basis of the origin of the logic frame in pixels.

### L3DY (L3 layer Display position Y)

Register address	DisplayBaseAddress + 6E <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L3DY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (Y coordinates) of the L3 layer on the basis of the origin of the logic frame in pixels.

### L3WX (L3 layer Window position X)

Register address	DisplayBaseAddress + 144 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L3WX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the X coordinates of the display position of the L3 layer window.

### L3WY (L3 layer Window position Y)

Register address	DisplayBaseAddress + 146 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L3WY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the Y coordinates of the display position of the L3 layer window.

### L3WW (L3 layer Window Width)

Register address	DisplayBaseAddress + 148 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L3WW											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal direction display size (width) of the L3 layer window. Do not specify "0".

### L3WH (L3-layer Window Height)

Register address	DisplayBaseAddress + 14A <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L3WH											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical direction display size (height) of the L3 layer window. Setting value + 1 is the height.

#### L4M (L4 layer Mode)

Register address	DisplayBaseAddress + 70 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L4C	L4FLP	Reserved					L4W					Reserved					L4H														
R/W	RW	RW	R0					RW					R0					RW														
Initial value			0					Don't care					0					Don't care														

Bit 11 to 0      L4H (L4 layer Height)  
Specifies the height of the logic frame of the L4 layer in pixel units. Setting value + 1 is the height

Bit 23 to 16      L4W (L4 layer memory Width)  
Sets the memory width (stride) logic frame of the L4 layer in 64-byte units

Bit 30 and 29      L4FLP (L4 layer Flip mode)  
Sets flipping mode for L4 layer  
00      Displays frame 0  
01      Displays frame 1  
10      Switches frame 0 and 1 alternately for display  
11      Reserved

Bit 31      L4C (L4 layer Color mode)  
Sets the color mode for L4 layer  
0      Indirect color (8 bits/pixel) mode  
1      Direct color (16 bits/pixel) mode

## L4EM (L4 layer Extended Mode)

Register address	DisplayBaseAddress + 150 <sub>H</sub>																													
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	---			4	3	2	1	0
Bit field name	L4EC		Reserved																								L4OM		L4WP	
R/W	RW		R0				RW		R0																		RW		RW	
Initial value	00		0				0		0																		0		0	

- Bit 0            L4 WP (L4 layer Window Position enable)  
Selects the display position of L4 layer  
0      Compatibility mode display (BL layer supported)  
1      Window display
- Bit 1            L4OM (L4 layer Overlay Mode)  
Selects the overlay mode for L4 layer  
0      Compatibility mode  
1      Extended mode
- Bit 23 to 20    L4PB (L4 layer Palette Base)  
Shows the value added to the index when subtracting palette of L4 layer. 16 times of setting value is added.
- Bit 31 and 30   L4EC (L4 layer Extended Color mode)  
Sets extended color mode for L4 layer  
00    Mode determined by L4C  
01    Direct color (24 bits/pixel) mode  
1x    Reserved

#### L4OA0 (L4 layer Origin Address 0)

Register address	DisplayBaseAddress + 74 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L4OA0																									
R/W	R0						RW																				R0					
Initial value	0						Don't care																				0000					

This register sets the origin address of the logic frame of the L4 layer in frame 0. Since lower 4 bits are fixed to "0", this address is 16-byte aligned.

#### L4DA0 (L4 layer Display Address 0)

Register address	DisplayBaseAddress + 78 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L4DA0																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the origin address of the L4 layer in frame 0. For the direct color mode (16 bits/pixel), the lower 1 bit is "0" and this address is 2-byte aligned.

#### L4OA1 (L4 layer Origin Address 1)

Register address	DisplayBaseAddress + 7C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L4OA1																									
R/W	R0						RW																				R0					
Initial value	0						Don't care																				0000					

This register sets the origin address of the logic frame of the L4 layer in frame 1. Since lower 4-bits are fixed to "0", this address is 16-byte aligned.

#### L4DA1 (L4 layer Display Address 1)

Register address	DisplayBaseAddress + 80 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L4DA1																									
R/W	R0						RW																									
Initial value	0						Don't care																									

This register sets the origin address of the L4 layer in frame 1. For the direct color mode (16 bits/pixel), the lower 1 bit is "0" and this address is 2-byte aligned.

#### L4DX (L4 layer Display position X)

Register address	DisplayBaseAddress + 84 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L4DX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (X coordinates) of the L4 layer on the basis of the origin of the logic frame in pixels.

#### L4DY (L4 layer Display position Y)

Register address	DisplayBaseAddress + 86 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L4DY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (Y coordinates) of the L4 layer on the basis of the origin of the logic frame in pixels.

#### L4WX (L4 layer Window position X)

Register address	DisplayBaseAddress + 154 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L4WX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the X coordinates of the display position of the L4 layer window.

#### L4WY (L4 layer Window position Y)

Register address	DisplayBaseAddress + 156 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L4WY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the Y coordinates of the display position of the L4 layer window.

#### L4WW (L4 layer Window Width)

Register address	DisplayBaseAddress + 158 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L4WW											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal direction display size (width) of the L4 layer window. Do not specify "0".

#### L4WH (L4 layer Window Height)

Register address	DisplayBaseAddress + 15A <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L4WH											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical direction display size (height) of the L4 layer window. Setting value + 1 is the height.



## L5M (L5 layer Mode)

Register address	DisplayBaseAddress + 88 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L5C		L5FLP		Reserved				L5W				Reserved				L5H															
R/W	RW		RW		R0				RW				R0				RW															
Initial value					0				Don't care				0				Don't care															

Bit 11 to 0      L5H (L5 layer Height)  
Specifies the height of the logic frame of the L5 layer in pixel units. Setting value + 1 is the height

Bit 23 to 16      L5W (L5 layer memory Width)  
Sets the memory width (stride) logic frame of the L5 layer in 64-byte units

Bit 30 and 29      L5FLP (L5 layer Flip mode)  
Sets flipping mode for L5 layer  
00      Displays frame 0  
01      Displays frame 1  
10      Switches frame 0 and 1 alternately for display  
11      Reserved

Bit 31      L5C (L5 layer Color mode)  
Sets the color mode for L5 layer  
0      Indirect color (8 bits/pixel) mode  
1      Direct color (16 bits/pixel) mode

## L5EM (L5 layer Extended Mode)

Register address	DisplayBaseAddress + 160 <sub>H</sub>																													
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	---			4	3	2	1	0
Bit field name	L5EC		Reserved																								L5OM		L5WP	
R/W	RW		R0																								RW		RW	
Initial value	00		0																										0	

Bit 0            L5 WP (L5 layer Window Position enable)  
Selects the display position of L5 layer  
0      Compatibility mode display (BR layer supported)  
1      Window display

Bit 1            L5OM (L5 layer Overlay Mode)  
Selects the overlay mode for L5 layer  
0      Compatibility mode  
1      Extended mode

Bit 31 to 30    L5EC (L5 layer Extended Color mode)  
Sets extended color mode for L5 layer  
00    Mode determined by L5C  
01    Direct color (24 bits/pixel) mode  
1x    Reserved

### L5OA0 (L5 layer Origin Address 0)

Register address	DisplayBaseAddress + 8C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L5OA0																									
R/W	R0						RW																							R0		
Initial value	0						Don't care																							0000		

This register sets the origin address of the logic frame of the L5 layer in frame 0. Since lower 4 bits are fixed to “0”, this address is 16-byte aligned.

### **L5DA0 (L5 layer Display Address 0)**

[illegible]

This register sets the origin address of the L5 layer in frame 0. For the direct color mode (16 bits/pixel), the lower 1 bit is “0” and this address is 2-byte aligned.

**L5OA1 (L5 layer Origin Address 1)**

Register address	DisplayBaseAddress + 94 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						L5OA1																									
R/W	R0						RW																								R0	
Initial value	0						Don't care																								0000	

This register sets the origin address of the logic frame of the L5 layer in frame 1. Since lower 4-bits are fixed to “0”, this address is 16-byte aligned.

### **L50A1 (L5 layer Display Address 1)**

[illegible]

This register sets the origin address of the L5 layer in frame 1. For the direct color mode (16 bits/pixel), the lower 1 bit is “0” and this address is 2-byte aligned.

**L5DX (L5 layer Display position X)**

[illegible]

This register sets the display starting position (X coordinates) of the L5 layer on the basis of the origin of the logic frame in pixels.

#### L5DY (L5 layer Display position Y)

Register address	DisplayBaseAddress + 9E <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L5DY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display starting position (Y coordinates) of the L5 layer on the basis of the origin of the logic frame in pixels.

#### L5WX (L5 layer Window position X)

Register address	DisplayBaseAddress + 164 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L5WX											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the X coordinates of the display position of the L5 layer window.

#### L5WY (L5 layer Window position Y)

Register address	DisplayBaseAddress + 166 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L5WY											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the Y coordinates of the display position of the L5 layer window.

#### L5WW (L5 layer Window Width)

Register address	DisplayBaseAddress + 168 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L5WW											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the horizontal direction display size (width) of the L5 layer window. Do not specify "0".

#### L5WH (L5 layer Window Height)

Register address	DisplayBaseAddress + 16A <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				L5WH											
R/W	R0				RW											
Initial value	0				Don't care											

This register controls the vertical direction display size (height) of the L5 layer window. Setting value + 1 is the height.

### CUTC (Cursor Transparent Control)

Register address	DisplayBaseAddress + A0 <sub>H</sub>																
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name	Reserved								CUZT	CUTC							
R/W	R0								RW	RW							
Initial value	0								Don't care	Don't care							

Bit 7 to 0     CUTC (Cursor Transparent Code)  
Sets color code handled as transparent code

Bit 8           CUZT (Cursor Zero Transparency)  
Defines handling of color code 0  
0     Code 0 as non-transparency color  
1     Code 0 as transparency color

### CPM (Cursor Priority Mode)

Register address	DisplayBaseAddress + A2 <sub>H</sub>							
Bit number	7	6	5	4	3	2	1	0
Bit field name	Reserved		CEN1	CEN0	Reserved		CUO1	CUO0
R/W	R0		RW	RW	R0		RW	RW
Initial value	0		0	0	0		0	0

This register controls the display priority of cursors.    Cursor 0 is always preferred to cursor 1.

Bit 0           CUO0 (Cursor Overlap 0)  
Sets display priority between cursor 0 and pixels of Console layer  
0     Puts cursor 0 at lower than L0 layer.  
1     Puts cursor 0 at higher than L0 layer.

Bit 1           CUO1 (Cursor Overlap 1)  
Sets display priority between cursor 1 and C layer  
0     Puts cursor 1 at lower than L0 layer.  
1     Puts cursor 1 at lower than L0 layer.

Bit 4           CEN0 (Cursor Enable 0)  
Sets enabling display of cursor 0  
0     Disabled  
1     Enabled

Bit 5           CEN1 (Cursor Enable 1)  
Sets enabling display of cursor 1  
0     Disabled  
1     Enabled

### CUOA0 (Cursor-0 Origin Address)

Register address	DisplayBaseAddress + A4 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						CUOA0																									
R/W	R0						RW																				R0					
Initial value	0						Don't care																				0000					

This register sets the start address of the cursor 0 pattern. Since lower 4 bits are fixed to "0", this address is 16-byte aligned.

### CUX0 (Cursor-0 X position)

Register address	DisplayBaseAddress + A8 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				CUX0											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display position (X coordinates) of the cursor 0 in pixels. The reference position of the coordinates is the top left of the cursor pattern.

### CUY0 (Cursor-0 Y position)

Register address	DisplayBaseAddress + Aa <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				CUY0											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display position (Y coordinates) of the cursor 0 in pixels. The reference position of the coordinates is the top left of the cursor pattern.

### CUOA1 (Cursor-1 Origin Address)

Register address	DisplayBaseAddress + AC <sub>H</sub>																																	
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved						CUOA1																											
R/W	R0						RW																										R0	
Initial value	0						Don't care																										0000	

This register sets the start address of the cursor 1 pattern. Since lower 4 bits are fixed to "0", this address is 16-byte aligned.

### CUX1 (Cursor-1 X position)

Register address	DisplayBaseAddress + B0 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				CUX1											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display position (X coordinates) of the cursor 1 in pixels. The reference position of the coordinates is the top left of the cursor pattern.

### CUY1 (Cursor-1 Y position)

Register address	DisplayBaseAddress + B2 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				CUY1											
R/W	R0				RW											
Initial value	0				Don't care											

This register sets the display position (Y coordinates) of the cursor 1 in pixels. The reference position of the coordinates is the top left of the cursor pattern.

## DLS (Display Layer Select)

Register address	DisplayBaseAddress + 180 <sub>H</sub>																													
Bit number	31	30	29	-----	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				DLS5				DLS4				DLS3				DLS2				DLS1				DSL0					
R/W	R0				R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW	R0	RW
Initial value							101					100				011					010				001				000	

This register defines the blending sequence.

- Bit 3 to 0      DSL0 (Display Layer Select 0)  
Selects the top layer subjected to blending.
- |      |              |
|------|--------------|
| 0000 | L0 layer     |
| 0001 | L1 layer     |
| :    | :            |
| 0101 | L5 layer     |
| 0110 | Reserved     |
| :    | :            |
| 0110 | Reserved     |
| 0111 | Not selected |
- Bit 7 to 4      DSL1 (Display Layer Select 1)  
Selects the second layer subjected to blending. The bit values are the same as DSL0.
- Bit 11 to 8      DSL2 (Display Layer Select 2)  
Selects the third layer subjected to blending. The bit values are the same as DSL0.
- Bit 15 to 12      DSL3 (Display Layer Select 3)  
Selects the fourth layer subjected to blending. The bit values are the same as DSL0.
- Bit 19 to 16      DSL4 (Display Layer Select 4)  
Selects the fifth layer subjected to blending. The bit values are the same as DSL0.
- Bit 23 to 20      DSL5 (Display Layer Select 5)  
Selects the bottom layer subjected to blending. The bit values are the same as DSL0.



### DBGC (Display Background Color)

Register address	DisplayBaseAddress + 184 <sub>H</sub>																															
Bit number	31	30	29	-----	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved				DBGR								DBGG								DBGB											
R/W	R0																															
Initial value																																

This register specifies the color to be displayed in areas outside the display area of each layer on the window.

Bit 7 to 0      DBGB (Display Background Blue)  
Specifies the blue level of the background color.

Bit 15 to 8    DBGG (Display Background Green)  
Specifies the green level of the background color.

Bit 23 to 16   DBGR (Display Background Red)  
Specifies the red level of the background color.

## L0BLD (L0 Blend)

Register address	DisplayBaseAddress + B4 <sub>H</sub>																											
Bit number	31	30	29	28	-----	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved					L0BE		L0BS		L0BI		L0BP		Reserved					L0BR									
R/W	R0					RW		RW		RW		RW		R0					RW									
Initial value						0		0		0		0							0									

This register specifies the blend parameters for the L0 layer. This register corresponds to BRATIO or BMODE for previous products.

- Bit 7 to 0      L0BR (L0 layer Blend Ratio)  
Sets the blend ratio. Basically, the blend ratio is setting value/256.
- Bit 13          L0BP (L0 layer Blend Plane)  
Specifies that the L5 layer is the blend plane.  
0      Value of L0BR used as blend ratio  
1      Pixel of L5 layer used as blend ratio
- Bit 14          L0BI (L0 layer Blend Increment)  
Selects whether or not 1/256 is added when the blend ratio is not "0".  
0      Blend ratio calculated as is  
1      1/256 added when blend ratio ≠ 0
- Bit 15          L0BS (L0 layer Blend Select)  
Selects the blend calculation expression.  
0      Upper image × Blend ratio + Lower image × (1 – Blend ratio)  
1      Upper image × (1 – Blend ratio) + Lower image × Blend ratio
- Bit 16          L0BE (L0 layer Blend Enable)  
This bit enables blending.  
0      Overlay via transparent color  
1      Overlay via blending

Before blending, the blend mode must be specified using L0BE, and alpha must also be enabled for L0 layer display data. For direct color, alpha is specified using the MSB of data; for indirect color, alpha is specified using the MSB of palette data.

## L1BLD (L1 Blend)

Register address	DisplayBaseAddress + 188 <sub>H</sub>																											
Bit number	31	30	29	28	-----	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved										L1BE	L1BS	L1BI	L1BP	Reserved							L1BR						
R/W	R0										RW	RW	RW	RW	R0							RW						
Initial value											0	0	0	0								0						

This register specifies the blend parameters for the L1 layer.

- Bit 7 to 0    L1BR (L1 layer Blend Ratio)  
Sets the blend ratio. Basically, the blend ratio is setting value/256.
- Bit 13       L1BP (L1 layer Blend Plane)  
Specifies that the L5 layer is the blend plane.  
0       Value of L1BR used as blend ratio  
1       Pixel of L5 layer used as blend ratio
- Bit 14       L1BI (L1 layer Blend Increment)  
Selects whether or not 1/256 is added when the blend ratio is not "0".  
0       Blend ratio calculated as is  
1       1/256 added when blend ratio ≠ 0
- Bit 15       L1BS (L1 layer Blend Select)  
Selects the blend calculation expression.  
0       Upper image × Blend ratio + Lower image × (1 – Blend ratio)  
1       Upper image × (1 – Blend ratio) + Lower image × Blend ratio
- Bit 16       L1BE (L1 layer Blend Enable)  
This bit enables blending.  
0       Overlay via transparent color  
1       Overlay via blending

Before blending, the blend mode must be specified using L1BE, and alpha must also be enabled for L1 layer display data. For direct color, alpha is specified using the MSB of data; for indirect color, alpha is specified using the MSB of palette data.

## L2BLD (L2 Blend)

Register address	DisplayBaseAddress + 18C <sub>H</sub>																											
Bit number	31	30	29	28	-----	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved					L2BE		L2BS		L2BI		L2BP		Reserved					L2BR									
R/W	R0					RW		RW		RW		RW		R0					RW									
Initial value						0		0		0		0							0									

This register specifies the blend parameters for the L2 layer.

- Bit 7 to 0    L2BR (L2 layer Blend Ratio)  
Sets the blend ratio. Basically, the blend ratio is setting value/256.
- Bit 13       L2BP (L2 layer Blend Plane)  
Specifies that the L5 layer is the blend plane.  
0       Value of L2BR used as blend ratio  
1       Pixel of L5 layer used as blend ratio
- Bit 14       L2BI (L2 layer Blend Increment)  
Selects whether or not 1/256 is added when the blend ratio is not "0".  
0       Blend ratio calculated as is  
1       1/256 added when blend ratio ≠ 0
- Bit 15       L2BS (L2 layer Blend Select)  
Selects the blend calculation expression.  
0       Upper image × Blend ratio + Lower image × (1 – Blend ratio)  
1       Upper image × (1 – Blend ratio) + Lower image × Blend ratio
- Bit 16       L2BE (L2 layer Blend Enable)  
This bit enables blending.  
0       Overlay via transparent color  
1       Overlay via blending

Before blending, the blend mode must be specified using L2BE, and alpha must also be enabled for L2 layer display data. For direct color, alpha is specified using the MSB of data; for indirect color, alpha is specified using the MSB of palette data.

### L3BLD (L3 Blend)

Register address	DisplayBaseAddress + 190 <sub>H</sub>																											
Bit number	31	30	29	28	-----	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved										L3BE	L3BS	L3BI	L3BP	Reserved					L3BR								
R/W											RW	Rw	RW	RW						RW								
Initial value											0	0	0	0						0								

This register specifies the blend parameters for the L3 layer.

- Bit 7 to 0    L3BR (L3 layer Blend Ratio)  
Sets the blend ratio. Basically, the blend ratio is setting value/256.
- Bit 13       L3BP (L3 layer Blend Plane)  
Specifies that the L5 layer is the blend plane.  
0       Value of L3BR used as blend ratio  
1       Pixel of L5 layer used as blend ratio
- Bit 14       L3BI (L3 layer Blend Increment)  
Selects whether or not 1/256 is added when the blend ratio is not "0".  
0       Blend ratio calculated as is  
1       1/256 added when blend ratio ≠ 0
- Bit 15       L3BS (L3 layer Blend Select)  
Selects the blend calculation expression.  
0       Upper image × Blend ratio + Lower image × (1 – Blend ratio)  
1       Upper image × (1 – Blend ratio) + Lower image × Blend ratio
- Bit 16       L3BE (L3 layer Blend Enable)  
This bit enables blending.  
0       Overlay via transparent color  
1       Overlay via blending

Before blending, the blend mode must be specified using L3BE, and alpha must also be enabled for L3 layer display data. For direct color, alpha is specified using the MSB of data; for indirect color, alpha is specified using the MSB of palette data.

## L4BLD (L4 Blend)

Register address	DisplayBaseAddress + 194 <sub>H</sub>																											
Bit number	31	30	29	28	-----	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	Reserved					L4BE		L4BS		L4BI		L4BP		Reserved					L4BR									
R/W	R0					RW		RW		RW		RW		R0					RW									
Initial value						0		0		0		0							0									

This register specifies the blend parameters for the L4 layer.

- Bit 7 to 0    L4BR (L4 layer Blend Ratio)  
Sets the blend ratio. Basically, the blend ratio is setting value/256.
- Bit 13       L4BP (L4 layer Blend Plane)  
Specifies that the L5 layer is the blend plane.  
0       Value of L4BR used as blend ratio  
1       Pixel of L5 layer used as blend ratio
- Bit 14       L4BI (L4 layer Blend Increment)  
Selects whether or not 1/256 is added when the blend ratio is not "0".  
0       Blend ratio calculated as is  
1       1/256 added when blend ratio ≠ 0
- Bit 15       L4BS (L4 layer Blend Select)  
Selects the blend calculation expression.  
0       Upper image × Blend ratio + Lower image × (1 – Blend ratio)  
1       Upper image × (1 – Blend ratio) + Lower image × Blend ratio
- Bit 16       L4BE (L4 layer Blend Enable)  
This bit enables blending.  
0       Overlay via transparent color  
1       Overlay via blending

Before blending, the blend mode must be specified using L4BE, and alpha must also be enabled for L4 layer display data. For direct color, alpha is specified using the MSB of data; for indirect color, alpha is specified using the MSB of palette data.

## L5BLD (L5 Blend)

Register address	DisplayBaseAddress + 198h																															
Bit number	31	30	29	28	-----	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
Bit field name	Reserved										L5BE	L5BS	L5BI	Reserved							L5BR											
R/W	R0										RW	RW	RW	R0							RW											
Initial value											0	0	0																			

This register specifies the blend parameters for the L5 layer.

Bit 7 to 0 L5BR (L5 layer Blend Ratio)

Sets the blend ratio. Basically, the blend ratio is setting value/256.

Bit 14 L5BI (L5 layer Blend Increment)

Selects whether or not 1/256 is added when the blend ratio is not "0".

0 Blend ratio calculated as is

1 1/256 added when blend ratio  $\neq$  0

Bit 15 L5BS (L5 layer Blend Select)

Selects the blend calculation expression.

0 Upper image  $\times$  Blend ratio + Lower image  $\times$  (1 – Blend ratio)

1 Upper image  $\times$  (1 – Blend ratio) + Lower image  $\times$  Blend ratio

Bit 16 L5BE (L5 layer Blend Enable)

This bit enables blending.

0 Overlay via transparent color

1 Overlay via blending

Before blending, the blend mode must be specified using L5BE, and alpha must also be enabled for L5 layer display data. For direct color, alpha is specified using the MSB of data; for indirect color, alpha is specified using the MSB of palette data.

### L0TC (L0 layer Transparency Control)

Register address	DisplayBaseAddress + BC <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L0ZT		L0TC													
R/W	RW		RW													
Initial value	0		0													

This register sets the transparent color for the L0 layer. Color set by this register is transparent in blend mode. When L0TC = 0 and L0ZT = 0, color 0 is displayed in black (transparent).

This register corresponds to the CTC register for previous products.

Bit 14 to 0    L0TC (L0 layer Transparent Color)  
Sets transparent color code for the L0 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 15        L0ZT (L0 layer Zero Transparency)  
Sets handling of color code 0 in L0 layer  
0:    Code 0 as transparency color  
1:    Code 0 as non-transparency color

### L2TC (L2 layer Transparency Control)

Register address	DisplayBaseAddress + C2 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L2ZT		L2TC													
R/W	RW		RW													
Initial value	0		0													

This register sets the transparent color for the L2 layer.

When L2TC = 0 and L2ZT = 0, color 0 is displayed in black (transparent).

This register corresponds to the MLTC register for previous products.

Bit 14 to 0    L2TC (L2 layer Transparent Color)  
Sets transparent color code for the L2 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 15        L2ZT (L2 layer Zero Transparency)  
Sets handling of color code 0 in L2 layer  
0    Code 0 as transparency color  
1    Code 0 as non-transparency color



### L3TC (L3 layer Transparency Control)

Register address	DisplayBaseAddress + C0 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L3ZT	L3TC														
R/W	RW	RW														
Initial value	0	0														

This register sets the transparent color for the L3 layer. When L3TC = 0 and L3ZT = 0, color 0 is displayed in black (transparent).

This register corresponds to the MLTC register for previous products.

Bit 14 to 0 L3TC (L3 layer Transparent Color)

Sets transparent color code for the L3 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 15 L3ZT (L3 layer Zero Transparency)

Sets handling of color code 0 in L3 layer

0 Code 0 as transparency color

1 Code 0 as non-transparency color

### L0ETC (L0 layer Extend Transparency Control)

Register address	DisplayBaseAddress + 1A0 <sub>H</sub>																													
Bit number	31	30	29	28	---	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	L0ETZ	Reserved					L0TEC																							
R/W	RW	R0					RW																							
Initial value	0						0																							

This register sets the transparent color for the L0 layer. The 24 bits/pixel transparent color is set using this register. The lower 15 bits of this register are physically the same as L0TC. Also, L0ETZ is physically the same as L0TZ.

When L0ETC = 0 and L0EZT = 0, color 0 is displayed in black (transparent).

Bit 23 to 0 L0ETC (L0 layer Extend Transparent Color)

Sets transparent color code for the L0 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 31 L0EZT (L0 layer Extend Zero Transparency)

Sets handling of color code 0 in L0 layer

0 Code 0 as transparency color

1 Code 0 as non-transparency color

### L1ETC (L1 layer Extend Transparency Control)

Register address	DisplayBaseAddress + 1A4 <sub>H</sub>																															
Bit number	31	30	29	28	---	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	L1ETZ		Reserved										L1TEC																			
R/W	RW		R0										RW																			
Initial value	0												0																			

This register sets the transparent color for the L1 layer. When L1ETC = 0 and L1EET = 0, color 0 is displayed in black (transparent).

For YCbCr display, transparent color checking is not performed; processing is always performed assuming that transparent color is not used.

Bit 23 to 0 L1ETC (L1 layer Extend Transparent Color)

Sets transparent color code for the L1 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 31 L1EET (L1 layer Extend Zero Transparency)

Sets handling of color code 0 in L1 layer

0 Code 0 as transparency color

1 Code 0 as non-transparency color

### L2ETC (L2 layer Extend Transparency Control)

Register address	DisplayBaseAddress + 1A8 <sub>H</sub>																															
Bit number	31	30	29	28	---	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	L2ETZ		Reserved										L2TEC																			
R/W	RW		R0										RW																			
Initial value	0												0																			

This register sets the transparent color for the L2 layer. The 24 bits/pixel transparent color is set using this register. The lower 15 bits of this register are physically the same as L2TC. Also, L2ETZ is physically the same as L2TZ.

When L2ETC = 0 and L2EET = 0, color 0 is displayed in black (transparent).

Bit 23 to 0 L2ETC (L2 layer Extend Transparent Color)

Sets transparent color code for the L2 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 31 L2EET (L2 layer Extend Zero Transparency)

Sets handling of color code 0 in L2 layer

0 Code 0 as transparency color

1 Code 0 as non-transparency color

### L3ETC (L3 layer Extend Transparency Control)

Register address	DisplayBaseAddress + 1AC <sub>H</sub>																															
Bit number	31	30	29	28	---	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	L3ETZ		Reserved										L3TEC																			
R/W	RW		R0										RW																			
Initial value	0												0																			

This register sets the transparent color for the L3 layer. The 24 bits/pixel transparent color is set using this register. The lower 15 bits of this register are physically the same as L3TC. Also, L3ETZ is physically the same as L3TZ.

When L3ETC = 0 and L3EETZ = 0, color 0 is displayed in black (transparent).

- Bit 23 to 0    L3ETC (L3 layer Extend Transparent Color)  
Sets transparent color code for the L3 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.
- Bit 31        L3EETZ (L3 layer Extend Zero Transparency)  
Sets handling of color code 0 in L3 layer
- 0    Code 0 as transparency color
- 1    Code 0 as non-transparency color

### L4ETC (L4 layer Extend Transparency Control)

Register address	DisplayBaseAddress + 1B0 <sub>H</sub>																															
Bit number	31	30	29	28	---	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	L4ETZ		Reserved										L4TEC																			
R/W	RW		R0										RW																			
Initial value	0												0																			

This register sets the transparent color for the L4 layer. This register sets the transparent color for the L4 layer. When L4ETC = 0 and L4EETZ = 0, color 0 is displayed in black (transparent).

- Bit 23 to 0    L4ETC (L4 layer Extend Transparent Color)  
Sets transparent color code for the L4 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.
- Bit 31        L4EETZ (L4 layer Extend Zero Transparency)  
Sets handling of color code 0 in L4 layer
- 0    Code 0 as transparency color
- 1    Code 0 as non-transparency color

## L5ETC (L5 layer Extend Transparency Control)

Register address	DisplayBaseAddress + 1B4 <sub>H</sub>																															
Bit number	31	30	29	28	---	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name	L5ETZ		Reserved										L5TEC																			
R/W	RW		R0										RW																			
Initial value	0												0																			

This register sets the transparent color for the L5 layer. This register sets the transparent color for the L5 layer. When L5ETC = 0 and L5EZE = 0, color 0 is displayed in black (transparent).

Bit 23 to 0 L5ETC (L5 layer Extend Transparent Color)

Sets transparent color code for the L5 layer. In indirect color mode (8 bits/pixel) bits 7 to 0 are used.

Bit 31 L5EZE (L5 layer Extend Zero Transparency)

Sets handling of color code 0 in L5 layer

0 Code 0 as transparency color

1 Code 0 as non-transparency color

### L0PAL0-255 (L0 layer Palette 0-255)

Register address	DisplayBaseAddress + 400 <sub>H</sub> -- DisplayBaseAddress + 7FF <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	A								R								G								B							
R/W	RW	R0							RW							R0	RW							R0	RW							R0
Initial value	Don't care	0000000							Don't care							00	Don't care							00	Don't care							00

These are color palette registers for L0 layer and cursors. In the indirect color mode, a color code in the display frame indicates the palette register number, and the color information set in that register is applied as the display color of that pixel. This register corresponds to the CPAL<sub>n</sub> register for previous products.

- Bit 7 to 2      B (Blue)  
Sets blue color component
  
- Bit 15 to 10    G (Green)  
Sets green color component
  
- Bit 23 to 18    R (Red)  
Sets red color component
  
- Bit 31          A (Alpha)  
Specifies whether or not to perform blending with lower layers when the blending mode is enabled.
  - 0      Blending not performed even when blending mode enabled  
Overlay is performed via transparent color.
  - 1      Blending performed

### L1PAL0-255 (L1 layer Palette 0-255)

Register address	DisplayBaseAddress + 800 <sub>H</sub> -- DisplayBaseAddress + BFF <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	A								R								G								B							
R/W	RW	R0							RW				R0				RW				R0				RW				R0			
Initial value	Don't care	00000000							Don't care				00				Don't care				00				Don't care				00			

These are color palette registers for L1 layer and cursors. In the indirect color mode, a color code in the display frame indicates the palette register number, and the color information set in that register is applied as the display color of that pixel. This register corresponds to the MBPALn register for previous products.

- Bit 7 to 2      B (Blue)  
Sets blue color component
  
- Bit 15 to 10    G (Green)  
Sets green color component
  
- Bit 23 to 18    R (Red)  
Sets red color component
  
- Bit 31          A (Alpha)  
Specifies whether or not to perform blending with lower layers when the blending mode is enabled.
  - 0      Blending not performed even when blending mode enabled  
Overlay is performed via transparent color.
  - 1      Blending performed

## L2PAL0-255 (L2 layer Palette 0-255)

Register address	DisplayBaseAddress + 1000 <sub>H</sub> -- DisplayBaseAddress + 13FF <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	A								R								G								B							
R/W	RW		R0						RW		R0						RW		R0						RW		R0					
Initial value	Don't care		00000000						Don't care		00						Don't care		00						Don't care		00					

These are color palette registers for L2 layer and cursors. In the indirect color mode, a color code in the display frame indicates the palette register number, and the color information set in that register is applied as the display color of that pixel.

- Bit 7 to 2      B (Blue)  
Sets blue color component
  
- Bit 15 to 10    G (Green)  
Sets green color component
  
- Bit 23 to 18    R (Red)  
Sets red color component
  
- Bit 31          A (Alpha)  
Specifies whether or not to perform blending with lower layers when the blending mode is enabled.
  - 0      Blending not performed even when blending mode enabled  
Overlay is performed via transparent color.
  - 1      Blending performed

### L3PAL0-255 (L3 layer Palette 0-255)

Register address	DisplayBaseAddress + 1400 <sub>H</sub> -- DisplayBaseAddress + 17FF <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	A								R								G								B							
R/W	RW	R0							RW				R0				RW				R0				RW				R0			
Initial value	Don't care	00000000							Don't care				00				Don't care				00				Don't care				00			

These are color palette registers for L3 layer and cursors. In the indirect color mode, a color code in the display frame indicates the palette register number, and the color information set in that register is applied as the display color of that pixel.

- Bit 7 to 2      B (Blue)  
Sets blue color component
  
- Bit 15 to 10    G (Green)  
Sets green color component
  
- Bit 23 to 18    R (Red)  
Sets red color component
  
- Bit 31          A (Alpha)  
Specifies whether or not to perform blending with lower layers when the blending mode is enabled.
  - 0      Blending not performed even when blending mode enabled  
Overlay is performed via transparent color.
  - 1      Blending performed



## 11.2.4 Video Capture Registers

### VCM (Video Capture Mode)

Register address	CaputureBaseAddress + 00 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	VIE	VIS	Reserved				CM	Reserved				VI	Reserved																		VS	Rsv
R/W	RW	RW	RX				RW	RX				RW	RX																		RW	RX
Initial value	0		X				00	X				0	X																		0	X

This register sets the video capture mode.

- Bit 31      VIE (Video Input Enable)  
Enables video capture function  
0:      Does not capture video  
1:      Captures video
- Bit 30      VIS (Video Input Select)  
0      RBT656  
1      RGB666
- Bit 25 to 24      CM (Capture Mode)  
Sets video capture mode  
To capture vides, set these bits to "11".  
00:      Initial value  
01:      Reserved  
10:      Reserved  
11:      Capture
- Bit 20      VI (Vertical Interpolation)  
Sets whether to perform vertical interpolation  
0:      Performs vertical interpolation  
         The graphics are enlarged vertically by two times  
1:      Does not perform vertical interpolation
- Bit 1      VS (Video Select)  
Selects NTSC or PAL  
0:      NTSC  
1:      PAL

### CSC (Capture SScale)

Register address	CaputureBaseAddress + 04 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	VSCI					VSCF											HSCI					HSCF										
R/W	RW					RW											RW					RW										
Initial value	00001					00000000000											00001					00000000000										

This register sets the video capture enlargement/reduction ratio.

- Bit 31 to 27 VSCI (Vertical SScale Integer)  
Sets integer part of vertical enlargement/reduction ratio
- Bit 26 to 16 VSCF (Vertical SScale Fraction)  
Sets fraction part of vertical enlargement/reduction ratio
- Bit 15 to 11 HSCI (Horizontal SScale Integer)  
Sets integer part of horizontal enlargement/reduction ratio
- Bit 10 to 0 HSCF (Horizontal SScale Fraction)  
Sets fraction part of horizontal enlargement/reduction ratio

### VCS (Video Capture Status)

Register address	CaputureBaseAddress + 08 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved																										CE					
R/W	RX																										RW					
Initial value	Don't care																										00000					

This register indicates the ITU-RBT656 SAV and EAV status.

To detect error codes, set NTSC/PAL in the VS bit of VCM. If NTSC is set, reference the number of data in the capture data count register (CDCN). If PAL is set, reference the number of data in the capture data counter register (CDCP). If the reference data does not match the stream data, or undefined Fourth word of SAV/EAV codes are detected, bits 4 to 0 of the video capture status register (VCS) will be values as follows.

Bits 4-0 CE (Capture Error)

Indicates error occurred during video capture

Bit4	1	:	RBT.656	H code error (End)	0	:	true
Bit3	1	:	RBT.656	H code error (Start)	0	:	true
Bit2	1	:	RBT.656	undefined error (Code Bit7-0)	0	:	true
Bit1	1	:	RBT.656	undefined error (Code Bit7-4)	0	:	true
Bit0	1	:	RBT.656	undefined error (Code Bit7)	0	:	true

### CBM (vide Capture Buffer Mode)

Register address	CaputureBaseAddress + 10 <sub>H</sub>																															
Bit #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	00		Reserved										CBW				Reserved															
R/W	RW		RX										RW				Rx															
Initial value	Don't care												Don't care				Don't care															

Bit 23 to 16    CBW (Capture Buffer memory Width)  
Sets memory width (stride) of capture buffer in 64 bytes

Bit 31            OO (Odd Only mode)  
Specifies whether to capture odd fields only  
0:    Normal mode  
1:    Odd only mode

### CBOA (video Capture Buffer Origin Address)

Register address	CaputureBaseAddress + 14 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved		CBOA																													
R/W	RX		RW																				R0									
Initial value	Don't care		Don't care																				0									

This register specifies the starting (origin) address of the video capture buffer.

### CBLA (video Capture Buffer Limit Address)

Register address	CaputureBaseAddress + 18 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved		CBLA																													
R/W	RX		RW																				R0									
Initial value	Don't care		Don't care																				0									

This register specifies the end (limit) address of the video capture buffer.

CBLA must be larger than CBOA.

#### CIHSTR (Capture Image Horizontal STaRt)

Register address	CaputureBaseAddress + 1C <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						CIHSTR									
R/W	RX						RW									
Initial value	Don't care						Don't care									

This register sets the range of the images to be written (captured) to the video capture buffer. Specify the X coordinates located in the top left of the image range as the count of pixels from the top left of the image. For reduction, apply this setting to the post-reduction image coordinates.

#### CIVSTR (Capture Image Vertical STaRt)

Register address	CaputureBaseAddress + 1E <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						CIVSTR									
R/W	RX						RW									
Initial value	Don't care						Don't care									

This register sets the range of the images to be written (captured) to the video capture buffer. Specify the Y coordinates located in the top left of the image range as the count of pixels from the top left of the image. For reduction, apply this setting to the post-reduction image coordinates.

#### CIHEND (Capture Image Horizontal END)

Register address	CaputureBaseAddress + 20 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						CIHEND									
R/W	RX						RW									
Initial value	Don't care						Don't care									

This register sets the range of the images to be written (captured) to the video capture buffer. Specify the X coordinates located in the bottom right of the image range as the count of pixels from the top left of the image. For reduction, apply this setting to the post-reduction image coordinates.

If the pixel at the right end of the image is not aligned on 64 bits/word boundary, extra data is written before 64 bits/word boundary.

If the width of the input image is less than the range set by this command, data is written only at the size of input image.

#### CIVEND (Capture Image Vertical END)

Register address	CaputureBaseAddress + 22 <sub>H</sub>															
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved						CIVEND									
R/W	RX						RW									
Initial value	Don't care						Don't care									

This register sets the range of the images to be written (captured) to the video capture buffer. Specify the Y coordinates located in the bottom right of the image range as the count of pixels from the top left of the original image to be input. For reduction, apply this setting to the post-reduction image coordinates.

If the count of rasters of the input image is less than the range set by this command, data is written only at the size of the input image.

### CHP (Capture Horizontal Pixel)

Register address	CaputureBaseAddress + 28 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved																CHP															
R/W	RX																RW															
Initial value	X																168 <sub>H</sub> (360 <sub>D</sub> )															

This register sets the count of horizontal pixels of the image output after scaling. Specify the count of horizontal pixels in 2 pixels.

### CVP (Capture Vertical Pixel)

Register address	CaputureBaseAddress + 2C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved				CVPP												Reserved				CVPN											
R/W	RX				RW												RX				RW											
Initial value	X				271 <sub>H</sub> (625 <sub>D</sub> )												X				20D <sub>H</sub> (525 <sub>D</sub> )											

This register sets the count of vertical pixels of the image output after scaling. The fields to be used depend on the video format to be used.

- Bit 25 to 16    CVPP (Capture Vertical Pixel for PAL)  
Set count of vertical pixels of output image in PAL format used
  
- Bit 9 to 0     CVPN (Capture Vertical Pixel for NTSC)  
Set count of vertical pixels of output image in NTSC format used

## CLPF (Capture Low Pass Filter)

Register address	CaputureBaseAddress + 40 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserve				CVLPF				Reserve				CHLPF				Reserve															
R/W	R0				R/W				R0				R/W				R0															
Initial value	0				0				0				0				0															

This register sets the Low Pass Filter Coefficient. It specifies independently in 2-bit coefficient code with a luminance signal (Y) and a color-difference signal (C). A coefficient is a right-and-left symmetrical coefficient.

A Vertical low path filter consists of FIR filters of three taps. A coefficient is specified in the following register.

Bit 27 to 26 CVLPF\_Y (Capture Vertical LPF coefficient Y)

Sets Y part of vertical LPF coefficient code

CVLPF_Y	K0	K1	K2
2'b00	0	1	0
2'b01	1/4	2/4	1/4
2'b10	3/16	10/16	3/16
2'b11	Reserve		

Bit 25 to 24 CVLPF\_C (Capture Vertical LPF coefficient C)

Sets C part of vertical LPF coefficient code

CVLPF_C	K0	K1	K2
2'b00	0	1	0
2'b01	1/4	2/4	1/4
2'b10	3/16	10/16	3/16
2'b11	Reserve		

A horizontal low path filter consists of FIR filters of five taps. A coefficient is specified in the following register.

Bit 19 to 18 CHLPF\_YI (Capture Horizontal LPF coefficient Y)

Sets Y part of horizontal coefficient code

CHLPF_Y	K0	K1	K2	K3	K4
2'b00	0	0	1	0	0
2'b01	0	1/4	2/4	1/4	0
2'b10	0	3/16	10/16	3/16	0
2'b11	3/32	8/32	10/32	10/32	3/32

Bit 17 to 16 CHLPF\_C (Capture Horizontal LPF coefficient C)

Sets C part of horizontal coefficient code

CHLPF_C	K0	K1	K2	K3	K4
2'b00	0	0	1	0	0
2'b01	0	1/4	2/4	1/4	0
2'b10	0	3/16	10/16	3/16	0
2'b11	3/32	8/32	10/32	10/32	3/32

LPF will be turned off if coefficient code 2'b00 are set up.

### CDCN (Capture Data Count for NTSC)

Register address	CaputureBaseAddress + 4000 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved								BDCN								Reserved								VDCN							
R/W	RX								RW								RX								RW							
Initial value	X								10f <sub>H</sub> (271 <sub>D</sub> )								X								5A3 <sub>H</sub> (1443)							

This register sets the count of data of the input video stream in NTSC format.

Bit 25 to 16    BDCN (Blanking Data Count for NTSC)  
Sets count of data processed during blanking period in NTSC format

Bit 10 to 0    VDCN (Valid Data Count for NTSC)  
Sets count of data processed during valid period in NTSC format

### CDCP (Capture Data Count for PAL)

Register address	CaputureBaseAddress + 4004 <sub>H</sub>																															
Bit #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	Reserved								BDCP								Reserved								VDCP							
R/W	RX								RW								RX								RW							
Initial value	X								11B <sub>H</sub> (283 <sub>D</sub> )								X								5A3 <sub>H</sub> (1443)							

This register sets the count of data of the input video stream in PAL format.

Bit 25 to 16    BDCP (Blanking Data Count for PAL)  
Sets count of data processed during blanking period in PAL format

Bit 10 to 0    VDCP (Valid Data Count for PAL)  
Sets count of data processed during valid period in PAL format

## 11.2.5 Drawing control registers

### CTR (Control Register)

Register address	DrawBaseAddress + 400 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name								FO		CE		FCNT						NF	FF	FE		SS		DS		PS						
R/W								RW		RW		R						R	R	R		R		R		R						
Initial value								0		0		011101						0	0	1		00		00		00						

This register indicates drawing flags and status information. Bits 24 to 22 are not cleared until 0 is set.

Bit 1 and 0 PS (Pixel engine Status)

Indicate status of pixel engine unit

- 00 Idle
- 01 Busy
- 10 Reserved
- 11 Reserved

Bit 5 and 4 DS (DDA Status)

Indicate status of DDA

- 00 Idle
- 01 Busy
- 10 Busy
- 11 Reserved

Bit 9 and 8 SS (Setup Status)

Indicate status of Setup unit

- 00 Idle
- 01 Busy
- 10 Reserved
- 11 Reserved

Bit 12 FE (FIFO Empty)

Indicates whether data contained or not in display list FIFO

- 0 Valid data
- 1 No valid data

Bit 13 FF (FIFO Full)

Indicates whether display list FIFO is full or not

- 0 Not full
- 1 Full

Bit 14 NF (FIFO Near Full)

Indicates how empty the display list FIFO is



	0	Empty entries equal to or more than half
	1	Empty entries less than half
Bit 20 to 15	FCNT (FIFO Counter)	
	Indicates count of empty entries of display list FIFO (0 to 100000 <sub>H</sub> )	
Bit 22	CE (Display List Command Error)	
	Indicates command error occurrence (Not all error can detect. Need software reset or hardware reset for recovery)	
	0	Normal
	1	Command error detected
Bit 24	FO (FIFO Overflow)	
	Indicates FIFO overflow occurrence	
	0	Normal
	1	FIFO overflow detected

#### IFSR (Input FIFO Status Register)

Register address	DrawBaseAddress + 404 <sub>H</sub>																																	
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name																																NF	FF	FE
R/W																																R	R	R
Initial value																																0	0	1

This is a mirror register for bits 14 to 12 of the CTR register.

#### IFCNT (Input FIFO Counter)

Register address	DrawBaseAddress + 408 <sub>H</sub>																																
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name																												FCNT					
R/W																												R					
Initial value																												011101					

This is a mirror register for bits 19 to 15 of the CTR register.

#### SST (Setup engine Status)

Register address	DrawBaseAddress + 40C <sub>H</sub>																																
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name																																	SS
R/W																																	R
Initial value																																	00

This is a mirror register for bits 9 to 8 of the CTR register.

#### DST (DDA Status)

Register address	DrawBaseAddress + 410 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																																DS
R/W																																RW
Initial value																																00

This is a mirror register for bits 5 to 4 of the CTR register.

#### PST (Pixel engine Status)

Register address	DrawBaseAddress + 414 <sub>H</sub>																																
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Bit field name																																	PS
R/W																																	R
Initial value																																	00

This is a mirror register for bits 1 to 0 of the CTR register.

#### EST (Error Status)

Register address	DrawBaseAddress + 418 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																																
R/W																																
Initial value																																

This is a mirror register for bits 24 to 22 of the CTR register.

## 11.2.6 Drawing mode registers

When write to the registers, use the **SetRegister** command. The registers cannot be accessed from the CPU.

### MDR0 (Mode Register for miscellaneous)

Register address	DrawBaseAddress + 420 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name												ZP					CF						CY	CX					BSV	BSH		
R/W	W0											RW	W0				RW	W0					RW	RW	W0				RW	RW		
Initial value												0					00						0	0					00	00		

Bit 1 to 0      BSH (Bitmap Scale Horizontal)  
Sets horizontal zoom ratio of bitmap draw  
00    x1  
01    x2  
10    x1/2  
01    Reserved

Bit 3 to 2      BSV (Bitmap Scale Vertical)  
Sets vertical zoom ratio of bitmap draw  
00    x1  
01    x2  
10    x1/2  
01    Reserved

Bit 8            CX (Clip X enable)  
Sets X coordinates clipping mode  
0    Disabled  
1    Enabled

Bit 9            CY (Clip Y enable)  
Sets Y coordinates clipping mode  
0    Disabled  
1    Enabled

Bit 16 and 15   CF (Color Format)  
Sets drawing color format  
00    Indirect color mode (8 bits/pixel)  
01    Direct color mode (16 bits/pixel)

Bit 20          ZP (Z Precision)  
Sets the precision of the Z value used for erasing hidden planes.  
16 bits/pixel  
8 bits/pixel

## MDR1 (Mode Register for LINE)

Register address	DrawBaseAddress + 424 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name			LW								BP		BL					LOG				BM		ZW	ZCL		ZC					
R/W			RW								RW		RW					RW				RW		RW	RW		RW					
Initial value			00000								0		0					0011				0		0	0000		0					

This register sets the mode of line and pixel drawing.

Please set ZC bit ( bit 2 ) to 0 when draw BitCopyAltAlphaBlendP command.

Bit 2	ZC (Z Compare mode) Sets Z comparison mode 0 Disabled 1 Enabled
Bit 5 to 3	ZCL (Z Compare Logic) Selects type of Z comparison 000 NEVER 001 ALWAYS 010 LESS 011 LEQUAL 100 EQUAL 101 GEQUAL 110 GREATER 111 NOTEQUAL
Bit 6	ZW (Z Write mode) Sets Z write mode 0 Writes Z values. 1 Not write Z values.
Bit 8 to 7	BM (Blend Mode) Sets blend mode 00 Normal (source copy) 01 Alpha blending 10 Drawing with logic operation 11 Reserved
Bit 12 to 9	LOG (Logical operation) Sets type of logic operation 0000 CLEAR 0001 AND 0010 AND REVERSE 0011 COPY 0100 AND INVERTED

	0101	NOP
	0110	XOR
	0111	OR
	1000	NOR
	1001	EQUIV
	1010	INVERT
	1011	OR REVERSE
	1100	COPY INVERTED
	1101	OR INVERTED
	1110	NAND
	1111	SET
Bit 19	BL (Broken Line)	
	Selects line type	
	0	Solid line
	1	Broken line
Bit 20	BP (Broken line Period)	
	Selects broken line cycle	
	0:	32 bits
	1:	24 bits
Bit 28 to 24	LW (Line Width)	
	Sets line width for drawing line	
	00000	1 pixel
	00001	2 pixels
	:	:
	11111	32 pixels

## MDR2 (Mode Register for Polygon)

Register address	DrawBaseAddress + 428 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name			TT										LOG				BM		ZW	ZCL		ZC	SM									
R/W	W0		RW										RW				RW		RW	RW		RW	RW									
Initial value			00										0011				0		0	0000		0	0									

This register sets the polygon drawing mode.

Bit 0	SM (Shading Mode)
	Sets shading mode
	0 Flat shading
	1 Gouraud shading
Bit 2	ZC (Z Compare mode)
	Sets Z comparison mode
	0 Disabled
	1 Enabled
Bit 5 to 3	ZCL (Z Compare Logic)
	Selects type of Z comparison
	000 NEVER
	001 ALWAYS
	010 LESS
	011 LEQUAL
	100 EQUAL
	101 GEQUAL
	110 GREATER
	111 NOTEQUAL
Bit 6	ZW (Z Write mask)
	Sets Z write mode
	0 Writes Z values
	1 Not write Z values
Bit 8 to 7	BM (Blend Mode)
	Sets blend mode
	00 Normal (source copy)
	01 Alpha blending
	10 Drawing with logic operation
	11 Reserved
Bit 12 to 9	LOG (Logical operation)
	Sets type of logic operation

0000	CLEAR
0001	AND
0010	AND REVERSE
0011	COPY
0100	AND INVERTED
0101	NOP
0110	XOR
0111	OR
1000	NOR
1001	EQUIV
1010	INVERT
1011	OR REVERSE
1100	COPY INVERTED
1101	OR INVERTED
1110	NAND
1111	SET

Bit 29 to 28	TT (Texture-Tile Select)	
	Selects texture or tile pattern	
	00	Neither used
	01	Enabled tiling
	10	Enabled texture
	11	Reserved

### MDR3 (Mode Register for Texture)

Register address	DrawBaseAddress + 42C <sub>H</sub>																																	
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
Bit field name											TAB				TBL						TWS		TWT				TF		TC					
R/W											RW				RW						RW		RW				RW		RW					
Initial value											00				00						00		00				0				0			

This register sets the texture mapping mode.

- Bit 3 TC (Texture coordinates Correct)  
Sets texture coordinates correction mode  
0 Disabled  
1 Reserved
- Bit 5 TF (Texture Filtering)  
Sets type of texture interpolation (filtering)  
0 Point sampling  
1 Bi-linear filtering
- Bit 9 and 8 TWT (Texture Wrap T)  
Sets type of texture coordinates T direction wrapping  
00 Cramp  
01 Repeat  
10 Border  
11 Reserved
- Bit 11 and 10 TWS (Texture Wrap S)  
Sets type of texture coordinates S direction wrapping  
00 Repeat  
01 Cramp  
10 Border  
11 Reserved
- Bit 17 and 16 TBL (Texture Blend mode)  
Sets texture blending mode  
00 De-curl  
01 Modulate  
10 Stencil  
11 Reserved
- Bit 21 and 20 TAB (Texture Alpha Blend mode)



Sets texture blending mode

The stencil mode and the stencil alpha mode are enabled only when the MDR2 register blend mode (BM) is set to the alpha blending mode. If it is not set to the alpha blending mode, the stencil mode and stencil alpha mode perform the same function as the normal mode.

- 00 Normal
- 01 Stencil
- 10 Stencil alpha
- 11 Reserved

#### MDR4 (Mode Register for BLT)

Register address	DrawBaseAddress + 430 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																					LOG		BM						TE			
R/W																					RW		RW						RW			
Initial value																					0011		00						0			

This register controls the BLT mode.

- Bit 1      TE (Transparent Enable)  
Sets transparent mode
- 0:      Not perform transparent processing
- 1:      Not draw pixels that corresponds to set transparent color in BLT (transparency copy)
- Note:    Set the blend mode (BM) to normal.
- 
- Bit 8 to 7    BM (Blend Mode)  
Sets blend mode
- 00      Normal (source copy)
- 01      Reserved
- 10      Drawing with logic operation
- 11      Reserved
- 
- Bit 12 to 9    LOG (Logical operation)  
Sets logic operation
- 0000    CLEAR
- 0001    AND
- 0010    AND REVERSE
- 0011    COPY
- 0100    AND INVERTED
- 0101    NOP
- 0110    XOR
- 0111    OR
- 1000    NOR
- 1001    EQUIV
- 1010    INVERT
- 1011    OR REVERSE
- 1100    COPY INVERTED
- 1101    OR INVERTED
- 1110    NAND
- 1111    SET

### FBR (Frame buffer Base)

Register address	DrawBaseAddress + 440 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name							FBASE																									
R/W							RW																		R0							
Initial value							Don't care																		0							

This register stores the base address of the drawing frame.

### XRES (X Resolution)

Register address	DrawBaseAddress + 444 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																					XRES											
R/W																					RW											
Initial value																					Don't care											

This register sets the drawing frame horizontal resolution.

### ZBR (Z buffer Base)

Register address	DrawBaseAddress + 448 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name							ZBASE																									
R/W							RW																R0									
Initial value							Don't care																0									

This register sets the Z buffer base address.

### TBR (Texture memory Base)

Register address	DrawBaseAddress + 44C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name							TBASE																									
R/W							RW																R0									
Initial value							Don't care																0									

This register sets the texture memory base address.

### PFBR (2D Polygon Flag-Buffer Base)

Register address	DrawBaseAddress + 450 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name							PFBASE																									
R/W							RW																R0									
Initial value							Don't care																0									

This register sets the polygon flag buffer base address.

### CXMIN (Clip X minimum)

Register address	DrawBaseAddress + 454 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																					CLIPXMIN											
R/W																					RW											
Initial value																					Don't care											

This register sets the clip frame minimum X position.

### CXMAX (Clip X maximum)

Register address	DrawBaseAddress + 458 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																					CLIPXMAX											
R/W																					RW											
Initial value																					Don't care											

This register sets the clip frame maximum X position.

### CYMIN (Clip Y minimum)

Register address	DrawBaseAddress + 45C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																					CLIPYMIN											
R/W																					RW											
Initial value																					Don't care											

This register sets the clip frame minimum Y position.

### CYMAX (Clip Y maximum)

Register address	DrawBaseAddress + 460 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																					CLIPYMAX											
R/W																					RW											
Initial value																					Don't care											

This register sets the clip frame maximum Y position.

## TXS (Texture Size)

Register address	DrawBaseAddress + 464 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	TXSN																TXSM															
R/W	RW																RW															
Initial value	000010000000																000010000000															

This register specifies the texture size (m, n).

### Bit 12 to 0 TXSM (Texture Size M)

Sets horizontal texture size. Any power of 2 between 4 and 4096 can be used. Values that are not a power of 2 cannot be used.

0_0000_0000_0100	M=4	0_0010_0000_0000	M=512
0_0000_0000_1000	M=8	0_0100_0000_0000	M=1024
0_0000_0001_0000	M=16	0_1000_0000_0000	M=2048
0_0000_0010_0000	M=32	1_0000_0000_0000	M=4096
0_0000_0100_0000	M=64		
0_0000_1000_0000	M=128		
0_0001_0000_0000	M=256	Other than the above	Setting disabled

### Bit 28 to 16 TXSN (Texture Size N)

Sets vertical texture size. Any power of 2 between 4 and 4096 can be used. Values that are not a power of 2 cannot be used.

0_0000_0000_0100	N=4	0_0010_0000_0000	N=512
0_0000_0000_1000	N=8	0_0100_0000_0000	N=1024
0_0000_0001_0000	N=16	0_1000_0000_0000	N=2048
0_0000_0010_0000	N=32	1_0000_0000_0000	N=4096
0_0000_0100_0000	N=64		
0_0000_1000_0000	N=128		
0_0001_0000_0000	N=256	Other than the above	Setting disabled

### TIS (Tile Size)

Register address	DrawBaseAddress + 468 <sub>H</sub>																																					
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
Bit field name																	TISM																					
R/W																	RW																					
Initial value																	1000000																					

This register specifies the tile size (m, n).

Bit 6 to 0 TISM (Title Size M)

Sets horizontal tile size. Any power of 2 between 4 and 64 can be used. Values that are not a power of 2 cannot be used.

0.000100 M=4

0001000 M=8

0010000 M=16

0100000 M=32

1000000 M=64

Other than the above Setting disabled

Bit 22 to 16 TISN (Title Size N)

Sets vertical tile size. Any power of 2 between 4 and 64 can be used. Values that are not a power of 2 cannot be used.

0000100 N=4

0001000 N=8

0010000 N=16

0100000 N=32

1000000 N=64

Other than the above Setting disabled

### TOA (Texture Buffer Offset address)

Register address	DrawBaseAddress + 46C <sub>H</sub>																																					
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
Bit field name																	XBO																					
R/W																	RW																					
Initial value																	Don't care																					

This register sets the texture buffer offset address. Using this offset value, texture patterns can be referred to the texture buffer memory.

Specify the word-aligned byte address (16 bits). (Bit 0 is always "0".)

### ABR (Alpha map Base)

Register address	DrawBaseAddress + 474 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	ABASE																															
R/W	RW																R0															
Initial value	Don't care																0															

This register sets the base address of the alpha map.

### FC (Foreground Color)

Register address	DrawBaseAddress + 480 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																	FGC															
R/W																	RW															
Initial value																	0															

This register sets the drawing foreground color. This color is for the object color for flat shading and foreground color for bitmap drawing and broken line drawing. All bits set to “1” are drawn in the color set at this register.

8 bit color mode:

- Bit 7 to 0 FGC8 (Foreground 8 bit Color)  
Sets the indirect color for the foreground (color index code).
- Bit 31 to 8 These bits are not used.

16 bit color mode:

- Bit 15 to 0 FGC16 (Foreground 16 bit Color)  
This field sets the 16-bit direct color for the foreground.  
Note that the handling of bit 15 is different from that in ORCHID.  
Up to ORCHID, bit 15 is “0” for other than bit map and rectangular drawing, but starting with MINT GDC, the setting value is reflected in memory as is. This bit is also reflected in bit 15 of the 16-bit color at Gouraud shading.
- Bit 31 to 16 These bits are not used.

## BC (Background Color)

Register address	DrawBaseAddress + 484 <sub>H</sub>
Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit field name	BGC8/16
R/W	RW
Initial value	0

This register sets the drawing frame background color. This color is used for the background color of bitmap drawing and broken line drawing. At bitmap drawing, all bits set to “0” are drawn in the color set at this register.

BT bit of this register allows the background color of be transparent (no drawing).

### 8 bit color mode:

- Bit 7 to 0      BGC8 (Background 8 bit Color)  
Sets the indirect color for the background (color index code)
- Bit 14 to 8      Not used
- Bit 15      BT (Background Transparency)  
Sets the transparent mode for the background color  
0      Background drawn using color set for BGC field  
1      Background not drawn (transparent)
- Bit 31 to 16      Not used

### 16 bit color mode:

- Bit 14 to 0      BGC16 (Background 16 bit Color)  
Sets 16-bit direct color (RGB) for the background
- Bit 15      BT (Background Transparency)  
Sets the transparent mode for the background color  
0      Background drawn using color set for BGC field  
1      Background not drawn (transparent)
- Bit 31 to 16      Not used



### ALF (Alpha Factor)

Register address	DrawBaseAddress + 488 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																									A							
R/W																									RW							
Initial value																									0							

This register sets the alpha blending coefficient.

### BLP (Broken Line Pattern)

Register address	DrawBaseAddress + 48C <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	BLP																															
R/W	RW																															
Initial value	0																															

This register sets the broken-line pattern. The bit 1 set in the broken-line pattern is drawn in the foreground color and bit 0 is drawn in the background color. The line pattern for 1 pixel line is laid out in the direction of MSB to LSB and when it reaches LSB, it goes back to MSB. The BLPO register manages the bit numbers of the broken-line pattern. 32 or 24 bits can be selected as the repetition of the broken-line pattern by the BP bit of the MDR1 register. When 24 bits are selected, bits 23 to 0 of the BLP register are used.

### TBC (Texture Border Color)

Register address	DrawBaseAddress + 494 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	BC8/16																															
R/W	RW																															
Initial value	0																															

This register sets the border color for texture mapping.

8 bit color mode:

Bit 7 to 0      BC8 (Border Color)  
Sets the 8-bit direct color for the texture border color

16 bit color mode:

Bit 15 to 0      BC16 (Border Color)  
Sets the 16-bit direct color for the texture border color  
Bit15 is used for controlling a stencil and stencil alpha

### BLPO (Broken Line Pattern Offset)

Register address	DrawBaseAddress + 3E0 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name																													BCR			
R/W																													RW			
Initial value																													11111			

This register stores the bit number of the broken-line pattern set to BLP registers, for broken line drawing. This value is decremented at each pixel drawing. Broken line can be drawn starting from any starting position of the specified broken-line pattern by setting any value at this register.

When no write is performed, the position of broken-line pattern is sustained.

## 11.2.7 Triangle drawing registers

Each register is used by the drawing commands. The registers cannot be accessed from the CPU or using the **SetRegister** command.

### (XY coordinates register)

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Ys	0000 <sub>H</sub>	S	S	S	S																												
Xs	0004 <sub>H</sub>	S	S	S	S																												
dXdY	0008 <sub>H</sub>	S	S	S	S																												
XUs	000C <sub>H</sub>	S	S	S	S																												
dXUdy	0010 <sub>H</sub>	S	S	S	S																												
XLs	0014 <sub>H</sub>	S	S	S	S																												
dXLdy	0018 <sub>H</sub>	S	S	S	S																												
USN	001b <sub>H</sub>	0	0	0	0																												
LSN	0020 <sub>H</sub>	0	0	0	0																												

Address    Offset value from DrawBaseAddress  
S            Sign bit or sign extension  
0            Not used or 0 extension  
Int          Integer or integer part of fixed point data  
Frac        Fraction part of fixed point data

Sets (X, Y) coordinates for triangle drawing

Ys	Y coordinates start position of long edge
Xs	X coordinates start position of long edge corresponding to Ys
dXdY	X DDA value of long edge direction
XUs	X coordinates start position of upper edge
dXUdy	X DDA value of upper edge direction
XLs	X coordinates start position of lower edge
dXLdy	X DDA value of lower edge direction
USN	Count of spans of upper triangle. If this value is "0", the upper triangle is not drawn.
LSN	Count of spans of lower triangle. If this value is "0", the lower triangle is not drawn.

### (Color setting register)

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Rs	0040 <sub>H</sub>	0	0	0	0	0	0	0	0																								
dRdx	0044 <sub>H</sub>	S	S	S	S	S	S	S	S																								
dRdy	0048 <sub>H</sub>	S	S	S	S	S	S	S	S																								
Gs	004C <sub>H</sub>	0	0	0	0	0	0	0	0																								
dGdx	0050 <sub>H</sub>	S	S	S	S	S	S	S	S																								
dGdy	0054 <sub>H</sub>	S	S	S	S	S	S	S	S																								
Bs	0058 <sub>H</sub>	0	0	0	0	0	0	0	0																								
dBdx	005C <sub>H</sub>	S	S	S	S	S	S	S	S																								
dBdy	0060 <sub>H</sub>	S	S	S	S	S	S	S	S																								
As	0064 <sub>H</sub>	0	0	0	0	0	0	0	0																								
dAdx	0068 <sub>H</sub>	S	S	S	S	S	S	S	S																								
dAdy	006C <sub>H</sub>	S	S	S	S	S	S	S	S																								

Address    Offset from DrawBaseAddress  
S            Sign bit or sign extension  
0            Not used or 0 extension  
Int          Integer or integer part of fixed point data  
Frac        Fraction part of fixed point data

Sets color parameters for triangle drawing. These parameters are enabled in the Gouraud shading mode.

Rs	R value at (Xs, Ys, Zs) of long edge corresponding to Ys
dRdx	R DDA value of horizontal direction
dRdy	R DDA value of long edge
Gs	G value at (Xs, Ys, Zs) of long edge corresponding to Ys
dGdx	G DDA value of horizontal direction
dGdy	G DDA value of long edge
Bs	B value at (Xs, Ys, Zs) of long edge corresponding to Ys
dBdx	B DDA value of horizontal direction
dBdy	B DDA value of long edge
As	Alpha value at (Xs, Ys, Zs) of long edge corresponding to Ys
dAdx	Alpha DDA value of horizontal direction
dAdy	Alpha DDA value of long edge

### (Z coordinates register)

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Zs	0080 <sub>h</sub>	0																															
dZdx	0084 <sub>h</sub>	S																															
dZdy	008C <sub>h</sub>	S																															

Address    Offset from DrawBaseAddress  
S            Sign bit or sign extension  
0            Not used or 0 extension  
Int          Integer or integer part of fixed point data  
Frac        Fraction part of fixed point data

Sets Z coordinates for 3D triangle drawing

Zs	Z coordinate start position of long edge
dZdx	Z DDA value of horizontal direction
dZdy	Z DDA value of long edge

**(Texture coordinates-setting register)**

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Ss	00c0 <sub>H</sub>	S	S	S																													
dSdx	00c4 <sub>H</sub>	S	S	S																													
dSdy	00c8 <sub>H</sub>	S	S	S																													
Ts	00cc <sub>H</sub>	S	S	S																													
dTdx	00d0 <sub>H</sub>	S	S	S																													
dTdy	00d4 <sub>H</sub>	S	S	S																													
Qs	00d8 <sub>H</sub>	0	0	0	0	0	0	0	0	Int																							
dQdx	00dc <sub>H</sub>	S	S	S	S	S	S	S	S	Int																							
dQdy	00e0 <sub>H</sub>	S	S	S	S	S	S	S	S	Int																							

Address    Offset from DrawBaseAddress  
S            Sign bit or sign extension  
0            Not used or 0 extension  
Int          Integer or integer part of fixed point data  
Frac        Fraction part of fixed point data

Sets texture coordinates parameters for triangle drawing

Ss	S texture coordinates (Xs, Ys, Zs) of long edge corresponding to Ys
dSdx	S DDA value of horizontal direction
dSdy	S DDA value of long edge direction
Ts	T texture coordinates (Xs, Ys, Zs) of long edge corresponding to Ys
dTdx	T DDA value of horizontal direction
dTdy	T DDA value of long edge direction
Qs	Q (Perspective correction value) of texture at (Xs, Ys, Zs) of long edge corresponding to Ys
dQdx	Q DDA value of horizontal direction
dQdy	Q DDA value of long edge direction

## 11.2.8 Line drawing registers

Each register is used by the drawing commands. The registers cannot be accessed from the CPU or by using the **SetRegister** command.

### (Coordinates setting register)

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPN	0140 <sub>H</sub>	0	0	0	0	Int												0															
LXs	0144 <sub>H</sub>	S	S	S	S	Int												Frac															
LXde	0148 <sub>H</sub>	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	Int	Frac															
LYs	014C <sub>H</sub>	S	S	S	S	Int												Frac															
LYde	0150 <sub>H</sub>	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	Int	Frac															
LZs	0154 <sub>H</sub>	S	Int												Frac																		
LZde	0158 <sub>H</sub>	S	Int												Frac																		

Address    Offset from DrawBaseAddress  
S           Sign bit or sign extension  
0           Not used or 0 extension  
Int        Integer or integer part of fixed point data  
Frac       Fraction part of fixed point data

Sets coordinates parameters for line drawing

LPN	Pixel count of principal axis direction
LXs	X coordinates start position of draw line (In principal axis X) Integer value of X coordinates rounded off (In principal axis Y) X coordinates in form of fixed point data
LXde	Inclination data for X coordinates (In principal axis X) Increment or decrement according to drawing direction (In principal axis Y) Fraction part of DX/DY
LYs	Y coordinates start position of draw line (In principal axis X) Y coordinates in form of fixed point data (In principal axis Y) Integer value of Y coordinates rounded off
LYde	Inclination data for Y coordinates (In principal axis X) Fraction part of DY/DX (In principal axis Y) Increment or decrement according to drawing direction
LZs	Z coordinates start position of line drawing line
LZde	Z Inclination

## 11.2.9 Pixel drawing registers

Each register is used by the drawing commands. The registers cannot be accessed from the CPU or using the **SetRegister** command.

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PXdc	0180 <sub>H</sub>	0	0	0	0	Int										0																	
PYdc	0184 <sub>H</sub>	0	0	0	0	Int										0																	
PZdc	0188 <sub>H</sub>	0	0	0	0	Int										0																	

Address    Offset from DrawBaseAddress  
S           Sign bit or sign extension  
0           Not used or 0 extension  
Int        Integer or integer part of fixed point data  
Frac       Fraction part of fixed point data

Sets coordinates parameter for drawing pixel. The foreground color is used.

PXdc	Sets X coordinates position
PYdc	Sets Y coordinates position
PZdc	Sets Z coordinates position

## 11.2.10 Rectangle drawing registers

Each register is used by the drawing commands. The registers cannot be accessed from the CPU or using the **SetRegister** command.

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXs	0200 <sub>H</sub>	0	0	0	0	Int										0																	
Rys	0204 <sub>H</sub>	0	0	0	0	Int										0																	
RsizeX	0208 <sub>H</sub>	0	0	0	0	Int										0																	
RsizeY	020C <sub>H</sub>	0	0	0	0	Int										0																	

Address    Offset from DrawBaseAddress  
S           Sign bit or sign extension  
0           Not used or 0 extension  
Int        Integer or integer part of fixed point data  
Frac       Fraction part of fixed point data

Sets coordinates parameters for rectangle drawing. The foreground color is used.

RXs	Sets the X coordinates of top left vertex
Rys	Sets the Y coordinates of top left vertex
RsizeX	Sets horizontal size
RsizeY	Sets vertical size

## 11.2.11 Blt registers

Sets the parameters of each register as described below:

Set the Tcolor register with the **SetRegister** command.

Note that the Tcolor register cannot be set at access from the CPU and by drawing commands.

Each register except the Tcolor register is set by executing a drawing command.

Note that access from the CPU and the **SetRegister** command cannot be used.

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
SADDR	0240 <sub>H</sub>	0	0	0	0	0	0	0	Address																											
SStride	0244 <sub>H</sub>	0	0	0	0	Int										0																				
SRXs	0248 <sub>H</sub>	0	0	0	0	Int										0																				
SRYs	024C <sub>H</sub>	0	0	0	0	Int										0																				
DADDR	0250 <sub>H</sub>	0	0	0	0	0	0	0	Address																											
DStride	0254 <sub>H</sub>	0	0	0	0	Int										0																				
DRXs	0258 <sub>H</sub>	0	0	0	0	Int										0																				
DRYs	025C <sub>H</sub>	0	0	0	0	Int										0																				
BRsizeX	0260 <sub>H</sub>	0	0	0	0	Int										0																				
BRsizeY	0264 <sub>H</sub>	0	0	0	0	Int										0																				
TColor	0280 <sub>H</sub>	0																	Color																	

Address Offset from DrawBaseAddress

S Sign bit or sign extension

0 Not used or 0 extension

Int Integer or integer part of fixed point data

Frac Fraction part of fixed point data

Sets parameters for Blt operations

SADDR	Sets start address of source rectangle area in byte address
SStride	Sets stride of source
SRXs	Sets X coordinates start position of source rectangle area
SRYs	Sets Y coordinates start position of source rectangle area
DADDR	Sets start address of destination rectangle area in byte address
DStride	Sets stride of destination
DRXs	Sets X coordinates start position of destination rectangle area
DRYs	Sets Y coordinates start position of destination rectangle area
BRsizeX	Sets horizontal size of rectangle
BRsizeY	Sets vertical size of rectangle
Tcolor	Sets transparent color For indirect color, set a palette code in the lower 8 bits.

## 11.2.12 2D line with XY setup drawing registers

Each register is used by the drawing commands. The registers cannot be accessed from the CPU.

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LX0dc	0540 <sub>H</sub>	0	0	0	0	Int										0																	
LY0dc	0544 <sub>H</sub>	0	0	0	0	Int										0																	
LX1dc	0548 <sub>H</sub>	0	0	0	0	Int										0																	
LY1dc	054C <sub>H</sub>	0	0	0	0	Int										0																	

Address    Offset from DrawBaseAddress  
S            Sign bit or sign extension  
0            Not used or 0 extension  
Int          Integer or integer part of fixed point data  
Frac        Fraction part of fixed point data

Sets coordinates of line end points for 2D Line with XY setup drawing

LX0dc	Sets X coordinates of vertex V0
LY0dc	Sets Y coordinates of vertex V0
LX1dc	Sets X coordinates of vertex V1
LY1dc	Sets Y coordinates of vertex V1



### 11.2.13 2D triangle with XY setup drawing registers

Each register is used by the drawing commands. The registers cannot be accessed from the CPU or using the **SetRegister** command.

Register	Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X0dc	0580h	0	0	0	0	Int														0													
Y0dc	0584h	0	0	0	0	Int														0													
X1dc	0588h	0	0	0	0	Int														0													
Y1dc	058ch	0	0	0	0	Int														0													
X2dc	0590h	0	0	0	0	Int														0													
Y2dc	0594h	0	0	0	0	Int														0													

Address    Offset from DrawBaseAddress

S           Sign bit or sign extension

0           Not used or 0 extension

Int          Integer or integer part of fixed point data

Frac        Fraction part of fixed point data

Sets coordinates of three vertices for 2D Triangle with XY setup drawing

X0dc	Sets X coordinates of vertex V0
Y0dc	Sets Y coordinates of vertex V0
X1dc	Sets X coordinates of vertex V1
Y1dc	Sets Y coordinates of vertex V1
X2dc	Sets X coordinates of vertex V2
Y2dc	Sets Y coordinates of vertex V2

11.2.14 Display list FIFO registers

DFIFO (Displaylist FIFO)

Register address	DrawBaseAddress+ 4A0 <sub>H</sub>																															
Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field name	DFIFO																															
R/W	W																															
Initial value	Don't care																															

FIFO registers for Display List transfer

## 11.2.15 I2C registers

### BSR (Bus Status Register)

Register address	I2C Base Address + 000h							
Bit No	7	6	5	4	3	2	1	0
Bit field name	BB	RSC	AL	LRB	TRX	AAS	GCA	FBT
R/W	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

All bits on this register are cleared while bit EN on CCR register is “0”.

Bit7	<b>BB (Bus Busy)</b> Indicate state of I2C-bus 0: STOP condition was detected. 1: START condition (The bus is in use.) was detected.
Bit6	<b>RSC (Repeated START Condition)</b> Indicate repeated START condition This bit is cleared by writing “0” to INT bit, the case of not addressed in a slave mode, the detection of START condition under bus stop, and the detection of STOP condition. 0: Repeated START condition was not detected. 1: START condition was detected again while the bus was in use.
Bit5	<b>AL(Arbitration Lost)</b> Detect Arbitration lost This bit is cleared by writing “0” to INT bit. 0: Arbitration lost was not detected. 1: Arbitration occurred during master transmission, or “1” writing was performed to MSS bit while other systems were using the bus.
Bit4	<b>LRB (Last Received Bit)</b> Store Acknowledge This bit is cleared by detection of START condition or STOP condition.
Bit3	<b>TRX (Transmit / Receive)</b> Indicate data receipt and data transmission. 0: receipt 1: transmission
Bit2	<b>AAS (Address As Slave)</b> Detect addressing This bit is cleared by detection of START condition or STOP condition. 0: Addressing was not performed in a slave mode. 1: Addressing was performed in a slave mode.
Bit1	<b>GCA (General Call Address)</b> Detect general call address (00h) This bit is cleared by detection of START condition or STOP condition. 0: General call address was not received in a slave mode. 1: General call address was received in a slave mode.
Bit0	<b>FBT (First Byte Transfer)</b> Detect the 1st byte Even if this bit is set to “1” by detection of START condition, it is cleared by writing “0” on INT bit or by not being addressed in a slave mode. 0: Received data is not the 1st byte. 1: Received data is the 1st byte (address data).

## BCR (Bus Control Register)

Register address	I2C Base Address + 0004h							
Bit No	7	6	5	4	3	2	1	0
Bit field name	BER	BEIE	SCC	MSS	ACK	GCAA	INTE	INT
R/W	R/W0	R/W	R0/W1	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

- Bit7**      **BER (Bus Error)**  
Flag bit for request of bus error interruption  
When this bit is set, EN bit on CCR register will be cleared, this module will be in a stop state and data transfer will be discontinued.  
write case  
0: A request of buss error interruption is cleared.  
1: Don't care.  
read case  
0: A bus error was not detected.  
1: Undefined START condition or STOP condition was detected while data transfer.
- Bit6**      **BEIE (Bus Error Interruption Enable)**  
Permit bus error interruption  
When both this bit and BER bit are "1", the interruption is generated.  
0: Prohibition of bus error interruption  
1: Permission of bus error interruption
- Bit5**      **SCC (Start Condition Continue)**  
Generate START condition  
write case  
0: Don't care.  
1: START condition is generated again at the time of master transmission.
- Bit4**      **MSS (Master Slave Select)**  
Select master / slave mode  
When arbitration lost is generated in master transmission, this bit is cleared and this module becomes a slave mode.  
0: This module becomes a slave mode after generating STOP condition and completing transfer.  
1: This module becomes a master mode, generates START condition and starts transfer.
- Bit3**      **ACK (ACKnowledge)**  
Permit generation of acknowledge at the time of data reception  
This bit becomes invalid at the time of address data reception in a slave mode.  
0: Acknowledge is not generated.  
1: Acknowledge is generated.
- Bit2**      **GCAA(General Call Address Acknowledge)**  
Permit generation of acknowledge at the time of general call address reception  
0: Acknowledge is not generated.  
1: Acknowledge is generated.
- Bit1**      **INTE (INTerrupt Enable)**  
Permit interruption  
When this bit is "1" interruption is generated if INT bit is "1".  
0: Prohibition of interrupt  
1: Permission of interrupt
- Bit0**      **INT (INTerrupt)**  
Flag bit for request of interruption for transfer end  
When this bit is "1" SCL line is maintained at "L" level. If this bit is cleared by being written "0", SCL line is released and the following byte transfer is started. Moreover, it is reset to "0" by generating of START condition or STOP condition at the time of a master.  
write case  
0: The flag is cleared.  
1: Don't care.

read case

0: The transfer is not ended.

1: It is set when 1 byte transfer including the acknowledge bit is completed and it corresponds to the following conditions.

- It is a bus master.
- It is an addressed slave.
- It was going to generate START condition while other systems by which arbitration lost happened used the bus.

### **Competition of SCC, MSS and INT bit**

Competition of the following byte transfer, generation of START condition and generation of STOP condition happens by the simultaneous writing of SCC, MSS and INT bit. The priority at this case is as follows.

1) The following byte transfer and generation of STOP condition

If “0” is written to INT bit and “0” is written to MSS bit, priority will be given to “0” writing to MSS bit and STOP condition will be generated.

2) The following byte transfer and generation of START condition

If “0” is written to INT bit and “1” is written to SCC bit, priority will be given to “1” writing to SCC bit and START condition will be generated.

3) Generation of START condition and STOP condition

The simultaneous writing of “1” to SCC bit and “0” to MSS bit is prohibition.

## CCR (Clock Control Register)

Register address	I2C Base Address + 0008h							
Bit No	7	6	5	4	3	2	1	0
Bit field name	-	HSM	EN	CS4	CS3	CS2	CS1	CS0
R/W	R1	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	1	0	0	-	-	-	-	-

- Bit7 Nonuse  
"1" is always read at read.
- Bit6 HSM (High Speed Mode)  
Select standard-mode / high-speed-mode  
0: Standard-mode  
1: High-speed-mode
- Bit5 EN (Enable)  
Permission of operation  
When this bit is "0", each bit of BSR and BCR register (except BER and BEIE bit) is cleared. This bit is cleared when BER bit is set.  
0: Prohibition of operation  
1: Permission of operation
- Bit4 CS4 - 0 (Clock Period Select4 - 0)  
  
Set up the frequency of a serial transfer clock  
Frequency fsc1 of a serial transfer clock is shown as the following formula.  
Please set up fsc1 not to exceed the value shown below at the time of master operation.  
standard-mode: 100KHz  
high-speed-mode: 400KHz

### standard-mode

$$fsc1 = \frac{A}{(2 \times m) + 2}$$

### high-speed-mode

$$fsc1 = \frac{A}{\text{int}(1.5 \times m) + 2}$$

A: I2C system clock = 1/24 of PLL output => About 16.6MHz

### <Notes>

+2 cycles are minimum overhead to confirm that the output level of SCL terminal changed. When the delay of the positive edge of SCL terminal is large or when the clock is extended by the slave device, it becomes larger than this value.

The value of m becomes like the following page to the value of CS 4-0.

CS4	CS3	CS2	CS1	CS0	m	
					standard	high-speed
0	0	0	0	0	65	inhibited
0	0	0	0	1	66	inhibited
0	0	0	1	0	67	inhibited
0	0	0	1	1	68	inhibited
0	0	1	0	0	69	inhibited
0	0	1	0	1	70	inhibited
0	0	1	1	0	71	inhibited
0	0	1	1	1	72	inhibited
0	1	0	0	0	73	9
0	1	0	0	1	74	10
0	1	0	1	0	75	11
0	1	0	1	1	76	12
0	1	1	0	0	77	13
0	1	1	0	1	78	14
0	1	1	1	0	79	15
0	1	1	1	1	80	16
1	0	0	0	0	81	17
1	0	0	0	1	82	18
1	0	0	1	0	83	19
1	0	0	1	1	84	20
1	0	1	0	0	85	21
1	0	1	0	1	86	22
1	0	1	1	0	87	23
1	0	1	1	1	88	24
1	1	0	0	0	89	25
1	1	0	0	1	90	26
1	1	0	1	0	91	27
1	1	0	1	1	92	28
1	1	1	0	0	93	29
1	1	1	0	1	94	30
1	1	1	1	0	95	31
1	1	1	1	1	96	32

## Address Register(ADR)

Register address	I2C Base Address + 000Ch							
Bit No	7	6	5	4	3	2	1	0
Bit field name	-	A6	A5	A4	A3	A2	A1	A0
R/W	R1	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	1	-	-	-	-	-	-	-

- Bit7            Nonuse  
                 “1” is always read at read.
- Bit6 - 0        A6 - 0 (Address6 - 0)  
                 Store slave address  
                 In a slave mode it is compared with DAR register after address data reception,  
                 and when in agreement, acknowledge is transmitted to a master.

## Data Register(DAR)

Register address	I2C Base Address + 0010h							
Bit No	7	6	5	4	3	2	1	0
Bit field name	D7	D6	D5	D4	D3	D2	D1	D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	-	-	-	-	-	-	-	-

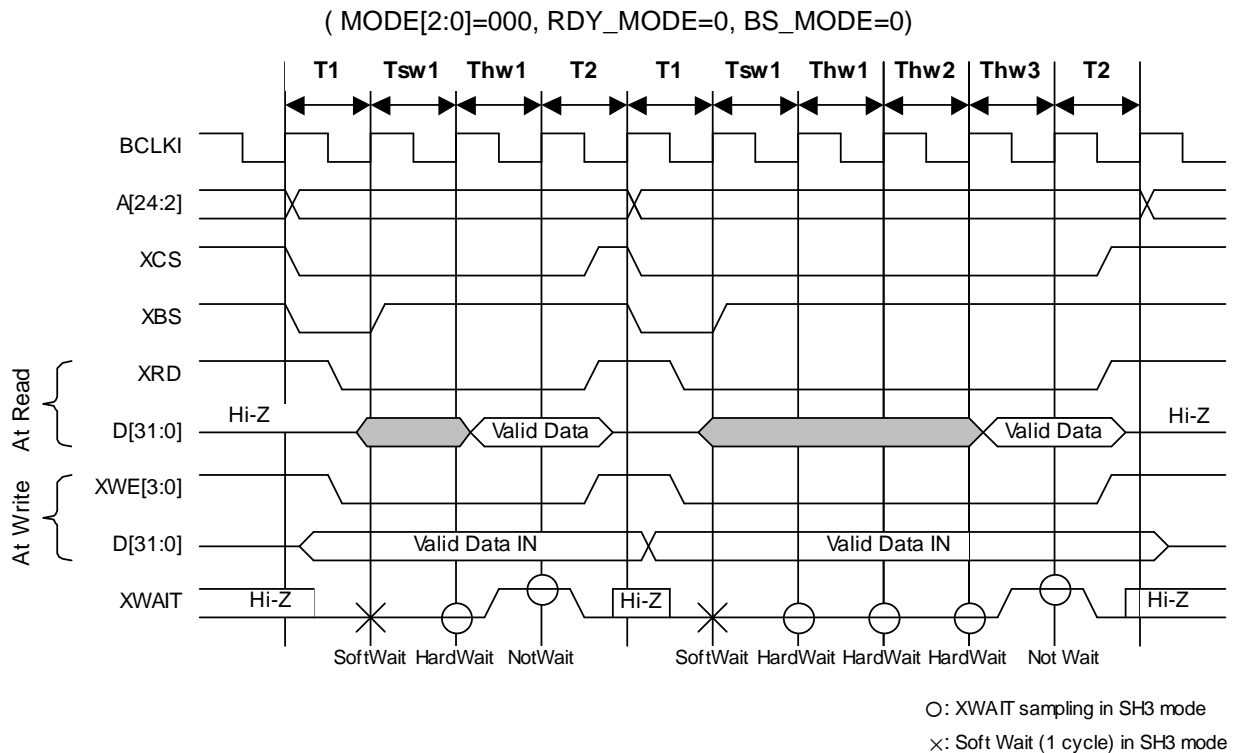
- Bit7 - 0        D7 - 0 (Data7 - 0)  
                 Store serial data  
                 This is a data register for serial data transfer. The data is transferred from  
                 MSB. At the time of data reception (TRX=0) the data output is set to “1”.  
                 The writing side of this register is a double buffer. When the bus is in use  
                 (BB=1), the write data is loaded to the register for serial transfer for every  
                 transfer. At the time of read-out, the receiving data is effective only when INT  
                 bit is set because the register for serial transfer is read directly at this time.



## 12 TIMING DIAGRAM

### 12.1 Host Interface

#### 12.1.1 CPU read/write timing diagram in SH3 mode (Normally Not Ready Mode)



T1: Read/write start cycle (XRDY in wait state)

Tsw\*: Software wait insertion cycle (1 cycle setting)

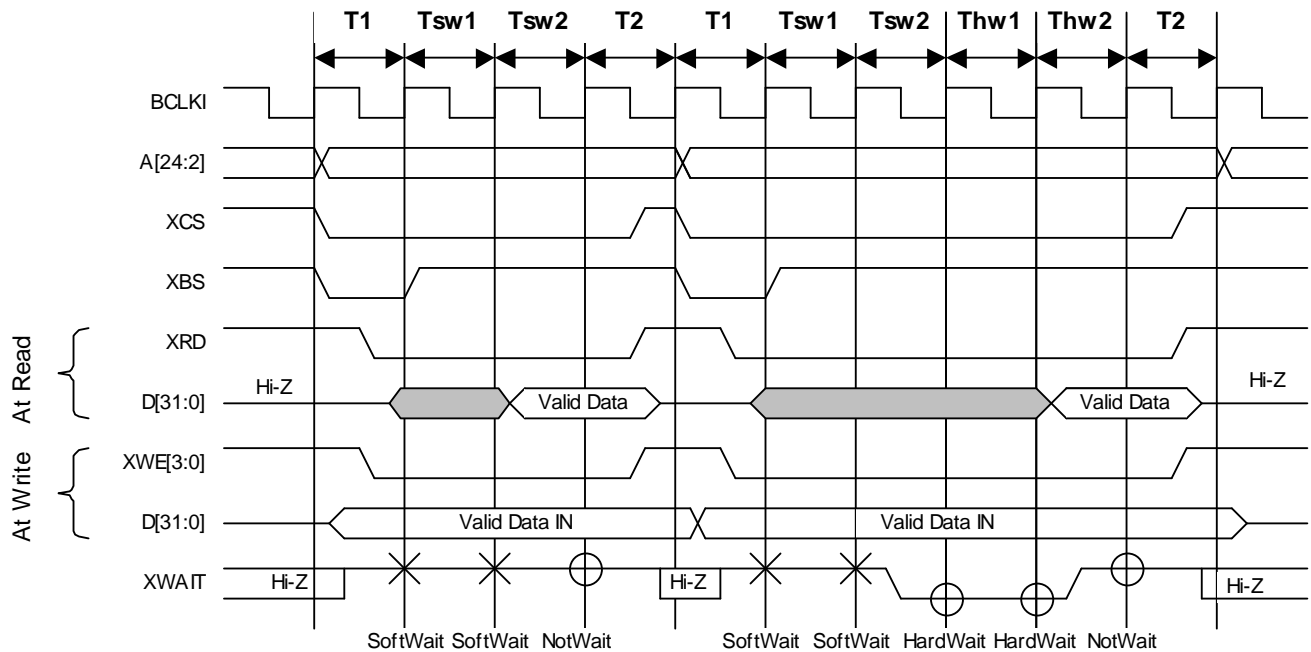
Thw\*: Hardware wait insertion cycle (XRDY cancels the wait state after the preparations)

T2: Read/write end cycle (XRDY ends in wait state)

**Fig. 10.1 Read/Write Timing Diagram for SH3 (Normally Not Ready Mode)**

## 12.1.2 CPU read/write timing diagram in SH3 mode (Normally Ready Mode)

(MODE[2:0]=000, RDY\_MODE=1, BS\_MODE=0)



O: XWAIT sampling in SH3 mode  
x: Soft Wait (2 cycles) in SH3 mode

T1: Read/write start cycle (XRDY in not wait state)

Tsw\*: Software wait insertion cycle (2-cycle setting required)

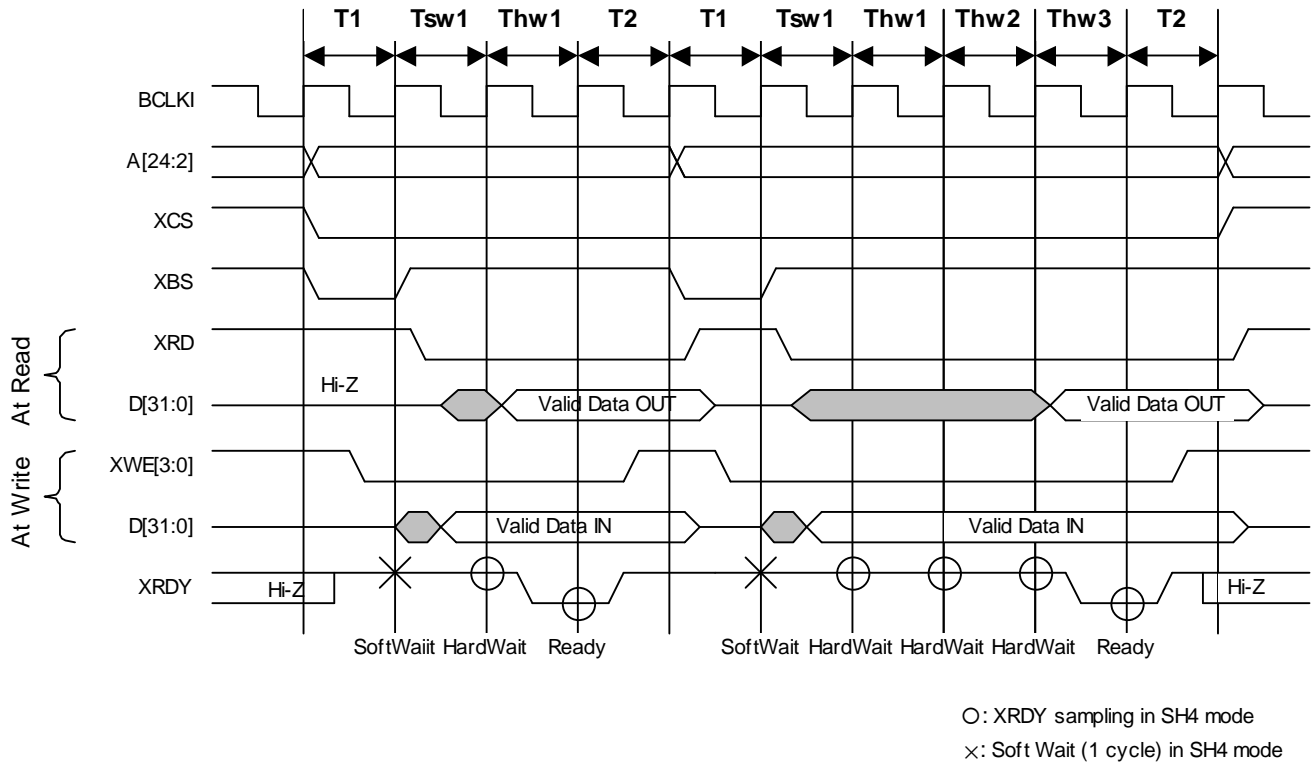
Thw\*: Hardware wait insertion cycle (In hardware state when the immediate accessing is disabled)

T2: Read/write end cycle (XRDY ends in not wait state)

**Fig. 10.2 Read/Write Timing Diagram for SH3 (Normally Ready Mode)**

### 12.1.3 CPU read/write timing diagram in SH4 mode (Normally Not Ready Mode)

(MODE[2:0]=001, RDY\_MODE=0, BS\_MODE=0)



T1: Read/write start cycle (XRDY in the not ready state)

Tsw\*: Software wait insertion cycle (1 cycle)

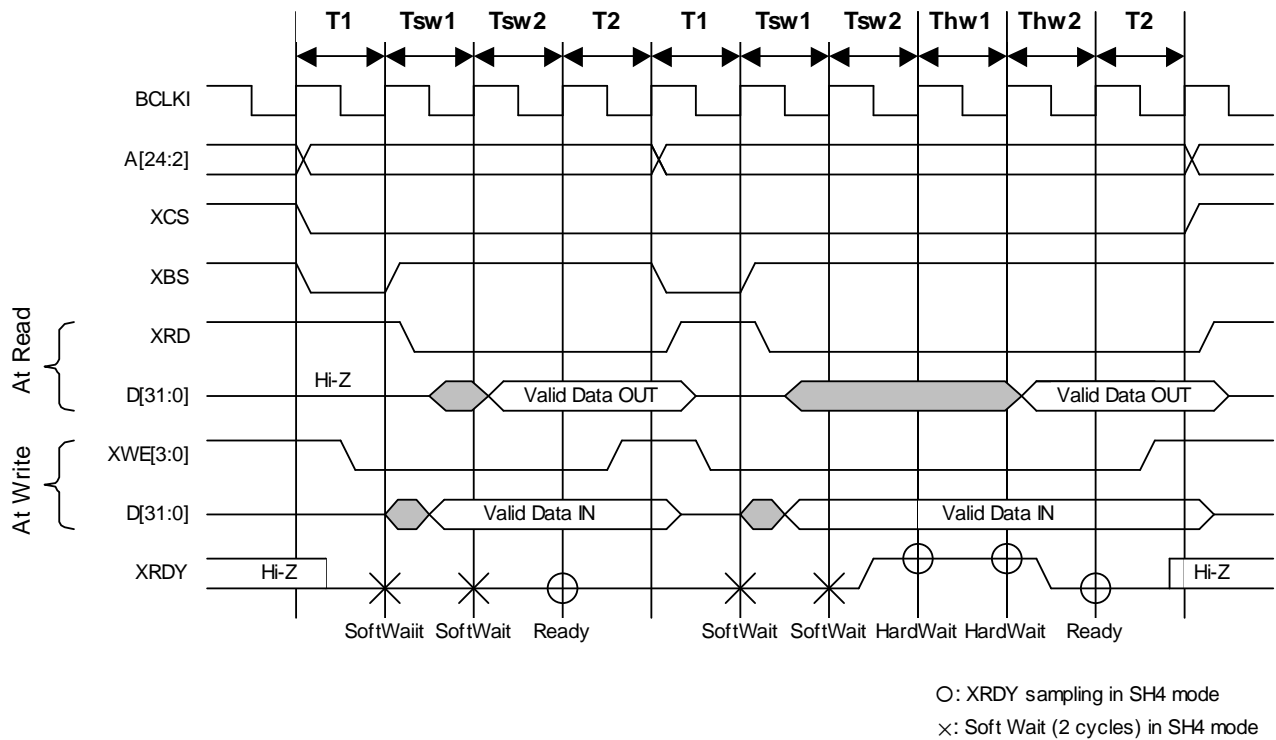
Twh\*: Hardware wait insertion cycle (XRDY asserts Ready after the preparations)

T2: Read/write end cycle (XRDY ends in not ready state)

**Fig. 10.3 Read/Write Timing Diagram for SH4 Mode (Normally Not Ready Mode)**

### 12.1.4 CPU read/write timing diagram in SH4 mode (Normally Ready Mode)

(MODE[2:0]=001, RDY\_MODE=1, BS\_MODE=0)

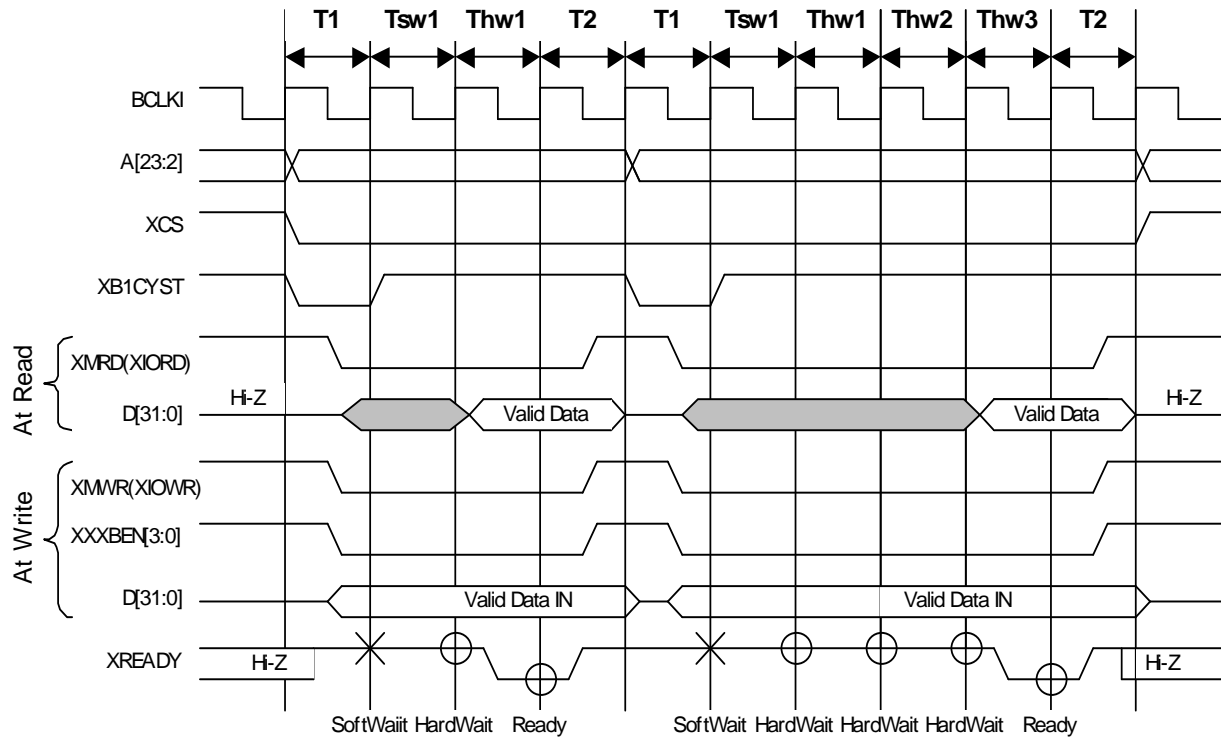


- T1: Read/write start cycle (XRDY in ready state)  
Tsw\*: Software wait insertion cycle (2-cycle setting required)  
Twh\*: Hardware wait insertion cycle (XRDY asserts Ready after the preparations)  
T2: Read/write end cycle (XRDY ends in ready state.)

**Fig. 10.4 CPU Read/Write Timing Diagram for SH4 Mode (Normally Ready Mode)**

### 12.1.5 CPU read/write timing diagram in V832 mode (Normally Not Ready Mode)

(MODE[2:0]=010, RDY\_MODE=0, BS\_MODE=0)



○: XREADY sampling in V832 mode

×: Soft Wait (1 cycle) in V832 mode

T1: Read/write start cycle (XREADY in not ready state)

Tsw\*: Software wait insertion cycle

Thw\*: Hardware wait insertion cycle (XREADY asserts Ready after the preparations)

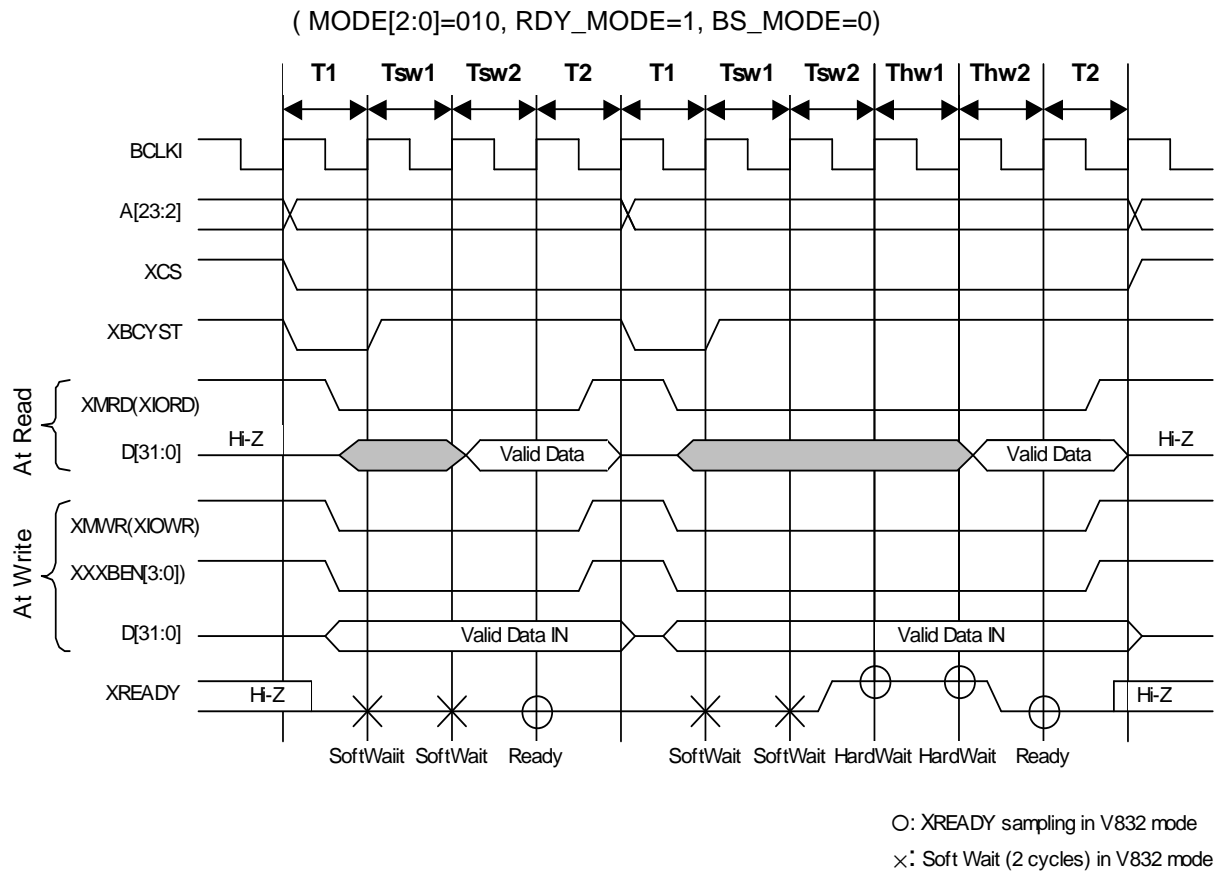
T2: Read/write end cycle (XREADY ends in not ready state)

Notes: 1.The XxxBEN signal is used only for a write from the CPU; it is not used for a read from the CPU.

2.The CPU always inserts one cycle wait after read access.

**Fig. 10.5 Read/Write Timing Diagram in V832 Mode (Normally Not Ready Mode)**

## 12.1.6 CPU read/write timing diagram in V832 mode (Normally Ready Mode)



T1: Read/write start cycle (XREADY in ready state)

Tsw\*: Software wait insertion cycle (2-cycle setting required)

Twh\*: Hardware wait insertion cycle (XREADY asserts Ready after the preparations)

T2: Read/write end cycle (XREADY ends in ready state)

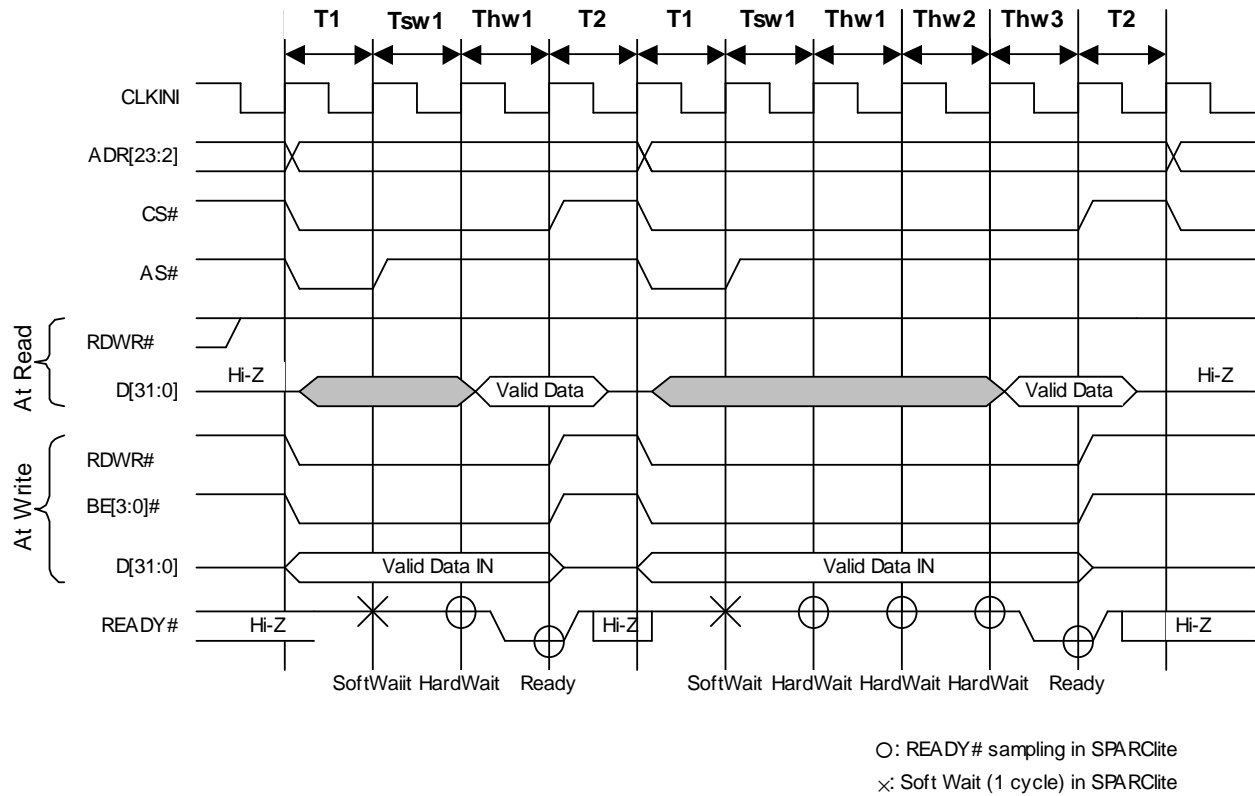
Notes: 1.The XxxBEN signal is used only for a write from the CPU; it is not used for a read from the CPU.

2.The CPU always inserts one cycle wait after read access.

**Fig. 10.6 Read/Write Timing Diagram in V832 Mode (Normally Ready Mode)**

## 12.1.7 CPU read/write timing diagram in SPARClite (Normally Not Ready Mode)

(MODE[2:0]=011, RDY\_MODE=0, BS\_MODE=0)



T1: Read/write start cycle (READY# in not ready state)

Tsw\*: Software wait insertion cycle

Thw\*: Hardware wait insertion cycle (READY# asserts Ready after the preparations)

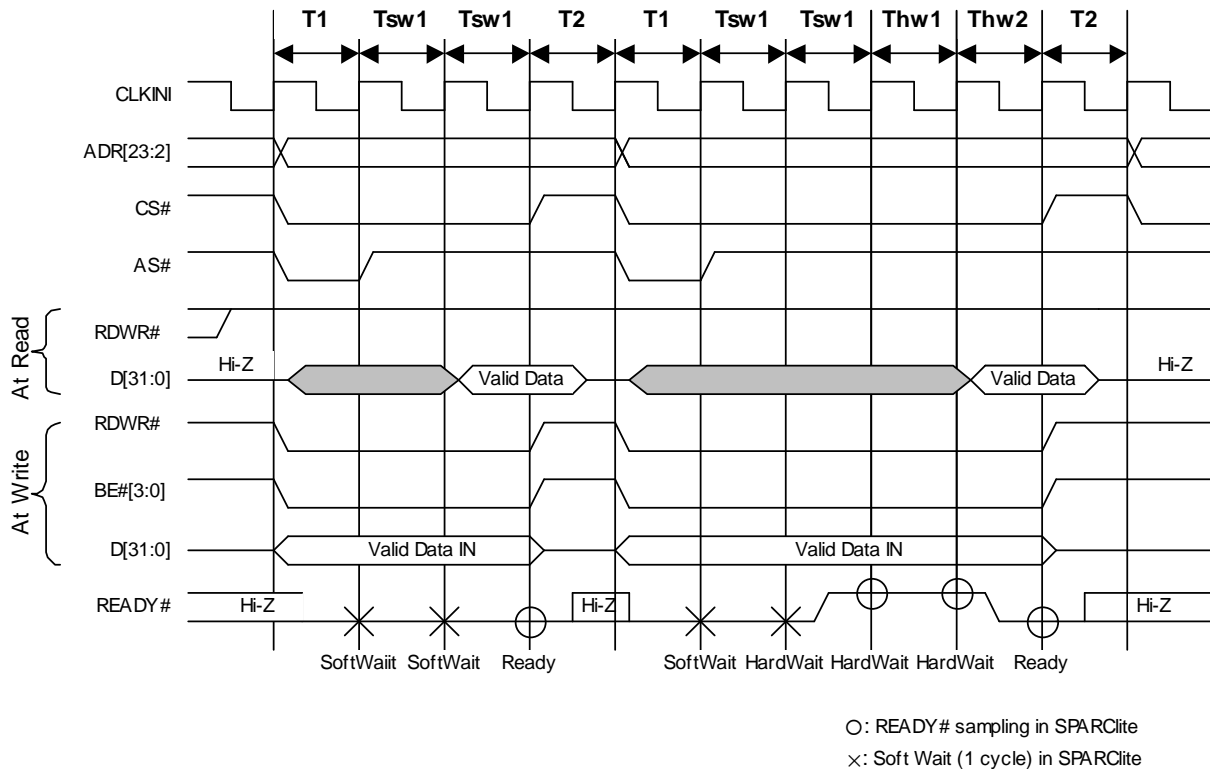
T2: Read/write end cycle (READY# ends in not ready state)

Note: BE# signal is used only for a write from the CPU; it is not used for a read from the CPU.

**Fig. 10.7 Read/Write Timing Diagram in SPARClite (Normally Not Ready Mode)**

## 12.1.8 CPU read/write timing diagram in SPARClite (Normally Ready Mode)

(MODE[2:0]=011, RDY\_MODE=1, BS\_MODE=0)



T1: Read/write start cycle (READY# in ready state)

Tsw\*: Software wait insertion cycle (2-cycle setting required)

Thw\*: Hardware wait insertion cycle (READY# asserts Ready after the preparations)

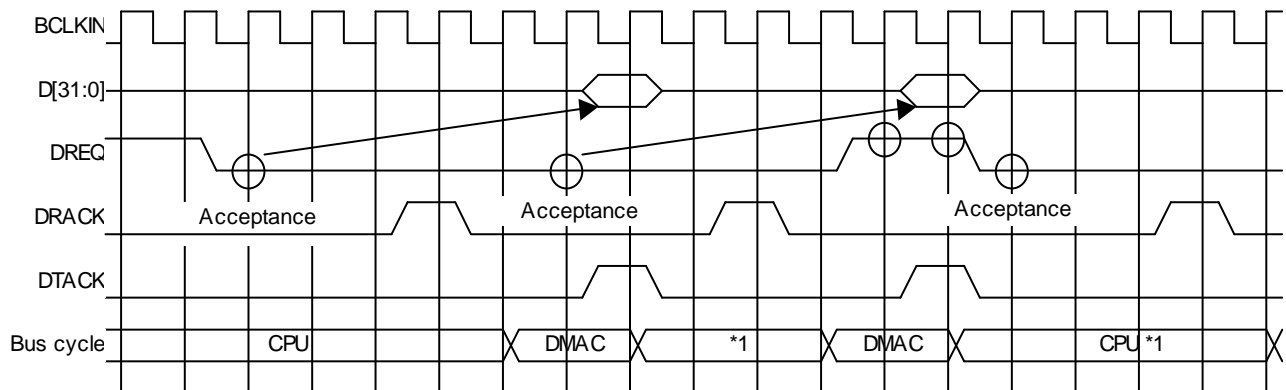
T2: Read/write end cycle (READY# ends in ready state)

Note: BE# signal is used only for a write from the CPU; it is not used for a read from the CPU.

**Fig. 10.8 Read/Write Timing Diagram in SPARClite (Normally Ready Mode)**



### 12.1.9 SH4 single-address DMA write (transfer of 1 long word)



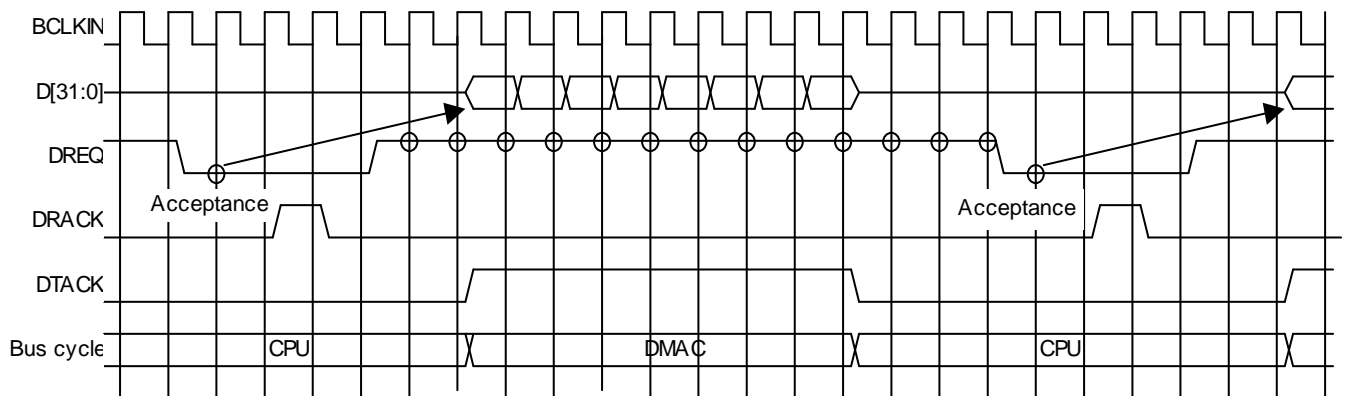
○: DREQ sampling and channel priority determination for SH mode (DREQ = level detection)

\*1: In the cycle steal mode, even when DREQ is already asserted at the 2nd DREQ sampling, the right to use the bus is returned to the CPU temporarily. In the burst mode, DMAC secures the right to use the bus unless DREQ is negated.

**Fig. 10.9 SH4 Single-address DMA Write (Transfer of 1 Long Word)**

MINT writes data according to the DTACK assert timing. When data cannot be received, the DREQ signal is automatically negated. And then the DREQ signal is reasserted as soon as data reception is ready.

### 12.1.10 SH4 single-address DMA write (transfer of 8 long words)

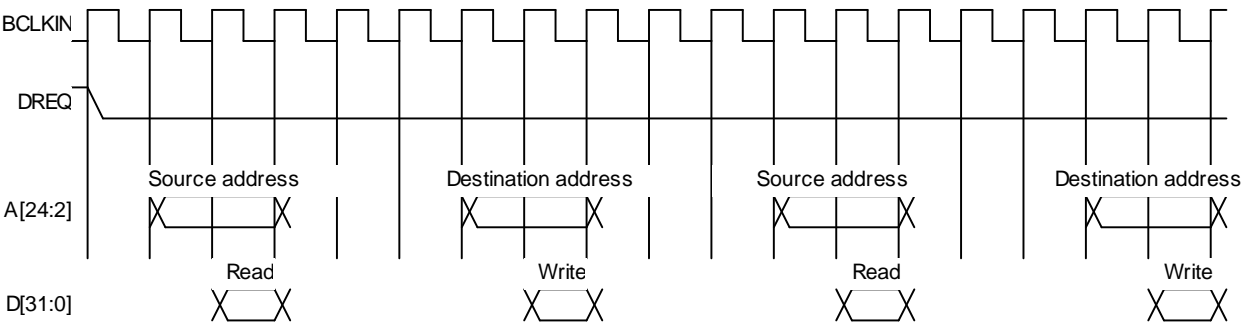


○: DREQ sampling and channel priority determination for SH mode (DREQ = level detection)

**Fig. 10.10 SH4 Single-address DMA Write (Transfer of 8 Long Words)**

After the CPU has asserted DRACK, MINT negates DREQ and receives 32-byte data in line with the DTACK assertion timing. As soon as the next data is ready to be received, MINT reasserts DREQ but the reassertion timing depends on the internal status.

12.1.11 SH3/4 dual-address DMA (transfer of 1 long word)

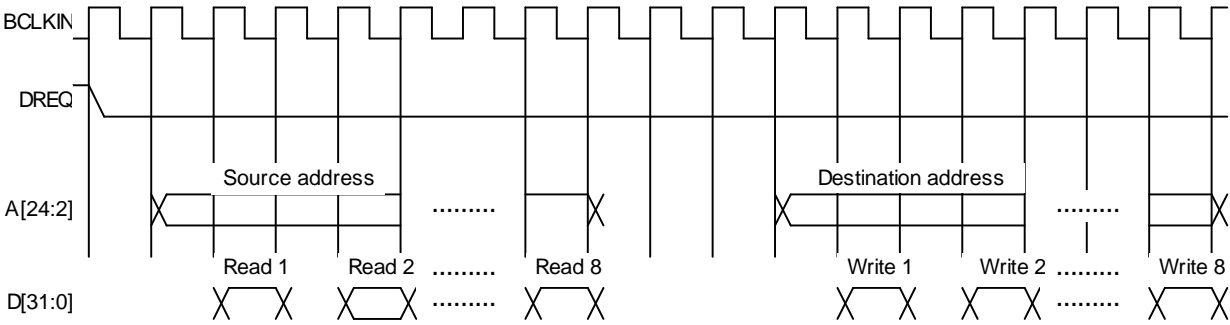


For the MINT, the read/write operation is performed according to the SRAM protocol.

Fig. 10.11 SH3/4 Dual-address DMA (Transfer of 1 Long Word)

In the dual-address mode, the DREQ signal is kept asserted until the transfer ends by default. Consequently, when MINT cannot return the ready signal immediately, in order to negate the DREQ signal set the DBM register.

12.1.12 SH3/4 dual-address DMA (transfer of 8 long words)

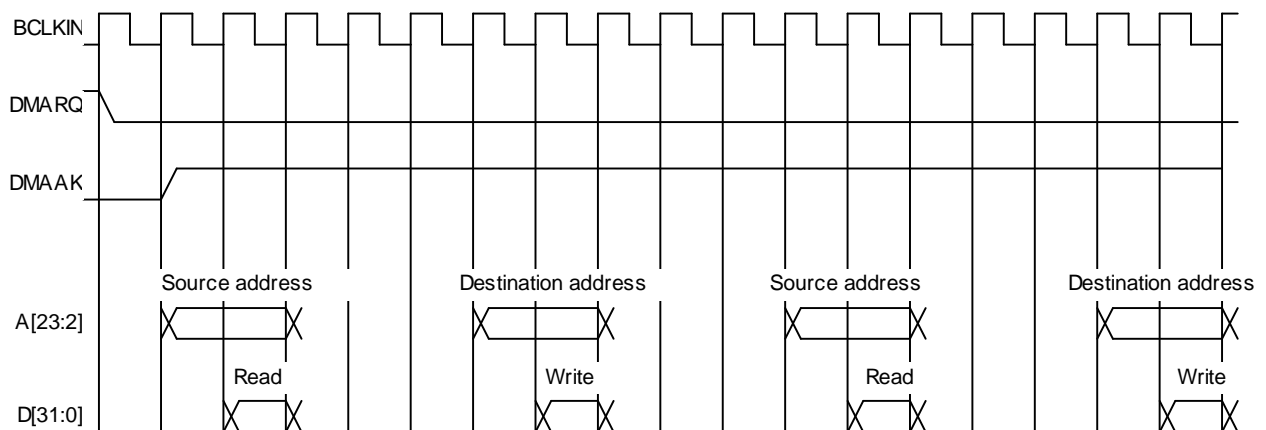


For the MINT, the read/write operation is performed according to the SRAM protocol.

Fig. 10.12 SH3/4 Dual-address DMA (Transfer of 8 Long Words)

In the dual-address mode, the DREQ signal is kept asserted until the transfer ends by default. Consequently, when MINT cannot return the ready signal immediately, in order to negate the DREQ signal set the DBM register.

### 12.1.13 V832 DMA transfer

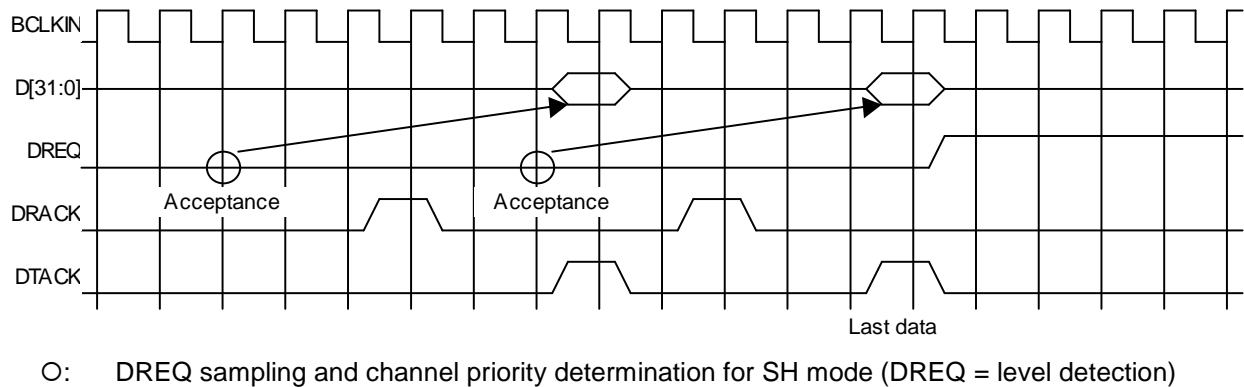


For the MINT, the read/write operation is performed according to the SRAM protocol.

**Fig. 10.13 V832 DMA Transfer**

In the dual-address mode, the DREQ signal is kept asserted until the transfer ends by default. Consequently, when MINT cannot return the ready signal immediately, in order to negate the DREQ signal set the DBM register.

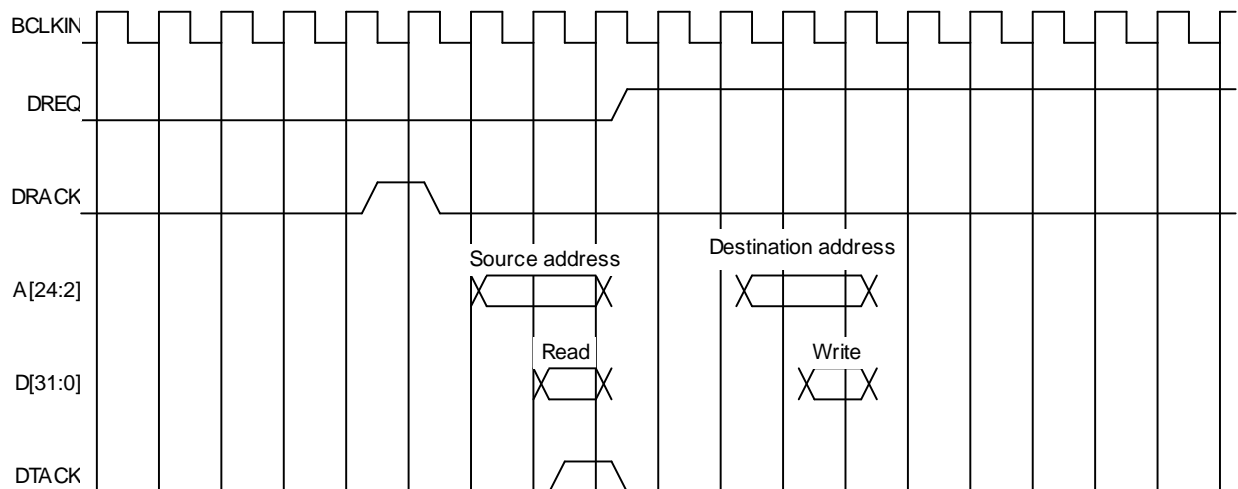
### 12.1.14 SH4 single-address DMA transfer end timing



**Fig. 10.14 SH4 Single-address DMA Transfer End Timing**

DREQ is negated three cycles after DRACK is written as the last data.

### 12.1.15 SH3/4 dual-address DMA transfer end timing



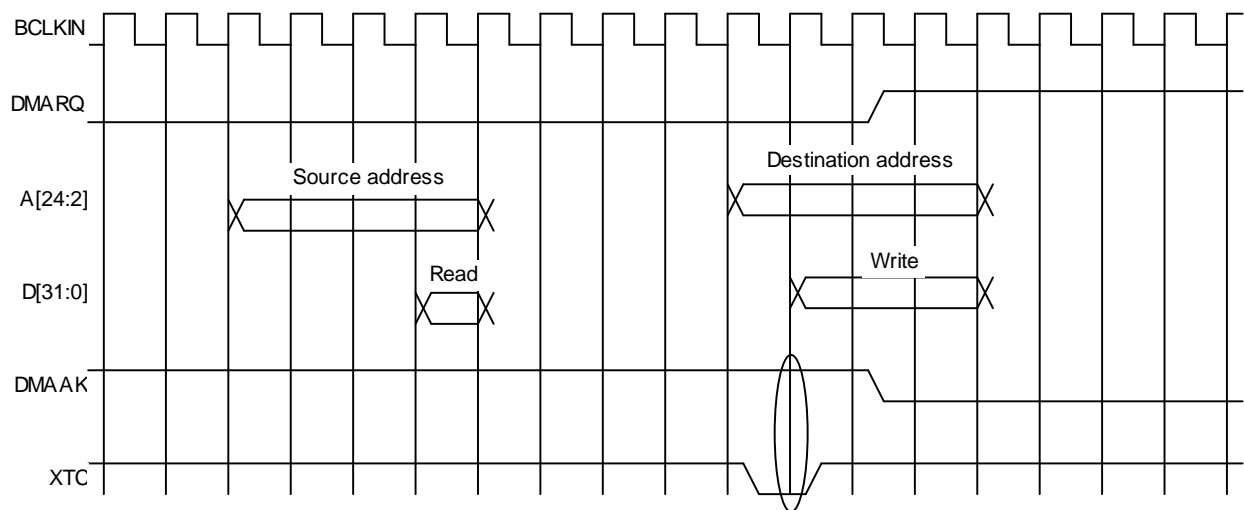
For the MINT, the read/write operation is performed according to the SRAM protocol.

**Fig. 10.15 SH3/4 Dual-address DMA Transfer End Timing**

DREQ is negated three cycles after DRACK is written as the last data.

Note: When the dual address mode (DMA) is used, the DTACK signal is not used.

### 12.1.16 V832 DMA transfer end timing

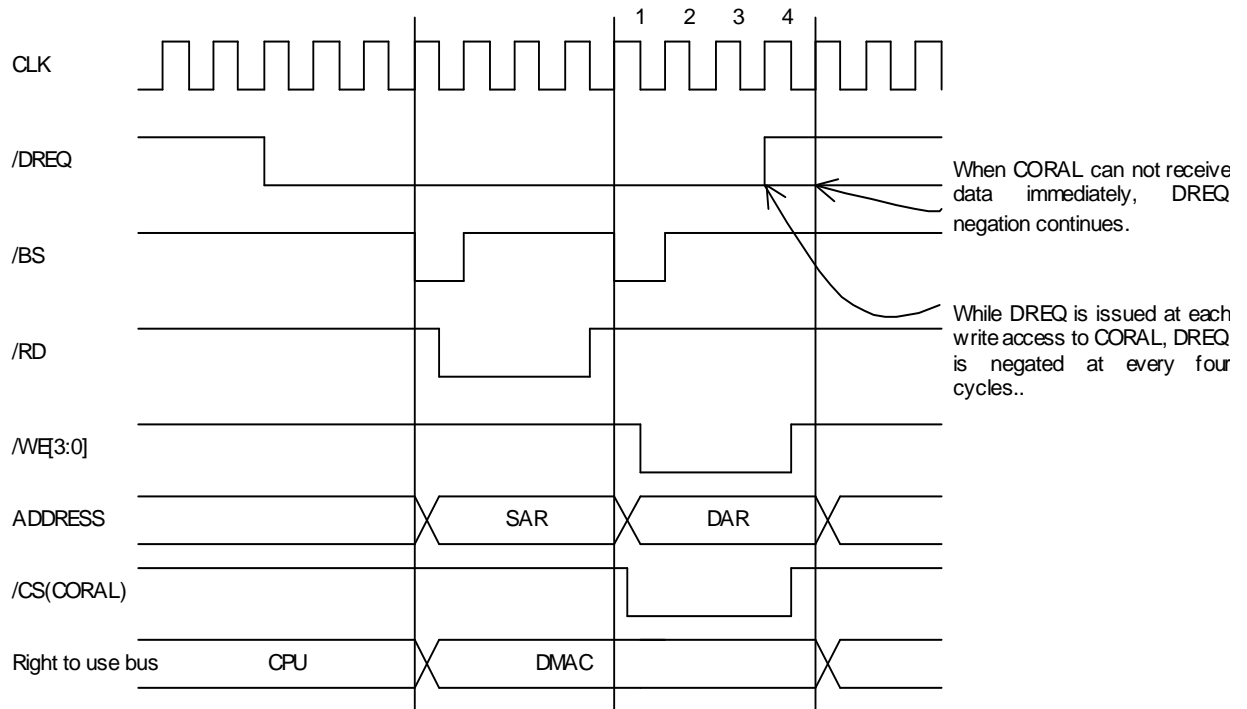


For the MINT, the read/write operation is performed according to the SRAM protocol.

**Fig. 10.16 V832 DMA Transfer End Timing**

DMAAK and XTC are logic ANDed inside MINT to end DMA.

### 12.1.17 SH4 dual DMA write without ACK



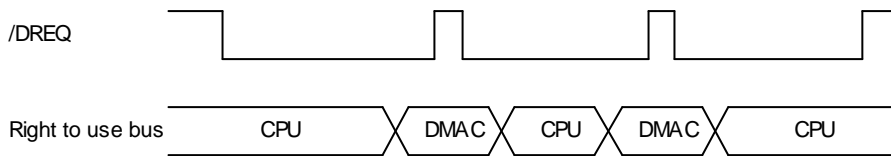
**Fig. 10.17 DREQ Negate Timing for Each Transfer**

At each DMA transfer, DREQ is negated and then reasserted at the next cycle.

Only the FIFO address can be used as the destination address.

When MINT cannot receive data immediately, DREQ negation continues. At that time, the negate timing is not only above diagram.

### 12.1.18 Dual-address DMA (without ACK) end timing



**Fig. 10.18 Dual-address DMA (without ACK) End Timing**

Example: DMA operation when DMA transfer performed twice

- (1) The CPU accesses the DREQ issue register (DRQ) of MINT to issue DREQ.
- (2) The right to use bus is transferred from the CPU to the DMAC.
- (3) In the first DMAC cycle, write is performed to MINT and DREQ is negated; DREQ is reasserted in the next cycle.
- (4) The right to use bus is returned to the CPU and the DREQ edge is detected, so the right to use bus is transferred to the DMAC.
- (5) The second write operation is performed and DREQ is negated, but DREQ is reasserted because MINT does not recognize that the transfer has ended.
- (6) The right to use bus is transferred to the CPU, so the CPU writes to the DTS register of MINT to negate DREQ.



## 12.2 Graphics Memory Interface

The MINT access timing and graphics memory access timing are explained here.

### 12.2.1 Timing of read access to same row address

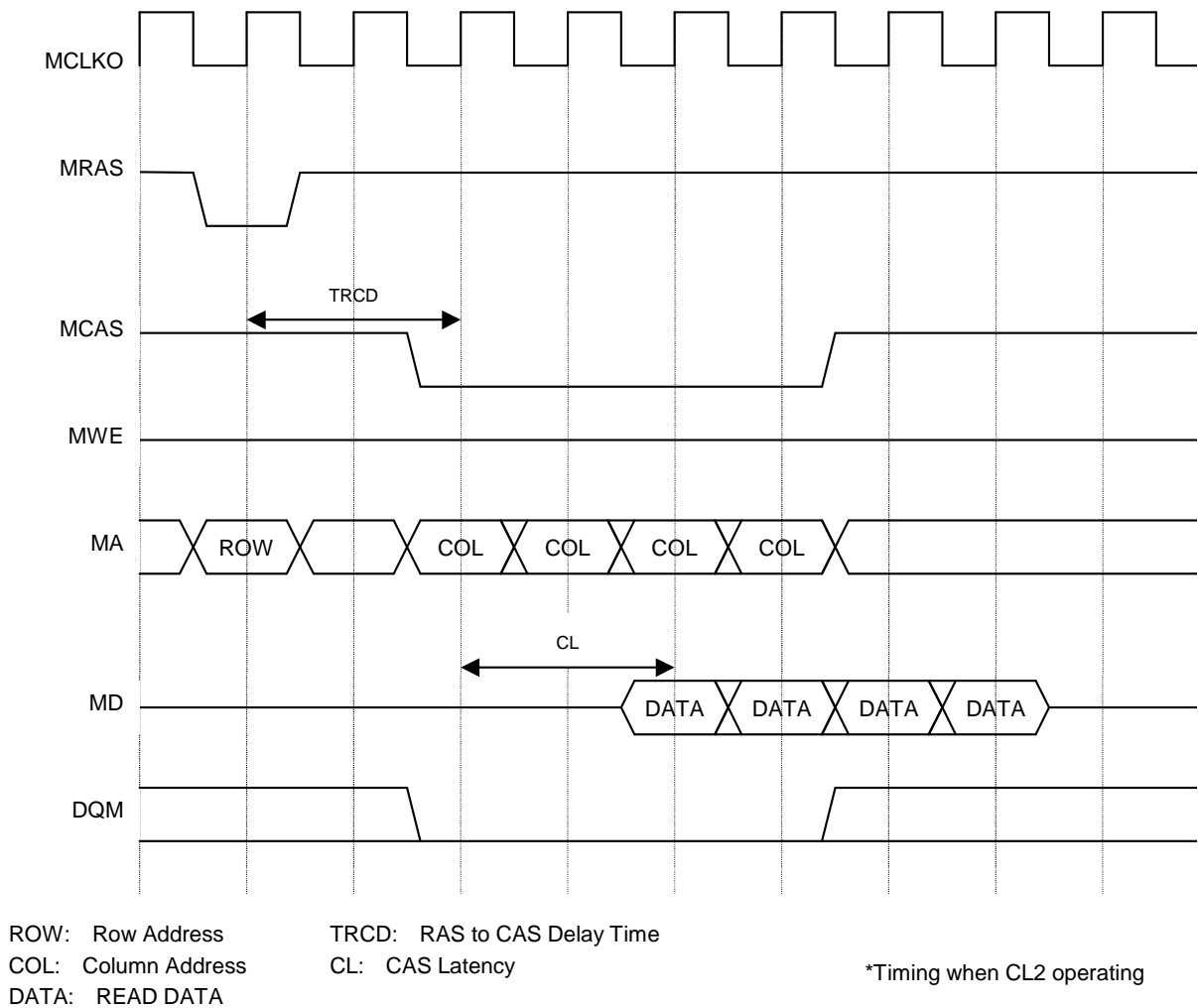
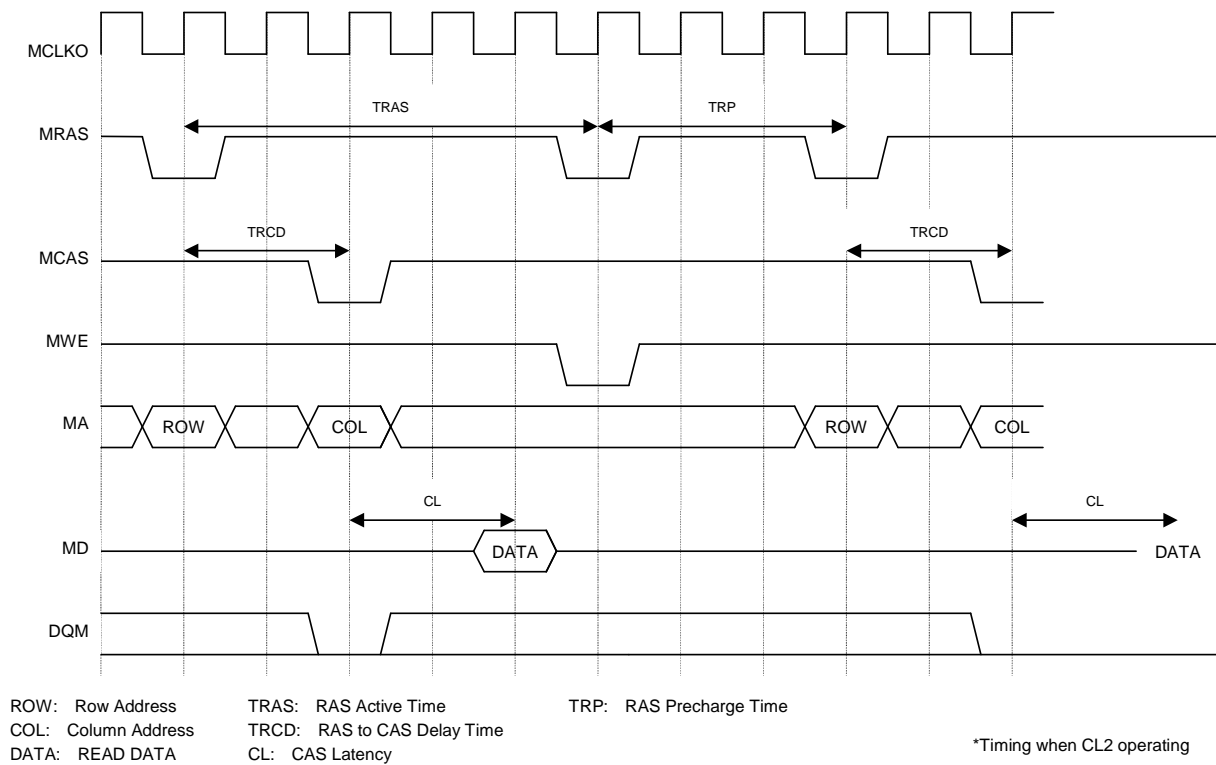


Fig. 10.19 Timing of Read Access to Same Row Address

The above timing diagram shows that read access is made four times from MINT to the same row address of SDRAM. The **ACTV** command is issued and then the **READ** command is issued after TRCD elapses. Then data that is output after the elapse of CL after the **READ** command is issued is captured into MINT.

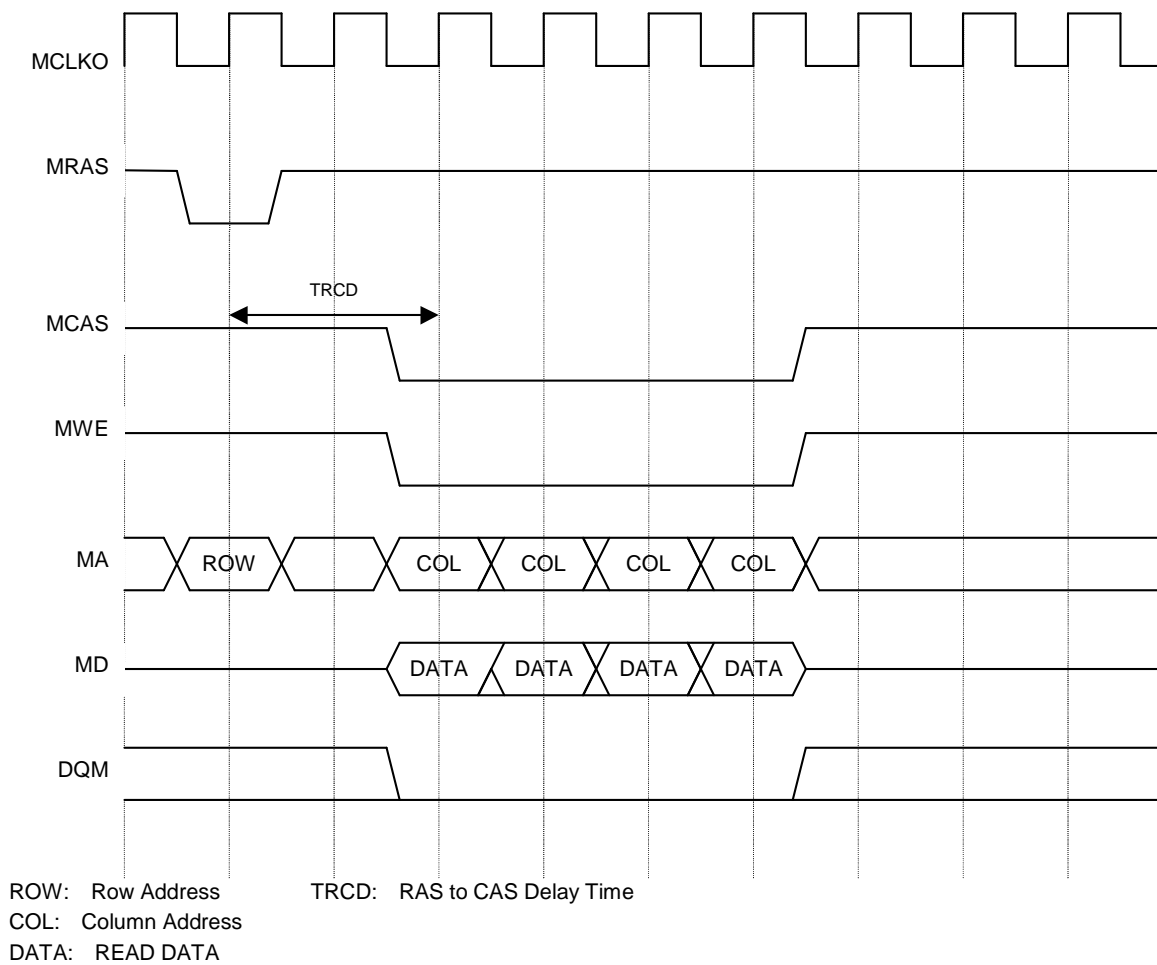
## 12.2.2 Timing of read access to different row addresses



**Fig. 10.20 Timing of Read Access to Different Row Addresses**

The above timing diagram shows that read access is made from MINT to different row addresses of SDRAM. The first and next address to be read fall across an SDRAM page boundary, so the **Pre-charge** command is issued at the timing satisfying TRAS, and then after the elapse of TRP, the **ACTV** command is reissued, and then the **READ** command is issued.

### 12.2.3 Timing of write access to same row address

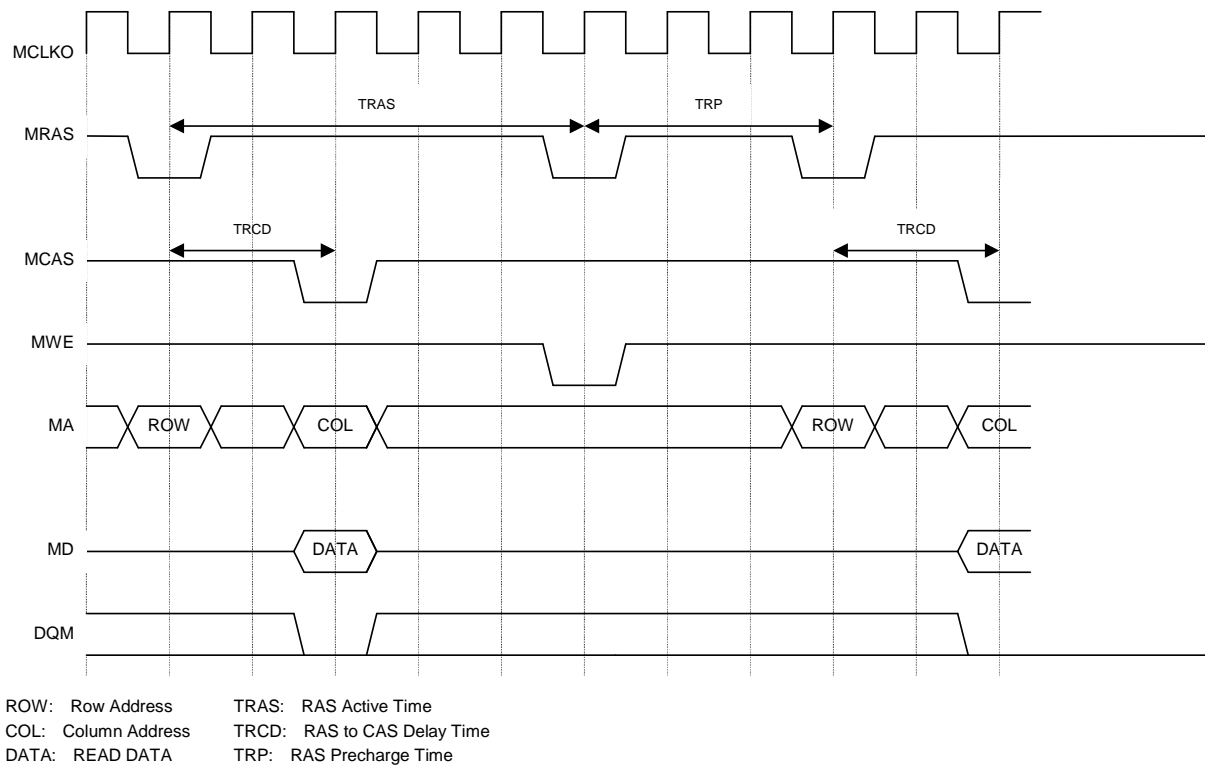


**Fig. 10.21 Timing of Write Access to Same Row Address**

The above timing diagram shows that write access is made from times from MINT to the same row address of SDRAM.

The **ACTV** command is issued, and then after the elapse of TRCD, the **WRITE** command is issued to write to SDRAM.

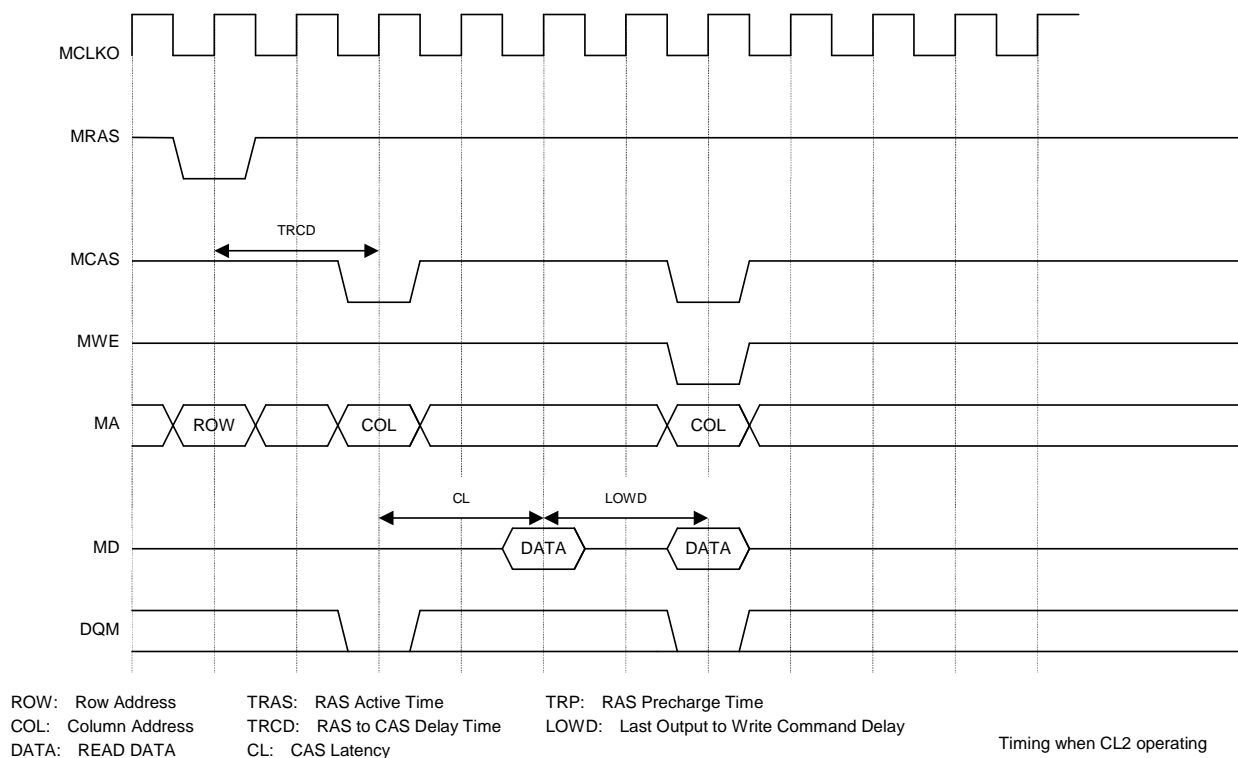
## 12.2.4 Timing of write access to different row addresses



**Fig. 10.22 Timing of Write Access to Different Row Addresses**

The above timing diagram shows that write access is made from MINT to different row addresses of SDRAM. The first and next address to be write fall across an SDRAM page boundary, so the **Pre-charge** command is issued at the timing satisfying TRAS, and then after the elapse of TRP, the **ACTV** command is reissued, and then the **WRITE** command is issued.

## 12.2.5 Timing of read/write access to same row address

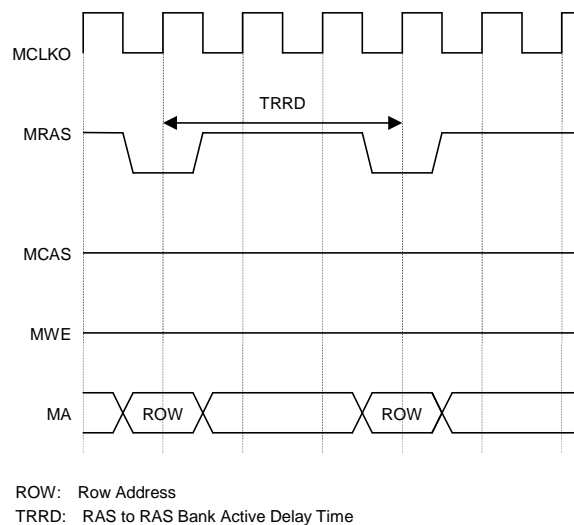


**Fig. 10.23 Timing of Read/Write Access to Same Row Address**

The above timing diagram shows that write access is made immediately after read access is made from MINT to the same row address of SDRAM.

Read data is output from SDRAM, LOWD elapses, and then the **WRITE** command is issued.

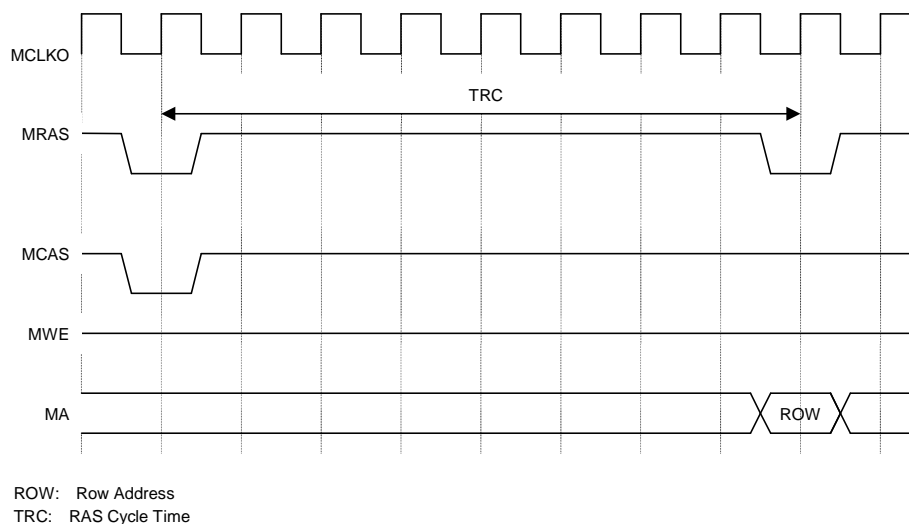
## 12.2.6 Delay between ACTV commands



**Fig.10.24 Delay between ACTV Commands**

The ACTV command is issued from MINT to the row address of SDRAM after the elapse of **TRRD** after issuance of the previous **ACTV** command.

## 12.2.7 Delay between Refresh command and next ACTV command

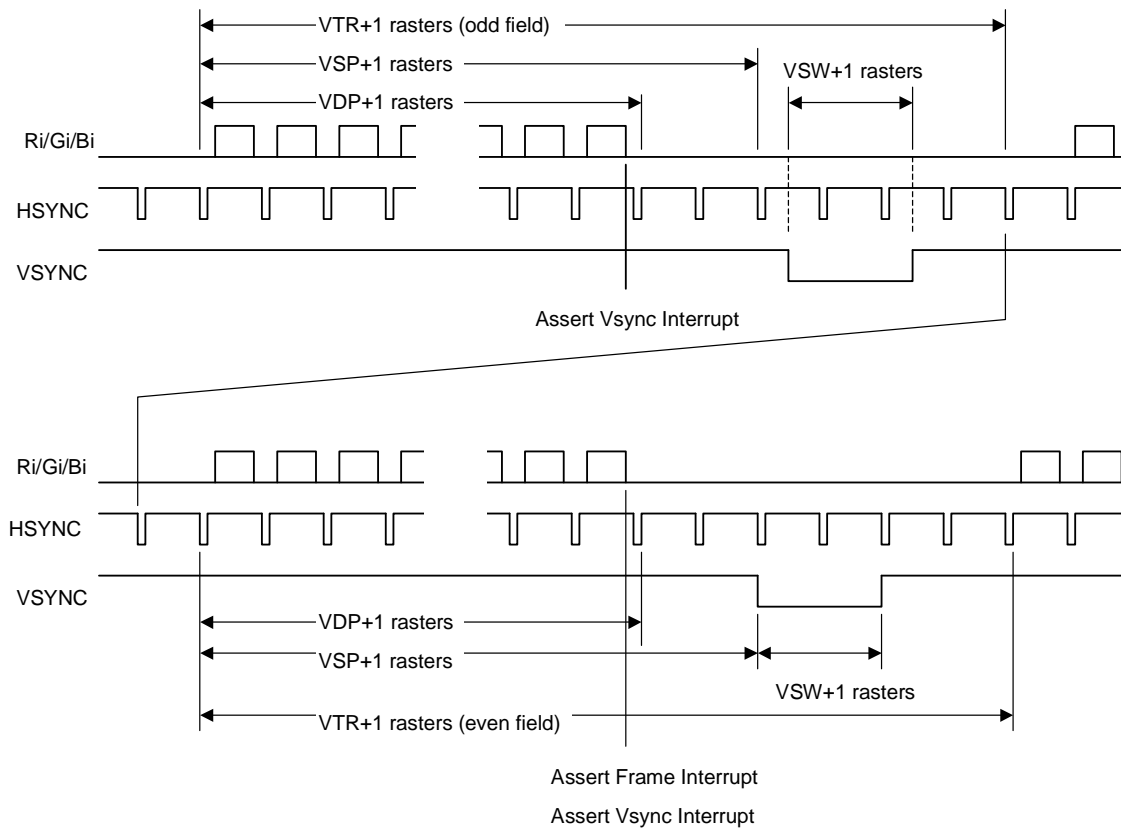


**Fig. 10.25 Delay between Refresh Command and Next ACTV Command**

The **ACTV** command is issued after the elapse of TRC after issuance of the **Refresh** command.



### 12.3.2 Interlace video mode



**Fig. 10.27 Interlace Video Timing**

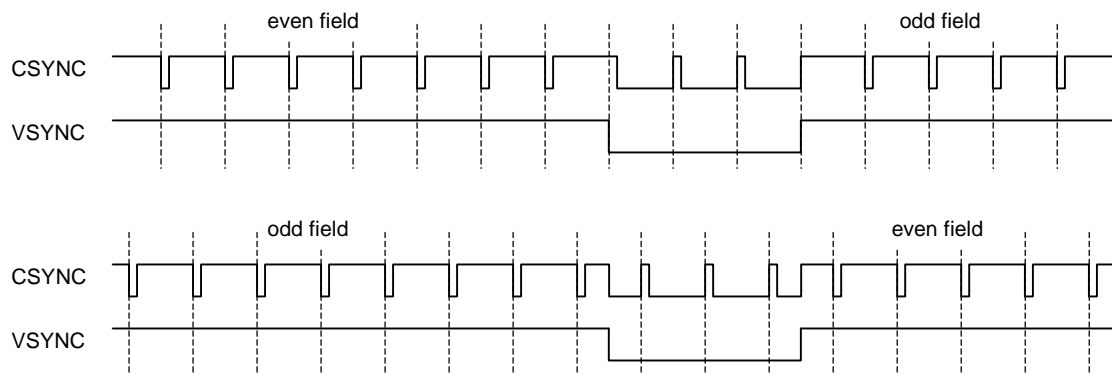
In the above diagram, VTR, HDP, etc., are the setting values of their associated registers.

The interlace mode also operates at the same timing as the interlace video mode. The only difference between the two modes is the output image data.



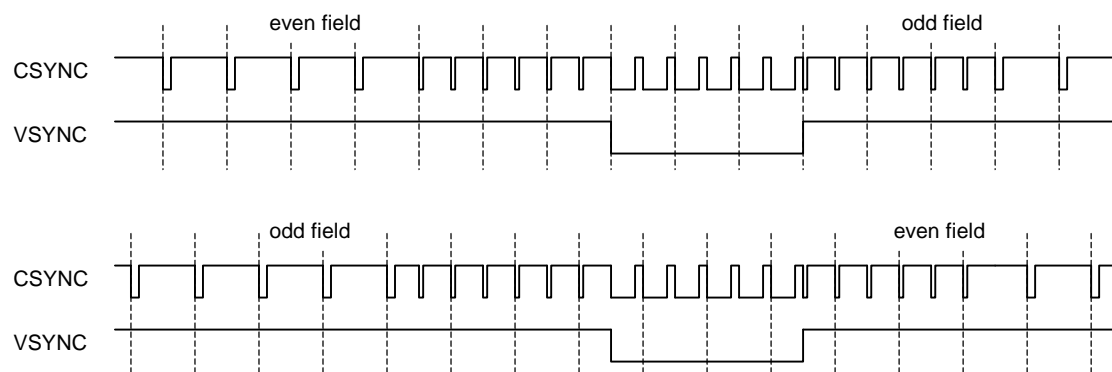
### 12.3.3 Composite synchronous signal

When the EEQ bit of the DCM register is “0”, the CSYNC signal output waveform is as shown below.



**Fig 10.28 Composite Synchronous Signal without Equalizing Pulse**

When the EEQ bit of the DCM register is “1”, the equalizing pulse is inserted into the CSYNC signal, producing the waveform shown below.



**Fig 10.29 Composite Synchronous Signal with Equalizing Pulse**

The equalizing pulse is inserted when the vertical blanking time period starts. It is also inserted three times after the vertical synchronization time period has elapsed.

## CAUTIONS

### 12.4 CPU Cautions

- 1) Enable the hardware wait for the areas to which MINT is connected. When the normally not ready mode (RDY\_MODE = 0) is used, set the software wait count to "1". When the normally ready mode (RDY\_MODE = 1) is used, set the count to "2". When the normally ready mode is used (RDY\_MODE = 1) and BS\_MODE = L, set the software wait to 2. When the normally ready mode is enabled and **BS\_MODE = H**, set the software wait to "3".
- 2) When starting DMA by issuing an external request, do so after setting the transfer count register (DTCR) and mode setting register (DSUR) of MINT to the same value as the CPU setting. In the dual DMA without ACK mode or V832 mode, there is no need to set DTCR.
- 3) When MINT is read-/write-accessed from the CPU during DMA transfer, do not access the registers and memories related to DMA transfer. If these registers and memories are accessed, reading and writing of the correct value is not assured.
- 4) Set DREQ (DMARQ) to "Low" level detection.
- 5) Set the DACK/DRACK of SH to high active output, DMAAK of V832 to high active, and XTC of V832 to low active.

## 12.5 SH3 Mode

- 1) When the XRDY pin is low, it is in the wait state.
- 2) DMA transfer in the single-address mode is not supported.
- 3) DMA transfer in the dual-address mode supports the direct address transfer mode, but does not support the indirect address transfer mode.
- 4) 16-byte DMA transfer in the dual-address mode is not supported.
- 5) The XINT signal asserts low active signal.

## 12.6 SH4 Mode

- 1) When the XRDY pin is low, it is in the ready state.
- 2) At DMA transfer in the single-address mode, transfer from the main memory (SH memory) to FIFO of MINT can be performed, but transfer from MINT to the main memory cannot be performed.
- 3) DMA transfer in the single-address mode is performed in units of 32 bits or 32 bytes.
- 4) SH4-mode 32-byte DMA transfer in the dual-address mode supports inter-memory transfer, but does not support transfer from memory to FIFO.
- 5) The XINT signal asserts low active signal.

## 12.7 V832 Mode

- 1) When the XRDY pin is low, it is in the ready state.
- 2) Set the active level of DMAAK to high active in V832 mode.
- 3) DMA transfer supports the single transfer and demand transfer modes.
- 4) The XINT signal asserts high active signal. Set the V832-mode registers to high level trigger.

## 12.8 SPARClite

- 1) When the XRDY pin is low, it is in the ready state.
- 2) The SPARClite does not support the DMA transfer that issues the DREQ.
- 3) The XINT signal asserts low active signal.

## 12.9 Supported DMA Transfer Modes

	Single address mode	Dual address mode
SH3	Not supported	Direct address transfer mode supported; indirect address transfer mode not supported. Transfer is performed in 32-bit units. Cycle steal mode and burst mode supported.
SH4	Transfer performed in units of 32 bits or 32 bytes Cycle steal mode and burst mode supported	Transfer is performed in 32-bit units. Transfer to memory is performed in 32-byte units. Transfer to FIFO not supported. Cycle steal mode and burst mode supported.
V832		Transfer is performed in 32-bit units. Single transfer mode and demand transfer mode supported.
SPARC lite		

## 13 ELECTRICAL CHARACTERISTICS

### 13.1 Introduction

The values in this chapter are the final specification for MINT.

### 13.2 Maximum Rating

Maximum Rating

Parameter	Symbol	Maximum rating	Unit
Power supply voltage	$V_{DDL}^{*1}$ $V_{DDH}$	$-0.5 < V_{DDL} < 2.5$ $-0.5 < V_{DDH} < 4.0$	V
Input voltage	$V_I$	$-0.5 < V_I < V_{DDH} + 0.5 (< 4.0)$	V
Output current	$I_O$	$\pm 13$	mA
Ambient for storage temperature	TST	$-55 < TST < +125$	°C

\*1 Includes PLL power supply

#### <Notes>

- Semiconductor devices can be permanently damaged by application of stress (voltage, current, temperature, etc) in excess of absolute maximum ratings. Do not exceed these ratings.
- Do not directly connect output pins or bidirectional pins of IC products to each other or VDD or VSS to avoid the breakdown of the device. However direct connection of the output pins or bidirectional pins to each other is possible, if the output pins are designed to avoid a conflict in a timing.
- Because semiconductor devices are particularly susceptible to damaged by static electricity, you must take the measure like ground all fixtures and instruments.
- In CMOS ICs, a latch-up phenomenon is caused when an voltage exceeding Vcc or an voltage below Vss is applied to input or output pins or a voltage exceeding the rating is applied across Vcc and Vss. When a latch-up is caused, the power supply current may be dramatically increased causing resultant thermal break-down of devices. To avoid the latch-up, make sure that the voltage does not exceed the maximum rating.

## 13.3 Recommended Operating Conditions

### 13.3.1 Recommended operating conditions

Recommended Operating Conditions

Parameter	Symbol	Rating			Unit
		Min.	Typ.	Max.	
Supply voltage	V <sub>DDL</sub> *1	1.65	1.8	1.95	V
	V <sub>DDH</sub>	3.0	3.3	3.6	
	AVD	2.7	3.3	3.6	
Input voltage (High level)	V <sub>IH</sub>	2.0		V <sub>DDH</sub> +0.3	V
Input voltage (low level)	V <sub>IL</sub>	-0.3		0.8	V
Input voltage to VREF	VREF	1.05	1.10	1.15	V
VRO External resistance	RREF		2.7		K ohm
AOUT External resistance*2	RL		75		ohm
ACOMP External capacitance*3	CACOMP		0.1		uF
Ambient temperature for operation	TA	-40		85	°C

\*1 Includes PLL power supply

\*2 AOUTR, AOUTG, AOUTB pins

\*3 ACOMPR, ACOMPG, ACOMPB pins

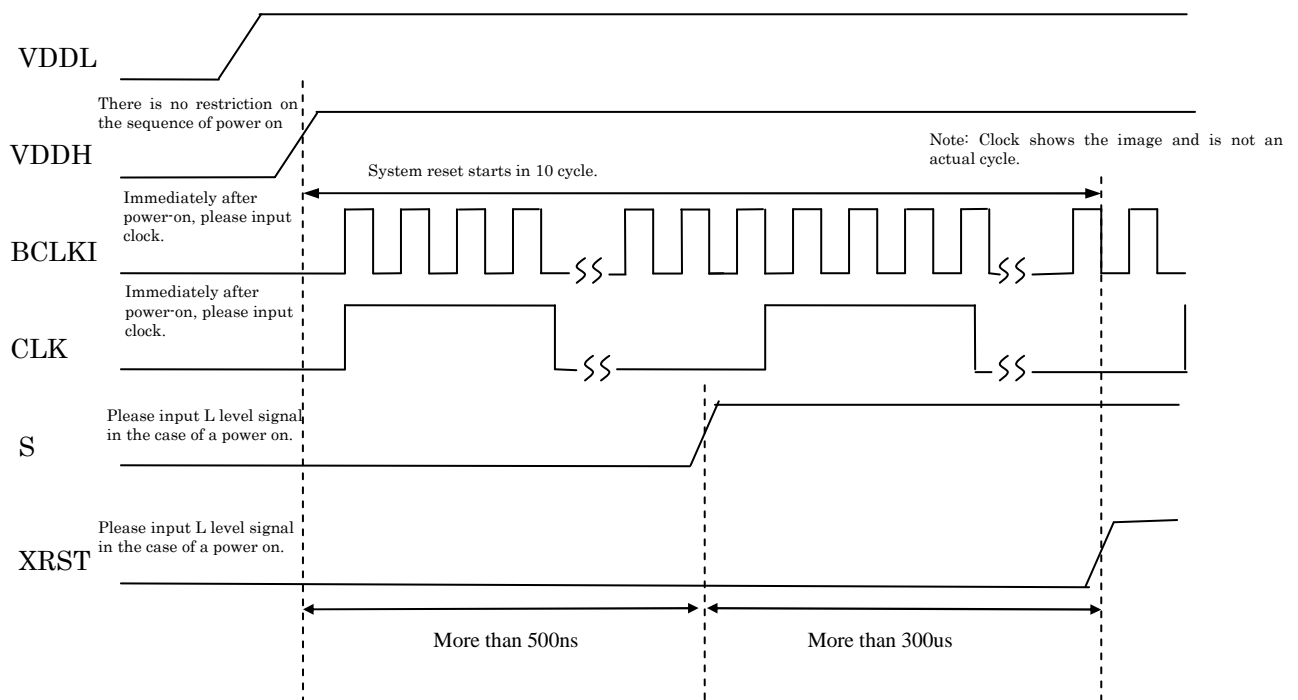
<Note>

- Recommended operating conditions are normal operating ranges for the semiconductor device. All the device's electrical characteristics are warranted when operated within these ranges. Always use semiconductor devices within the recommended operating conditions. Operation outside these ranges may adversely affect reliability and could result in device failure. No warranty is made with respect to uses, operating conditions, or combinations not represented on the manual. Users considering application outside the listed conditions are advised to contact their FUJITSU representative beforehand.

### 13.3.2 Note at power-on

- There is no restriction on the sequence of power-on/power-off between  $V_{DDL}$  and  $V_{DDH}$ . However, do not apply only  $V_{DDH}$  for more than a few seconds.
- Do not input HSYNC, VSYNC, and EO signals when the power supply voltage is not applied. (See the input voltage item in **Maximum rating**.)
- Immediately after power-on, please reset immediately because CMOS IC is in an unstable state.
  - 1) Immediately after power-on, input the “Low” level to the S and XRST pins.
  - 2) Immediately after power-on, input clock to the BCLKI pin. It is necessary to input 10 clk or more in order that “Low” level signal reach to the whole internal circuit completely.
  - 3) Immediately after power-on, input clock to the CLK pin.

It is necessary to supply the stable clock before S pin is changed “Low” level to “High” level in order that PLL is oscillated stably.
- There is a reset sequences as described below.



Immediately after power-on, input the “Low” level to the S and XRST pins. After 500ns or more, input the “High” level to S pin. After the S pin is set to “High” level, input the “Low” level to the XRST pin for 300us or more.

Immediately after power-on, input clock to the BCLKI and CLK pins.

## 13.4 DC Characteristics

### 13.4.1 DC Characteristics

**Measuring condition:**  $V_{DDL} = 1.8 \pm 1.5 \text{ V}$ ,  $V_{DDH} = 3.3 \pm 0.3 \text{ V}$ ,  $V_{SS} = 0.0 \text{ V}$ ,  $T_a = -40 \text{ to } +85^\circ\text{C}$

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
Output voltage ("High" level)	$V_{OH}$	$I_{OH}=-100\mu\text{A}$	$V_{DDH}-0.2$		$V_{DDH}$	V
Output voltage ("Low" level)	$V_{OL}$	$I_{OL}=100\mu\text{A}$	0.0		0.2	V
Output current ("High" level)	--	$V_{DDH}=3.3\text{V}\pm 0.3\text{V}$	(*1)			mA
Output current ("Low" level)	--	$V_{DDH}=3.3\text{V}\pm 0.3\text{V}$	(*1)			mA
AOUT Output current* <sup>2</sup> Full Scale* <sup>3</sup> Zero Scale	IAOUT	$V_{REF}=1.1\text{V}$ , $R_{REF}=2.7\text{k ohm}$	9.38 0	10.42 2	11.48 20	mA uA
AOUT Output Voltage* <sup>2</sup>	VAOUT	$V_{REF}=1.1\text{V}$ , $R_{REF}=2.7\text{k ohm}$ $R_L=75 \text{ ohm}$	0		0.7815	V
Input leakage current	IL				$\pm 5$	$\mu\text{A}$
Pin capacitance	C				16	pF

\*1: Please refer "V-I characteristics diagram".

**L Type:** Output characteristics of MD0-63, MDQM0-7 pins

**M Type:** Output characteristics of pins other than signals indicated by L type and H type

**H Type:** Output characteristics of XINT, DREQ, XRDY, MCLKO pins

\*2: AOUTR, AOUTG, AOUTB pin

\*3: Full Scale Output Current =  $(V_{REF}/R_{REF}) \times 25.575$

## 13.4.2 V-I characteristics diagram

Condition MIN: Process=Slow, Ta=85°C, V<sub>DD</sub>=3.0V  
TYP: Process=Typical, Ta=25°C, V<sub>DD</sub>=3.3V  
MAX: Process=Fast, Ta=-40°C, V<sub>DD</sub>=3.6V

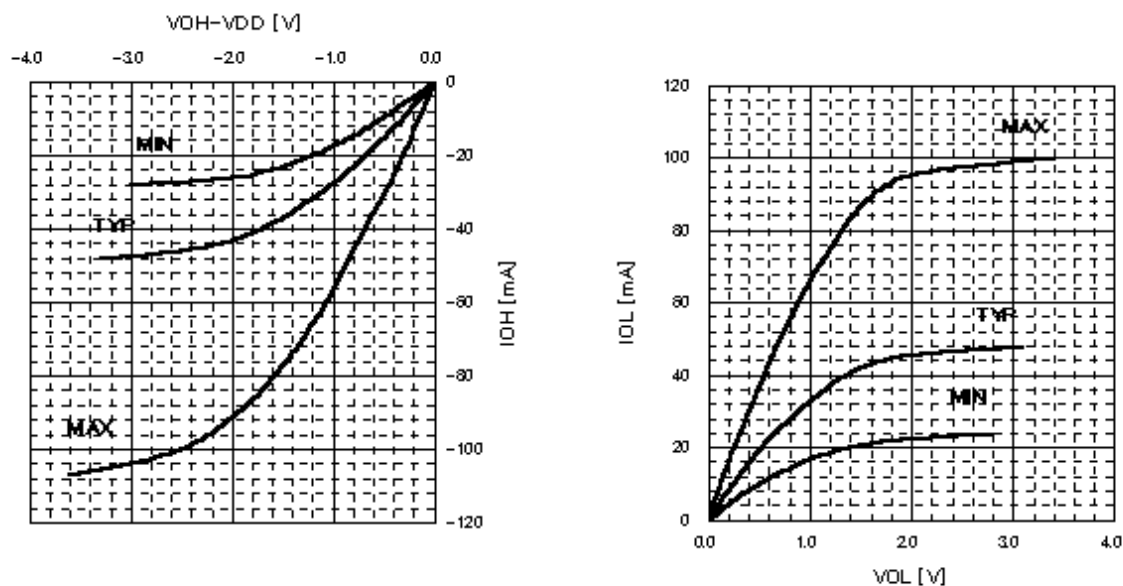


Fig. V-I characteristics L, M type

Condition MIN: Process=Slow, Ta=85°C, V<sub>DD</sub>=3.0V  
TYP: Process=Typical, Ta=25°C, V<sub>DD</sub>=3.3V  
MAX: Process=Fast, Ta=-40°C, V<sub>DD</sub>=3.6V

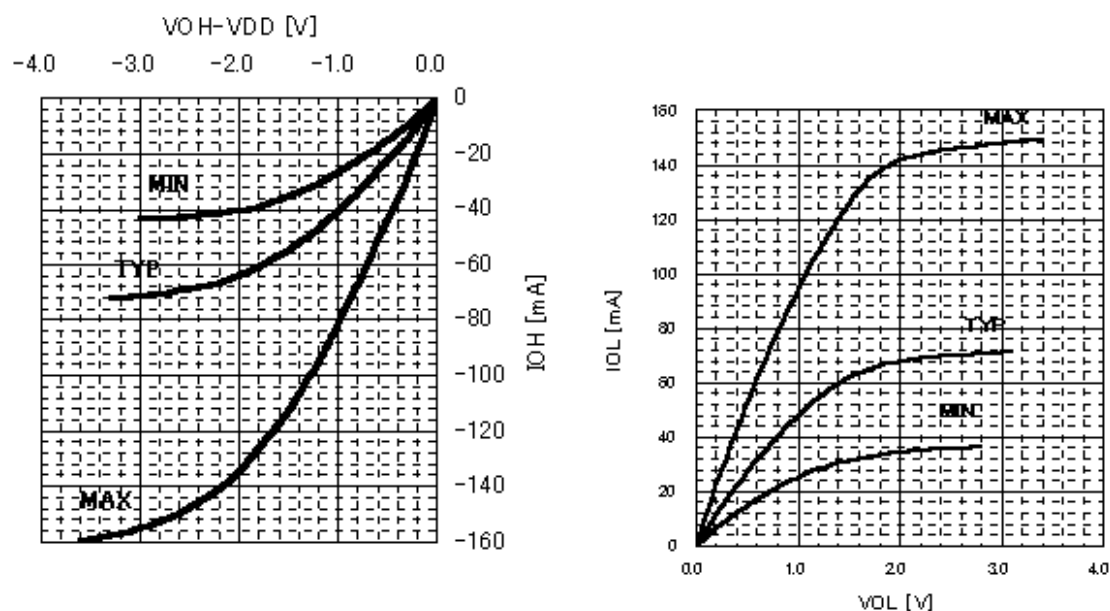


Fig. V-I characteristics H type



## 13.5 AC Characteristics

### 13.5.1 Host interface

#### Clock

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
BCLKI frequency	f <sub>BCLKI</sub>				100	MHz
BCLKI H-width	t <sub>HBCLKI</sub>		1			ns
BCLKI L-width	t <sub>LBCLKI</sub>		1			ns

#### Host interface signals

(Operating condition: external load = 20 pF)

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
Address set up time	t <sub>ADS</sub>		3.0			ns
Address hold time	t <sub>ADH</sub>		0.0			ns
XBS Set up time	t <sub>BSS</sub>		3.0			ns
XBS Hold time	t <sub>BSH</sub>		0.0			ns
XCS Set up time	t <sub>CSS</sub>		3.0			ns
XCS Hold time	t <sub>CSH</sub>		0.0			ns
XRD Set up time	t <sub>RDS</sub>		3.0			ns
XRD Hold time	t <sub>RDH</sub>		0.0			ns
XWE Set up time	t <sub>WES</sub>		5.5			ns
XWE Hold time	t <sub>WEH</sub>		0.0			ns
Write data set up time	t <sub>WDS</sub>		3.5			ns
Write data hold time	t <sub>WDH</sub>		0.0			ns
DTACK Set up time	t <sub>DAKS</sub>		3.0			ns
DTACK Hold time	t <sub>DAKH</sub>		0.0			ns
DRACK Set up time	t <sub>DRKS</sub>		3.0			ns
DRACK Hold time	t <sub>DRKH</sub>		0.0			ns
Read data delay time (for XRD)	t <sub>RDDZ</sub>		4.5		10.5	ns
Read data delay time	t <sub>RDD</sub>	*2	4.5		9.5	ns
XRDY Delay time (for XCS)	t <sub>RDYDZ</sub>		3.5		7.0	ns
XRDY Delay time	t <sub>RDYD</sub>		2.5		6.0	ns
XINT Delay time	t <sub>INTD</sub>		3.0		7.0	ns
DREQ Delay time	t <sub>DQRD</sub>		3.5		7.0	ns
MODE Hold time	t <sub>MODH</sub>	*1			20.0	ns

\*1 Hold time required for canceling reset

\*2 Valid data is output at assertion of XRDY and is retained until XRD is negated.

## 13.5.2 Video interface

### Clock

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
CLK Frequency	$f_{CLK}$			14.318		MHz
CLK H-width	$t_{HCLK}$		25			ns
CLK L-width	$t_{LCLK}$		25			ns
DCLKI Frequency	$f_{DCLKI}$				67	MHz
DCLKI H-width	$t_{HDCLKI}$		5			ns
DCLKI L-width	$t_{LDCLKI}$		5			ns
DCLKO frequency	$f_{DCLKO}$				67	MHz

### Input signals

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
HSYNC Input pulse width	$t_{WHSYNC0}$	*1	3			clock
	$t_{WHSYNC1}$	*2	3			clock
HSYNC Input setup time	$t_{SHSYNC}$	*2	10			ns
HSYNC Input hold time	$t_{HHSYNC}$	*2	10			ns
VSYNC Input pulse width	$t_{WHSYNC1}$		1			HSYNC 1 cycle

\*1 Applied only in PLL synchronization mode (CKS = 0), reference clock output from internal PLL (cycle =  $1/14 \cdot f_{CLK}$ )

\*2 Applied only in DCLKI synchronization mode (CKS = 1), reference clock = DCLKI

### Output signals

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
RGB Output delay time	$T_{RGB}$		2		10	ns
DISPE Output delay time	$t_{DEO}$		2		10	ns
HSYNC Output delay time	$t_{DHSYNC}$		2		10	ns
VSYNC Output delay time	$t_{DVSYSN}$		2		10	ns
CSYNC Output delay time	$t_{DCSYN}$		2		10	ns
GV Output delay time	$t_{DGV}$		2		10	ns

### 13.5.3 Video Capture Interface

#### Clock

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
CCLK Frequency	$f_{\text{CCLK}}$				27	MHz
CCLK H-width	$t_{\text{HCCLKI}}$		5			ns
CCLK L-width	$t_{\text{LCCLKI}}$		5			ns

#### Input signals

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
VI setup time	$t_{\text{VIS}}$		11			ns
VI hold time	$t_{\text{VIH}}$		0			ns

### 13.5.4 Graphics memory interface

**Condition:** Clock frequency=100MHz, BCLK. Printed-wiring is isometry.

**An assumed external capacitance**

Parameter	An assumed external capacitance			Unit
	Min	Typ	Max	
Board pattern	5.0		15.0	pF
SDRAM (CLK)	2.5		4.0	pF
SDRAM (D)	4.0		6.5	pF
SDRAM (A, DQM)	2.5		5.0	pF

#### Clock

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
MCLKO Frequency	f <sub>MCLKO</sub>				*1	MHz
MCLKO H-width	t <sub>HMCLKO</sub>		1.0			ns
MCLKO L-width	t <sub>LMCLKO</sub>		1.0			ns
MCLKI Frequency	f <sub>MCLKI</sub>				*1	MHz
MCLKI H-width	t <sub>HMCLKI</sub>		1.0			ns
MCLKI L-width	t <sub>LMCLKI</sub>		1.0			ns

\*1 For the bus-asynchronous mode, the frequency is 1/3 of the oscillation frequency of the internal PLL. For the bus-synchronous mode, the frequency is the same as the frequency of BCLKI.

#### Input signals

Parameter	Symbol	Condition	Rating			Unit
			Min.	Typ.	Max.	
MD Input data setup time	t <sub>MDIDS</sub>	*2	2.0			ns
MD Input data hold time	t <sub>MDIDH</sub>	*2	0.7			ns

\*2 It means against MCLKI.

There are some cases regarding AC specifications of output signals.

The following tables shows typical six cases of external SDRFAM capacitance.

(1) External SDRAM capacitance case 1

External SDRAM capacitance

SDRAM x1	Total capacitance	Unit
MCLKO	9.9pF (DRAM CLK 2.5pF, Board pattern 5pF)	pF
MA,MRAS,MCAS,MWE	7.5pF (DRAM A.DQM 2.5pF, Board pattern 5pF)	pF
MD,DQM	9.0pF (DRAM D 4pF, Board pattern 5pF)	pF

Output signals

Parameter	Symbol	Condition	Rating *1			Unit
			Min.	Typ.	Max.	
MCLKI signal delay time against MCLKO	$t_{DID}$		0		4.2	ns
MA, MRAS, MCAS, MWE Access time	$t_{MAD}$		1.0		5.0	ns
MDQM Access time	$t_{MDQMD}$		1.1		5.4	ns
MD Output access time	$t_{MDOD}$		1.1		5.4	ns

(2) External SDRAM capacitance case 2

External SDRAM capacitance

SDRAM x1	Total capacitance	Unit
MCLKO	25.4pF (DRAM CLK 4.0pF, Board pattern 15pF)	pF
MA,MRAS,MCAS,MWE	20.0pF (DRAM A.DQM 5pF, Board pattern 15pF)	pF
MD,DQM	21.5pF (DRAM D 6.5pF, Board pattern 15pF)	pF

Output signals

Parameter	Symbol	Condition	Rating *1			Unit
			Min.	Typ.	Max.	
MCLKI signal delay time against MCLKO	$t_{DID}$		0		3.5	ns
MA, MRAS, MCAS, MWE Access time	$t_{MAD}$		1.0		5.2	ns
MDQM Access time	$t_{MDQMD}$		1.2		5.5	ns
MD Output access time	$t_{MDOD}$		1.2		5.5	ns

### (3) External SDRAM capacitance case 3

#### External SDRAM capacitance

SDRAM x2	Total capacitance	Unit
MCLKO	12.4pF (DRAM CLK 2.5pF x2, Board pattern 5pF)	pF
MA,MRAS,MCAS,MWE	10.0pF (DRAM A.DQM 2.5pF x2, Board pattern 5pF)	pF
MD,DQM	9.0pF (DRAM D 4pF, Board pattern 5pF)	pF

#### Output signals

Parameter	Symbol	Condition	Rating *1			Unit
			Min.	Typ.	Max.	
MCLKI signal delay time against MCLKO	t <sub>DID</sub>		0		4.1	ns
MA, MRAS, MCAS, MWE Access time	t <sub>MAD</sub>		1.0		5.0	ns
MDQM Access time	t <sub>MDQMD</sub>		1.1		5.2	ns
MD Output access time	t <sub>MDOD</sub>		1.1		5.2	ns

### (4) External SDRAM capacitance case 4

#### External SDRAM capacitance

SDRAM x2	Total capacitance	Unit
MCLKO	29.4pF (DRAM CLK 4.0pF x2, Board pattern 15pF)	pF
MA,MRAS,MCAS,MWE	25.0pF (DRAM A.DQM 5pF x2, Board pattern 15pF)	pF
MD,DQM	21.5pF (DRAM D 6.5pF, Board pattern 15pF)	pF

#### Output signals

Parameter	Symbol	Condition	Rating *1			Unit
			Min.	Typ.	Max.	
MCLKI signal delay time against MCLKO	t <sub>DID</sub>		0		3.4	ns
MA, MRAS, MCAS, MWE Access time	t <sub>MAD</sub>		1.1		5.4	ns
MDQM Access time	t <sub>MDQMD</sub>		1.1		5.5	ns
MD Output access time	t <sub>MDOD</sub>		1.1		5.5	ns

#### (5) External SDRAM capacitance case 5

##### External SDRAM capacitance

SDRAM x4	Total capacitance	Unit
MCLKO	17.4pF (DRAM CLK 2.5pF x4, Board pattern 5pF)	pF
MA,MRAS,MCAS,MWE	15.0pF (DRAM A.DQM 2.5pF x4, Board pattern 5pF)	pF
MD,DQM	9.0pF (DRAM D 4pF, Board pattern 5pF)	pF

##### Output signals

Parameter	Symbol	Condition	Rating *1			Unit
			Min.	Typ.	Max.	
MCLKI signal delay time against MCLKO	t <sub>DID</sub>		0		3.9	ns
MA, MRAS, MCAS, MWE Access time	t <sub>MAD</sub>		1.0		5.2	ns
MDQM Access time	t <sub>MDQMD</sub>		1.0		5.0	ns
MD Output access time	t <sub>MDOD</sub>		1.0		5.0	ns

#### (6) External SDRAM capacitance case 6

##### External SDRAM capacitance

SDRAM x4	Total capacitance	Unit
MCLKO	37.3pF (DRAM CLK 4.0pF x4, Board pattern 15pF)	pF
MA,MRAS,MCAS,MWE	35.0pF (DRAM A.DQM 5pF x4, Board pattern 15pF)	pF
MD,DQM	21.5pF (DRAM D 6.5pF, Board pattern 15pF)	pF

##### Output signals

Parameter	Symbol	Condition	Rating *1			Unit
			Min.	Typ.	Max.	
MCLKI signal delay time against MCLKO	t <sub>DID</sub>		0		3.4	ns
MA, MRAS, MCAS, MWE Access time	t <sub>MAD</sub>		1.2		5.7	ns
MDQM Access time	t <sub>MDQMD</sub>		1.0		5.3	ns
MD Output access time	t <sub>MDOD</sub>		1.0		5.3	ns

### 13.5.5 PLL specifications

Parameter	Rating	Description
Input frequency (typ.)	14.31818 MHz	
Output frequency	400.9090 MHz	× 28
Duty ratio	101.6 to 93.0%	H/L Pulse width ratio of PLL output
Jitter	60 to -60 ps	Frequency tolerant of two consecutive clock cycles

CLKSEL1	CLKSEL0	Input frequency	Assured operation range (*1)
L	L	13.5 MHz	13.365 to 13.5 MHz
L	H	14.32 MHz	14.177 to 14.32 MHz
H	L	17.73 Hz	17.553 to 17.73 MHz

\*1 Assured operation input frequency range: Standard value –1%



### 13.5.6 I2C Interface

#### I2C bus timing

symbol			MIN	MAX	unit
T <sub>S2SDAI</sub>	SDA(I) setup time	standard	250		ns
		high-speed	100		ns
T <sub>H2SDAI</sub>	SCL(I) hold time	standard	0		ns
		high-speed	0		ns
T <sub>CSCLI</sub>	SCL(I) cycle time	standard	10.0		us
		high-speed	2.5		us
T <sub>WHSCLI</sub>	SCL(I) H period	standard	4.0		us
		high-speed	0.6		us
T <sub>WLSCLI</sub>	SCL(I) L period	standard	4.7		us
		high-speed	1.3		us
T <sub>CSCLO</sub>	SCL(O) cycle time	standard	$2 \cdot m + 2^{(*)2}$		PCLK <sub>1</sub>
		high-speed	$\text{int}(1.5 \cdot m) + 2^{(*)2}$		PCLK <sub>1</sub>
T <sub>WHSCLO</sub>	SCL(O) H period	standard	$m + 2^{(*)2}$		PCLK <sub>1</sub>
		high-speed	$\text{int}(0.5 \cdot m) + 2^{(*)2}$		PCLK <sub>1</sub>
T <sub>WLSCLLO</sub>	SCL(O) L period	standard	$m^{(*)2}$		PCLK <sub>1</sub>
		high-speed	$m^{(*)2}$		PCLK <sub>1</sub>
T <sub>W2SCLI</sub>	SCL(I) setup time	standard	4.0		us
		high-speed	0.6		us
T <sub>H2SCLI</sub>	SCL(I) hold time	standard	4.7		us
		high-speed	1.3		us
T <sub>WBF1</sub>	bus free time	standard	4.7		us
		high-speed	1.3		us
T <sub>S2SCLLO</sub>	SCL(O) set up time	standard	$m + 2^{(*)2}$		PCLK <sub>1</sub>
		high-speed	$\text{int}(0.5 \cdot m) + 2^{(*)2}$		PCLK <sub>1</sub>
T <sub>H2SCLLO</sub>	SCL(O) hold time	standard	$m - 2^{(*)2}$		PCLK <sub>1</sub>
		high-speed	$\text{int}(0.5 \cdot m) - 2^{(*)2}$		PCLK <sub>1</sub>
T <sub>H2SDAO</sub>	SDA(O) hold time		5		PCLK <sub>1</sub>

\*1 PCLK is an internal clock of I2C module. (16.6MHz)

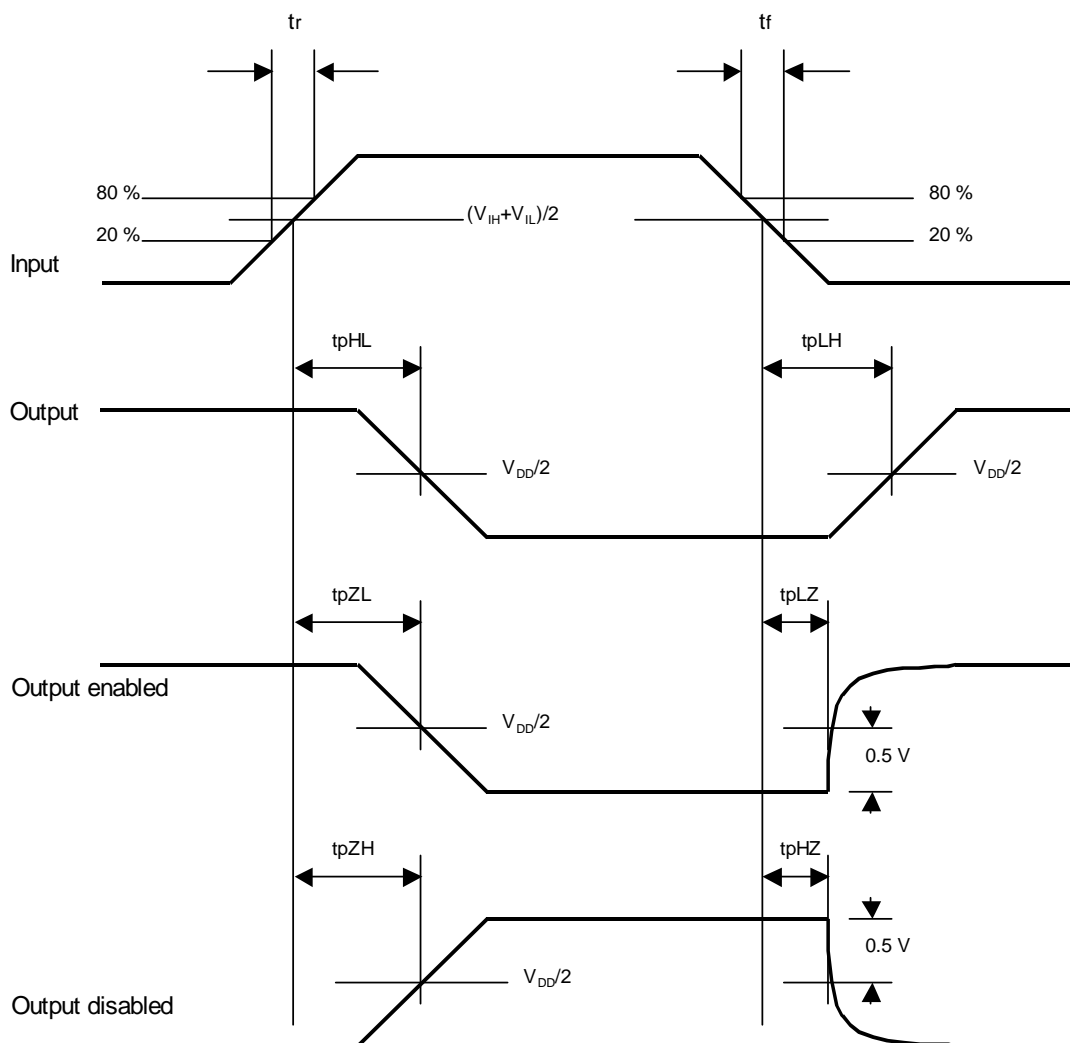
\*2 Refer to the clock control register (CCR) for the value of m.

#### Timing of interrupt

symbol		MIN	MAX	unit
T <sub>PHINTR</sub>	XINT delay (bus error)		4	PCLK
T <sub>PHINTR</sub>	XINT delay (except bus error)		4	PCLK



## 13.6 AC Characteristics Measuring Conditions



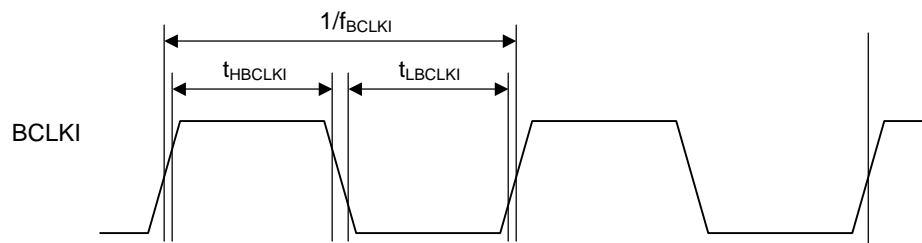
$t_r, t_f \leq 5 \text{ ns}$

$V_{IH}=2.0 \text{ V}, V_{IL}=0.8 \text{ V}$  (3.3-V CMOS interface input)

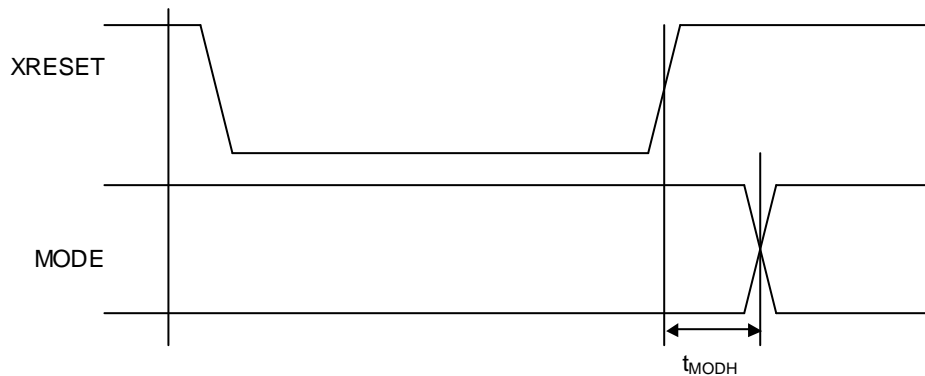
## 13.7 Timing Diagram

### 13.7.1 Host interface

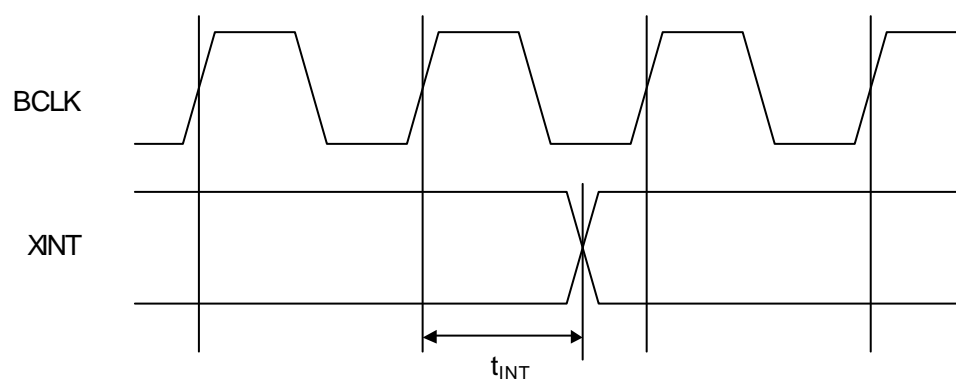
#### Clock



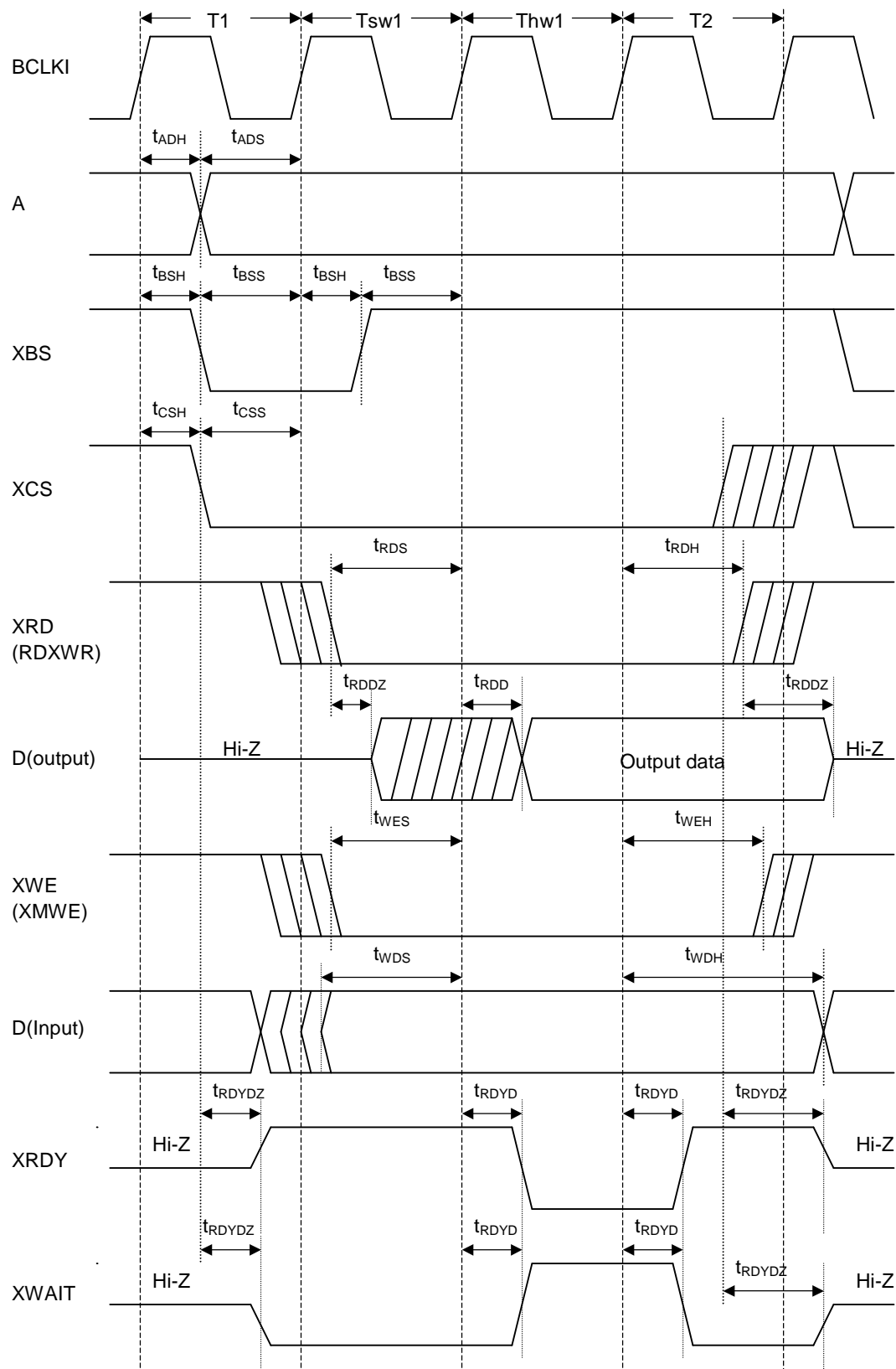
#### MODE hold time



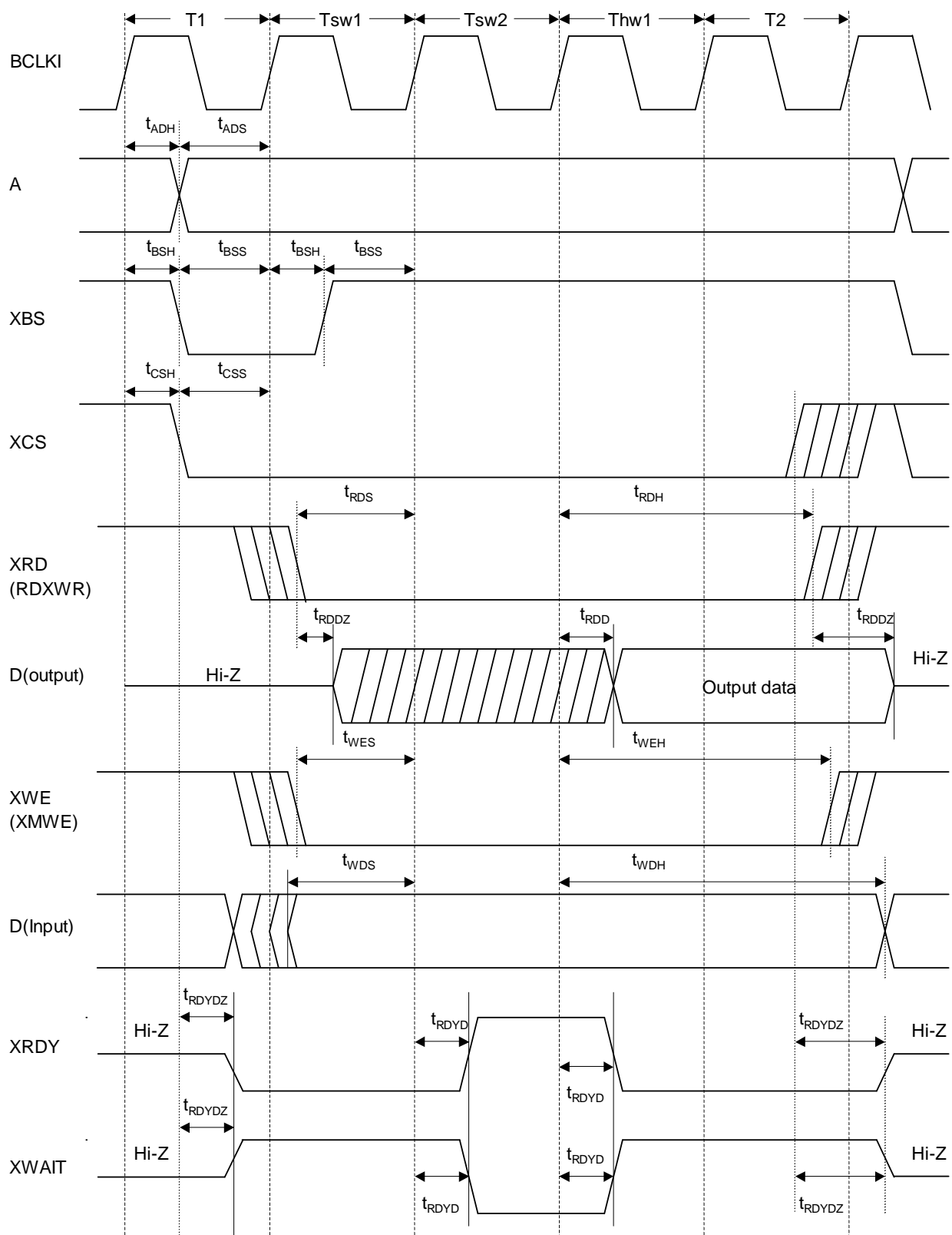
#### XINT output delay times



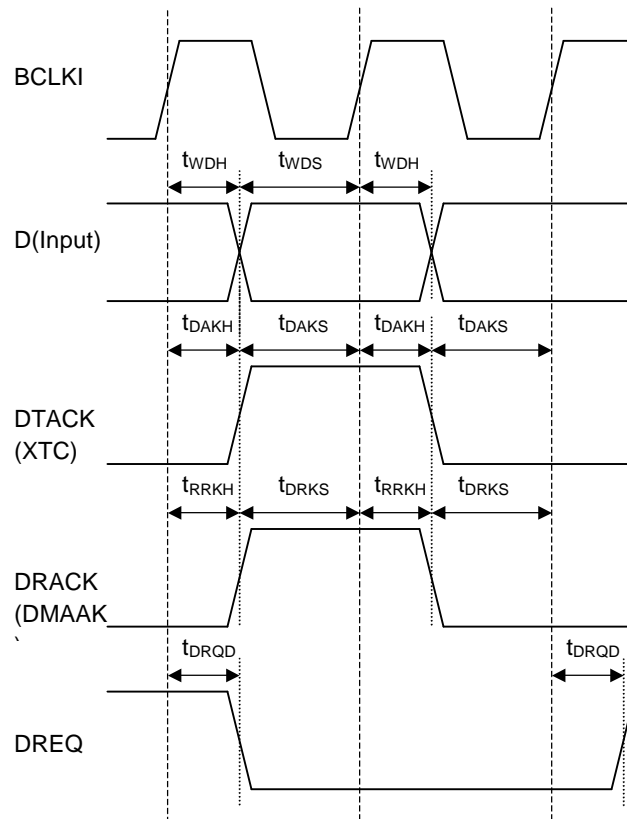
## Host bus AC timing (Normally Not Ready)



## Host bus AC timing (Normally Ready)



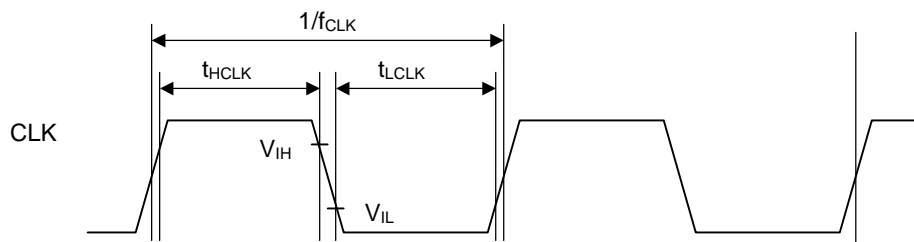
## DMA AC timing



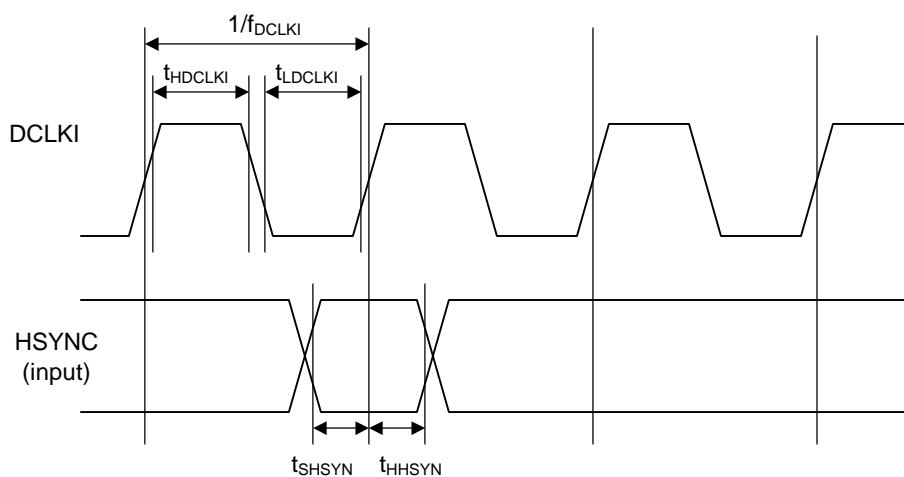
\*: The above timing diagram for the D pin is that of when a single DMA is used.  
When a dual DMA is used, see the host bus-timing diagram.

## 13.7.2 Video interface

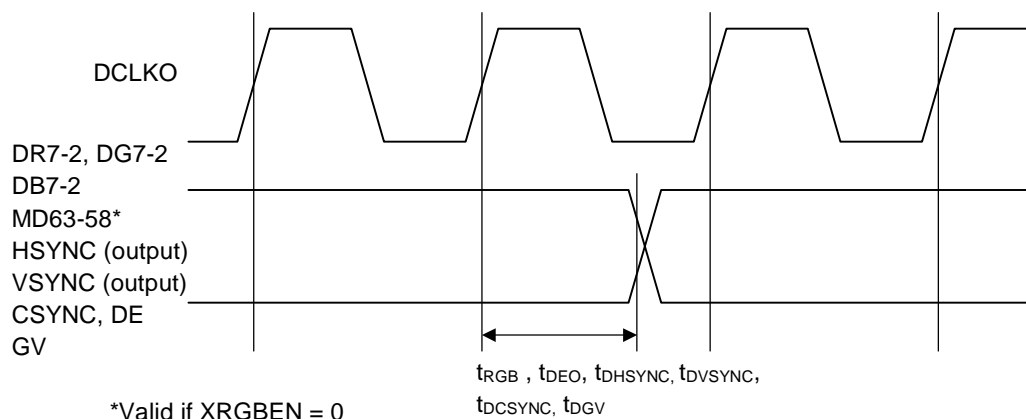
### Clock



### HSYNC signal setup/hold



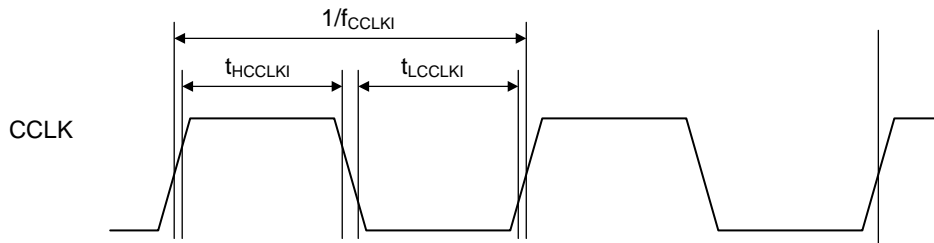
### Output signal delay



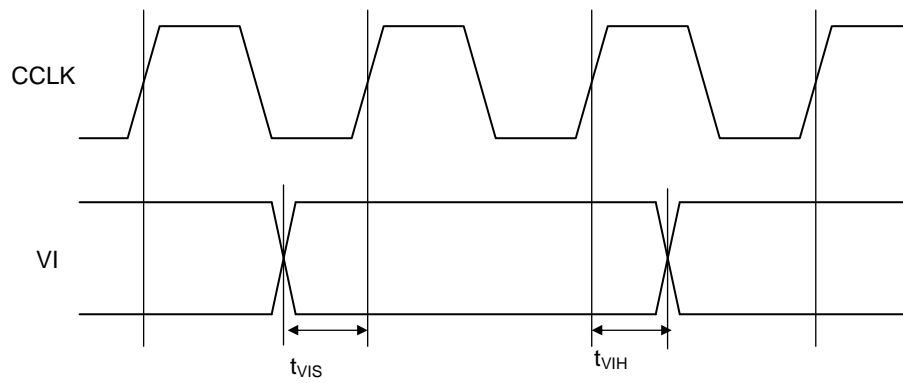


### 13.7.3 Video Capture Interface

#### Clock

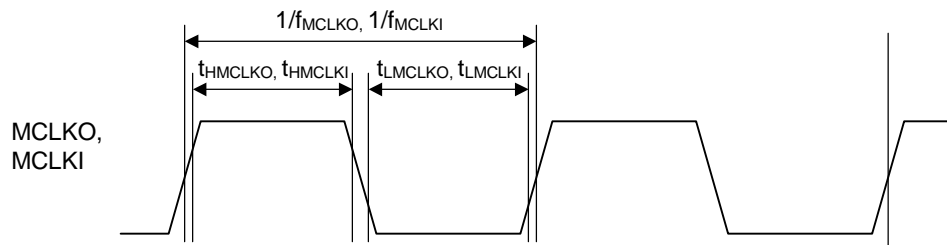


#### Video input

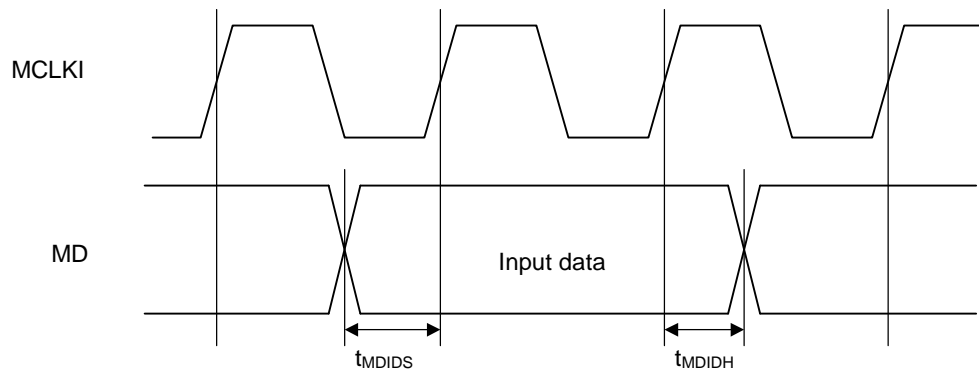


## 13.7.4 Graphics memory interface

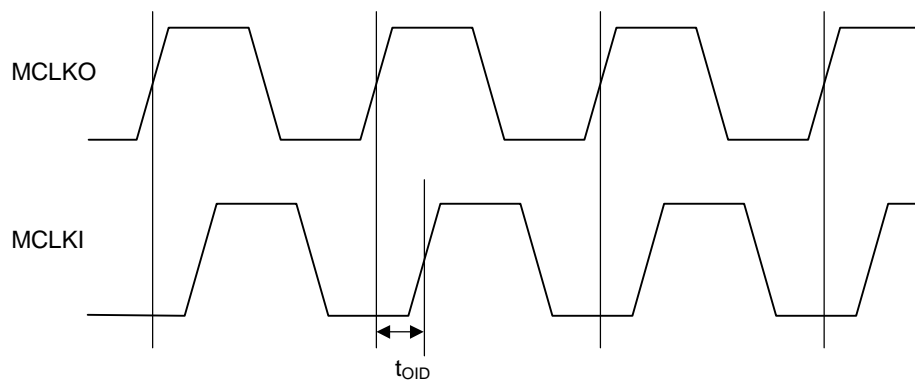
### Clock



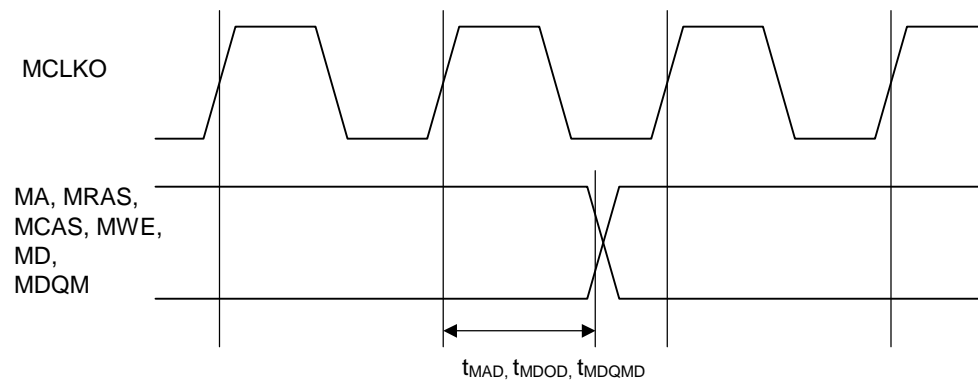
### Input signal setup/hold time



### MCLKI signal delay



## Output signal delay



13.7.5 I2C interface

I2C bus timing

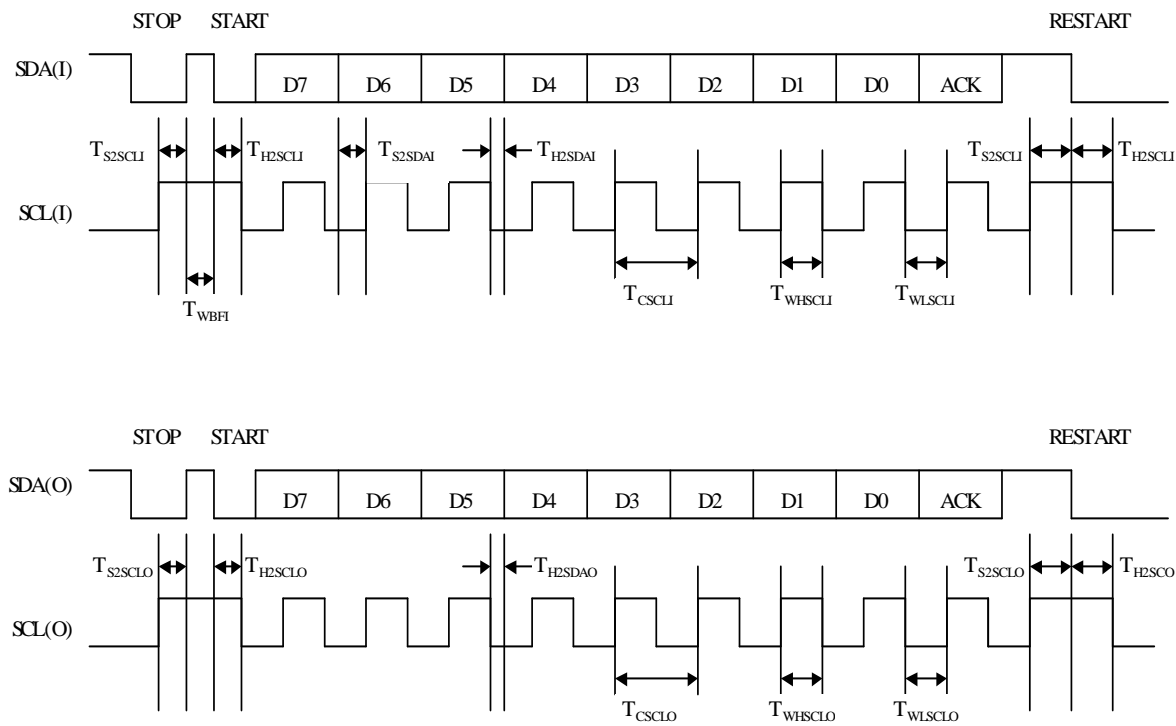
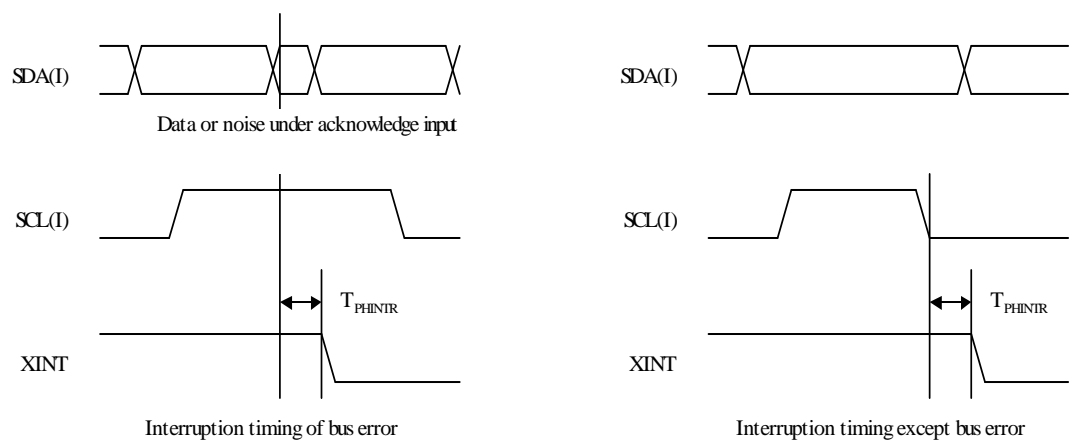


Fig7-1 Figure of I2C bus timing

Interrupt ion timing



\* INT signal is HI-active in V83x mode.

Fig 7-2 Figure of interruption timing

