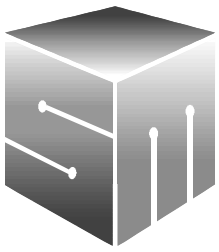


Silicon Motion, Inc.

SM501

Mobile Multimedia

Companion Chip

Databook

Version 1.01

Silicon Motion®, Inc.

SM501 MMCC Databook

Chip Versions

This document is applicable to revision AA of the SM501 (first production part).

Notice

Silicon Motion®, Inc. has made best efforts to ensure that the information contained in this document is accurate and reliable. However, the information is subject to change without notice. No responsibility is assumed by Silicon Motion, Inc. for the use of this information, nor for infringements of patents or other rights of third parties.

Copyright Notice

Copyright, 2003 Silicon Motion, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, or transmitted in any form, without the prior written consent of Silicon Motion, Inc. Silicon Motion, Inc. reserves the right to make changes to the product specification without reservation and without notice to our users.

Revision Record

Version Number	Date	Note
1.01	9/22/03	• Deleted Icc values in Table 18-2. (See SM501 Design Guide for more information about power supply design.) • Changed title of Table 12-2 from “GPIO[15:0] in 16-bit...” to “GPIO[15:0] in 8-bit...”
1.0	8/18/03	**All changes marked with change bars.** • Updated signal names in timing diagrams in Chapter 18. • Added DC characteristics in Chapter 18. • Incorporated errata 6/9/03 (new mechanical drawings for SM501, Figure 17-2) • Changed name to Mobile Multimedia Companion Chip (MMCC) • Changed version numbering to match the chip revision. • Added ROP tables for 2D Control Register, Chapter 4 • Added expanded definitions of fields for 2D Source, 2D Destination, 2D Dimension, and 2D Control registers, Chapter 4 • Corrected variable names for Panel Registers, Chapter 5 • Added example for Video Scale Register, Chapter 5 • Changed XTALPWR voltage spec to 3.3V, Table 17-2
B-1	6/13/03	• Updated arrows in Figure 1-5 Hitachi SH4 to SM501 Bus Interface diagram • Changed DRAM Control Register Bit 1 • Changed Table 17-2 Pin Descriptions B19 and B21 from Type O to I/O • Changed Figure 18-10 SH4 Read Timing • Changed Mode Select Register Bit 4 • Changed opening paragraph for 8051-Controlled External Bus Interface, and bit address modes. • Miscellaneous Timing Changed bit 31:28 host clock and updated bit 23

Version Number	Date	Note
B	4/8/03	• Added Timing Waveforms and Parameter Tables to Specifications chapter • Reformatted register descriptions • Added Soldering Profile section to Chapter 18 • Replaced Voyager GX with SM501 • Updated 1:8 scaling in Chapter 1 with maximum power of 2 values (ZV Port and 2D Engine subsections) • Updated arrows in Figure 10-1, ZV Port chapter • Added Figure 17-1, Package Outline • Renamed H18 to XTALPWR, D19 to PLLGND, E21 to PLLPWR • Added new information in Chapter 2, under Power Mode Control register on PLLs, Power and Sleep Modes, Clock Frequency • Added new field to bits [31:28] in Miscellaneous Timing register, Chapter 2 • Reorganized description of Configuration 1 & 2 registers in Chapter 2 • Added 1st/2nd clock entries to Table 1-3
A-1	2/14/03	• Global change to delete space between Voyager and GX • Added ZV Port Timing Diagram to Chapter 18 • Deleted Pin J9, K9, L9, M9, N9 from pinout diagram 17-1 • Changed opening paragraph in Chapter 17 • Added Company Confidential in header • Changed version number • Added Chapter Titles to footer • Added limits to Table 18-1

Contents

Introduction 1-1
Overview 1-1
Typical System Block Diagram 1-2
UMA Architectures 1-3
NAND Tree Scan Testing 1-4
Internal Block Description 1-5
Host CPU Memory or PCI Interface 1-5
Command Interpreter / Command List Processor 1-10
Zoom Video Port 1-10
2D Engine 1-11
Video Display Layers 1-12
LCD Panel 1-14
Analog RGB (Analog LCD or CRT) 1-19
Internal or External Memory 1-19
GPIO 1-19
Flat Panel Data 1-22
USB Controllers 1-22
DMA Controller 1-22
Interrupt Controller 1-23
Clock Control 1-23
Power Management 1-24
Memory Map and Register Space 1-26
MMIO Space 1-26
MMIO Addressing 1-27
System Configuration 2-1
Functional Overview 2-1
Register Descriptions 2-1
Configuration 1 Register Descriptions 2-3
Command List Register Descriptions 2-16
Interrupt / Debug Register Descriptions 2-18
Power Management Register Descriptions 2-25
Configuration 2 Register Descriptions 2-38
PCI Configuration Space 3-1
Register Descriptions 3-1
Drawing Engine 4-1
Functional Overview 4-1
Programmer's Model 4-1
Register Descriptions 4-1
2D Drawing Engine Registers 4-4
Color Space Conversion Registers 4-20
Display Controller 5-1
Programmer's Model 5-1
Register Descriptions 5-2
Panel Graphics Control Registers 5-7
Video Control Registers 5-18
Video Alpha Control Registers 5-27
Panel Cursor Control Registers 5-34
Alpha Control Registers 5-37
CRT Graphics Control Registers 5-42
CRT Cursor Control Registers 5-50

Palette RAM Registers 5-53
Command List Interpreter 6-1
Functional Overview 6-1
Programming 6-1
Flow Commands 6-1
Appending to the Command List 6-1
Register Descriptions 6-2
Commands 6-5
Load Memory 6-5
Load Register 6-6
Load Memory Immediate 6-6
Load Register Immediate 6-6
Load Memory Indirect 6-7
Load Register Indirect 6-7
Status Test 6-8
Finish 6-9
Goto 6-9
Gosub 6-9
Return 6-10
Conditional Jump 6-10
USB Host Controller 7-1
Functional Overview 7-1
Register Descriptions 7-1
Control and Status Registers 7-4
Memory Pointer Registers 7-12
Frame Counter Registers 7-17
Root Hub Registers 7-22
USB Slave 8-1
Register Descriptions 8-1
Address Map 8-1
Address and Power Management Registers 8-5
Interrupt Registers 8-7
Interrupt Enable Registers (R7, R9, R11) 8-10
Frame Number Registers 8-12
Index Register 8-13
IN MAXP Register 8-14
IN CSR Registers 8-15
OUT MAXP Register 8-17
OUT CSR Registers 8-18
Endpoint 0 CSR Register 8-20
OUT Write Counter Registers 8-21
GPIO/I^2C 9-1
Functional Overview 9-1
GPIO Interface 9-1
I^2C Interface 9-1
Programmer's Model 9-2
Register Descriptions 9-3
GPIO Register Descriptions 9-4
I^2C Master Register Descriptions 9-8
ZV Port 10-1
Functional Overview 10-1
ZV Port Overview 10-1
Video Capture Unit Overview 10-1
Programmer's Model 10-3

Register Descriptions 10-4
AC97-Link & I²S 11-1
Functional Overview 11-1
Programmer's Model 11-2
Register Descriptions 11-3
AC97-Link Register Descriptions 11-4
I²S Register Descriptions 11-12
8051 µ-Controller 12-1
Functional Overview 12-1
Programmer's Model 12-2
Register Descriptions 12-3
8051 µ-Controller SRAM Address Space 12-6
8051-Controlled External Bus Interface 12-7
Example: Implementing MOST for Automotive Usage 12-8
DMA Controller (DMAC) 13-1
Functional Overview 13-1
Register Descriptions 13-3
UART 14-1
Functional Overview 14-1
Modem Signals 14-1
Data Reception 14-1
Data Transmission 14-2
Hardware Flow Control 14-2
Software Flow Control 14-3
UART Timings 14-4
Transmit Engine 14-4
FIFO Reset Timing 14-4
TX Holding Register Empty Interrupt Timing 14-4
Baud Rate Generator Timing 14-4
IrDA Modulation/Demodulation 14-4
Register Descriptions 14-5
Normal Mode Registers (DLAB = 0) 14-8
Configuration Mode Registers (DLAB = 1) 14-17
Enhanced Register Set 14-19
PWM Specification 15-1
Functional Overview 15-1
Delay Counter with Interrupt 15-1
Internal Timer with Interrupt 15-1
External Pulse 15-1
Register Descriptions 15-2
Synchronous Serial Port 16-1
Functional Overview 16-1
Register Block 16-1
Clock Prescaler 16-1
Transmit FIFO 16-1
Receive FIFO 16-1
Transmit/Receive Logic 16-1
Interrupt Generation Logic 16-2
Synchronizing Registers and Logic 16-2
Operation 16-3
Configuring the SSP 16-3
Enable SSP Operation 16-3
Clock Signals 16-3
Programming the SSP Control 0 Register 16-3

Programming the SSP Control 1 Register 16-4
Bit Rate Generation 16-4
Frame Formats 16-5
Texas Instruments Synchronous Serial Frame Format 16-5
Motorola SPI Frame Format 16-6
National Semiconductor Microwire Frame Format 16-8
Examples of Master and Slave Configurations 16-11
Register Descriptions 16-14
Interrupts 16-20
Pin & Packaging Information 17-1
Signal List 17-1
Pinout 17-3
Pin Descriptions 17-4
Packaging 17-12
Specifications 18-1
Environmental 18-1
Soldering Profile 18-1
DC Characteristics 18-2
AC Timing 18-3
PCI Interface Timing 18-3
Host Interface Timing 18-7
Display Controller Timing 18-16
USB Interface Timing 18-18
ZV Port Timing 18-19
UART Timing 18-20
AC97-Link and I^2S Timing 18-22
8051 µ-Controller Timing 18-25
Local SDRAM Timing 18-28

Figures

Figure 1-1: System Block Diagram 1-1
Figure 1-2: Example System Diagram 1-2
Figure 1-3: System Configurations 1-3
Figure 1-4: Internal Block Diagram 1-5
Figure 1-5: Hitachi SH4 to the SM501 Bus Interface 1-6
Figure 1-6: Intel XScale (PXA250/255) to the SM501 Bus Interface 1-8
Figure 1-7: NEC MIPS to the SM501 Bus Interface 1-9
Figure 1-8: PCI Bus Interconnection 1-10
Figure 1-9: Video Layers and Data Processing 1-12
Figure 1-10: Typical 24-bit TFT Panel Interface 1-15
Figure 1-11: Typical 18-bit TFT Panel Interface 1-16
Figure 1-12: Typical 24-bit LVDS Interface (DS90C385) 1-17
Figure 1-13: GPIO Layout 1-19
Figure 1-14: Clock Tree 1-24
Figure 1-15: Power State Diagram 1-25
Figure 1-16: Memory Map 1-26
Figure 1-17: MMIO Space 1-27
Figure 2-1: System Configuration Register Space 2-1
Figure 2-2: Configuration Register Space 1 2-3
Figure 2-3: Command List Register Space 2-16
Figure 2-4: Interrupt / Debug Register Space 2-18
Figure 2-5: Power Management Register Space 2-25
Figure 2-6: Configuration Register Space 2 2-38
Figure 3-1: PCI Configuration Register Space 3-2
Figure 4-1: Drawing Engine Register Space 4-1
Figure 4-2: 2D Drawing Register Space 4-4
Figure 4-3: Color Space Conversion Register Space 4-20
Figure 5-1: Display Controller Register Space 5-5
Figure 5-2: Video Layers 5-6
Figure 5-3: Panel Graphics Control Register Space 5-7
Figure 5-4: Video Windowing 5-8
Figure 5-5: Video Control Register Space 5-18
Figure 5-6: Video Alpha Control Register Space 5-27
Figure 5-7: Panel Cursor Control Register Space 5-34
Figure 5-8: Alpha Control Register Space 5-37
Figure 5-9: CRT Graphics Control Register Space 5-42
Figure 5-10: CRT Cursor Control Register Space 5-50
Figure 5-11: Palette RAM Register Space 5-53
Figure 6-1: Command List Register Space 6-2
Figure 7-1: USB Control and Status Partition Register Space 7-4
Figure 7-2: USB Memory Pointer Partition Register Space 7-12
Figure 7-3: USB Frame Counter Partition Register Space 7-17
Figure 7-4: USB Root Hub Partition Register Space 7-22
Figure 8-1: USB Slave Register Space 8-4
Figure 9-1: I^2C Interface Block Diagram 9-1
Figure 9-2: GPIO / I^2C Master Register Space 9-2

Figure 9-3: GPIO Register Space 9-4
Figure 9-4: I^2C Master Register Space 9-8
Figure 10-1: TV Decoder Interface through the ZV Port 10-1
Figure 10-2: ZV Port Register Space 10-3
Figure 11-1: AC97-Link and I^2S Interface Block Diagram 11-1
Figure 11-2: AC97-Link and I^2S Register Space 11-2
Figure 11-3: AC97-Link Register Space 11-4
Figure 11-4: I^2S Register Space 11-12
Figure 12-1: 8051 µ-Controller Register Space 12-2
Figure 12-2: 8051 µ-Controller SRAM Address Space 12-6
Figure 12-3: 8051 µ-Controller to MOST Controller Connections 12-8
Figure 13-1: SM501 Functional Block Connections to Memory Controllers 13-1
Figure 13-2: DMA Channel 0 13-2
Figure 13-3: DMA Channel 1 13-2
Figure 13-4: DMA Register Space 13-3
Figure 14-1: UART and IrDA Register Space 14-7
Figure 14-2: UART / IrDA Normal Mode Registers (DLAB = 0) 14-8
Figure 14-3: Configuration Mode Registers (DLAB = 1) 14-17
Figure 14-4: Enhanced Register Set 14-19
Figure 15-1: PWM Register Space 15-2
Figure 16-1: Texas Instruments Synchronous Serial Frame Format (Single Transfer) 16-5
Figure 16-2: TI Synchronous Serial Frame Format (Continuous Transfer) 16-6
Figure 16-3: Motorola SPI Frame Format (Single Transfer) 16-7
Figure 16-4: Motorola SPI Frame Format (Continuous Transfer) SPO = 0, SPH = 0 16-7
Figure 16-5: Motorola SPI Frame Format with Ph = 0 16-8
Figure 16-6: Motorola SPI Frame Format with Ph = 1 16-8
Figure 16-7: Microwire Frame Format, Single Transfer 16-9
Figure 16-8: Microwire Frame Format, Continuous Transfers 16-10
Figure 16-9: Microwire Frame Format, SFRMIN Input Setup and Hold Requirements 16-10
Figure 16-10: SSP Master Coupled to Two Slaves 16-11
Figure 16-11: SSPMS (PL021) Master Coupled to Two SPI Slaves 16-12
Figure 16-12: SPI Master Coupled to Two SSP Slaves 16-13
Figure 16-13: SSP Register Space 16-14
Figure 16-14: Single SSP Register Space 16-15
Figure 17-1: Pinout for the SM501 17-3
Figure 17-2: SM501 Package Outline (two sheets) 17-12
Figure 18-1: Temperature Profile 18-1
Figure 18-2: PCI Clock Timing 18-3
Figure 18-3: PCI Clock to Output Timing 18-4
Figure 18-4: PCI Clock to Input Timing 18-4
Figure 18-5: PCI Bus Timing Diagram 18-5
Figure 18-6: XScale System DRAM (Master) Timing 18-7
Figure 18-7: XScale Read Timing 18-8
Figure 18-8: XScale Write Timing 18-9
Figure 18-9: SH4 System DRAM (Master) Timing 18-11
Figure 18-10: SH4 Read Timing 18-12
Figure 18-11: SH4 Write Timing 18-12
Figure 18-12: NEC System DRAM (Master) Timing 18-14
Figure 18-13: NEC DRAM (Slave) Timing 18-15
Figure 18-14: FP and FP_DISP Timing 18-16

Figure 18-15: FHSYNC and FVSYNC Timing 18-17
Figure 18-16: USB Timing 18-18
Figure 18-17: ZV Port Timing 18-19
Figure 18-18: UART Transmit Timing 18-20
Figure 18-19: UART Receive Timing 18-20
Figure 18-20: UART Modem Timing 18-21
Figure 18-21: AC-Link I/O Timing 18-22
Figure 18-22: AC97-Link Cycle Timing 18-22
Figure 18-23: I^2S Transmitter Timing 18-23
Figure 18-24: I^2S Receiver Timing 18-23
Figure 18-25: 8051 Read Timing (Address High Latched) 18-25
Figure 18-26: 8051 Write Timing (Address High Latched) 18-25
Figure 18-27: 8051 Read Timing (Address High Without Latch) 18-26
Figure 18-28: 8051 Write Timing (Address High Without Latch) 18-26
Figure 18-29: Local SDRAM Timing 18-28

Tables

Table 1-1: SM501 Supported Host Interfaces and Memory Configurations . . . 1-2
Table 1-2: SM501 Display Data . . . 1-18
Table 1-3: Function Summary, GPIO[63:56] . . . 1-20
Table 1-4: GPIO Strap Pins . . . 1-21
Table 1-5: MMIO Base Addresses for Host Interface . . . 1-28
Table 2-1: SM501 System Configuration Register Summary . . . 2-2
Table 2-2: Programmable Clock Branches . . . 2-37
Table 3-1: PCI Configuration Space Register Summary . . . 3-1
Table 4-1: Drawing Engine Register Summary . . . 4-2
Table 4-2: ROP3 Operations . . . 4-9
Table 5-1: Display Controller Register Summary . . . 5-2
Table 6-1: Command List Register Summary . . . 6-2
Table 6-2: SM501 Commands . . . 6-5
Table 7-1: USB Host Controller Driver Register Summary . . . 7-1
Table 8-1: USB Device Internal Register Summary . . . 8-2
Table 8-2: FIFO Register Address Map . . . 8-3
Table 9-1: GPIO and I^2C Register Summary . . . 9-3
Table 10-1: ZV Port Register Summary . . . 10-4
Table 11-1: AC97-Link and I^2S Port Register Summary . . . 11-3
Table 12-1: 8051 µ-Controller Register Summary . . . 12-3
Table 12-2: GPIO[15:0] in 16-bit and 12-bit Modes . . . 12-7
Table 13-1: DMA Controller Register Summary . . . 13-3
Table 14-1: Software Flow Control Configuration Bits . . . 14-3
Table 14-2: UART0 and IrDA0 Register Summary . . . 14-5
Table 14-3: UART1 and IrDA1 Register Summary . . . 14-6
Table 14-4: Baud Rate Divisor . . . 14-17
Table 15-1: PWM Register Summary . . . 15-2
Table 16-1: SSP Register Summary . . . 16-14
Table 17-1: Signal Summary for the SM501 . . . 17-1
Table 17-2: Pin Descriptions . . . 17-4
Table 18-1: Absolute Maximum Ratings . . . 18-2
Table 18-2: DC Characteristics . . . 18-2
Table 18-3: PCI Clock Timing Parameters . . . 18-3
Table 18-4: PCI I/O Timing Parameters . . . 18-4
Table 18-5: PCI Bus Timing Parameters (33 MHz) . . . 18-6
Table 18-6: XScale Timing Parameters . . . 18-10
Table 18-7: SH4 Timing Parameters . . . 18-13
Table 18-8: NEC MIPS Timing Parameters . . . 18-15
Table 18-9: Color TFT Interface Timing Parameters . . . 18-17
Table 18-10: USB Timing Parameters . . . 18-18
Table 18-11: ZV Port Timing Parameters . . . 18-19
Table 18-12: UART Timing Parameters . . . 18-21
Table 18-13: AC97-Link Timing Parameters . . . 18-23
Table 18-14: I^2S Timing Parameters . . . 18-23
Table 18-15: 8051 Timing Parameters . . . 18-27
Table 18-16: Local SDRAM Timing Parameters . . . 18-29

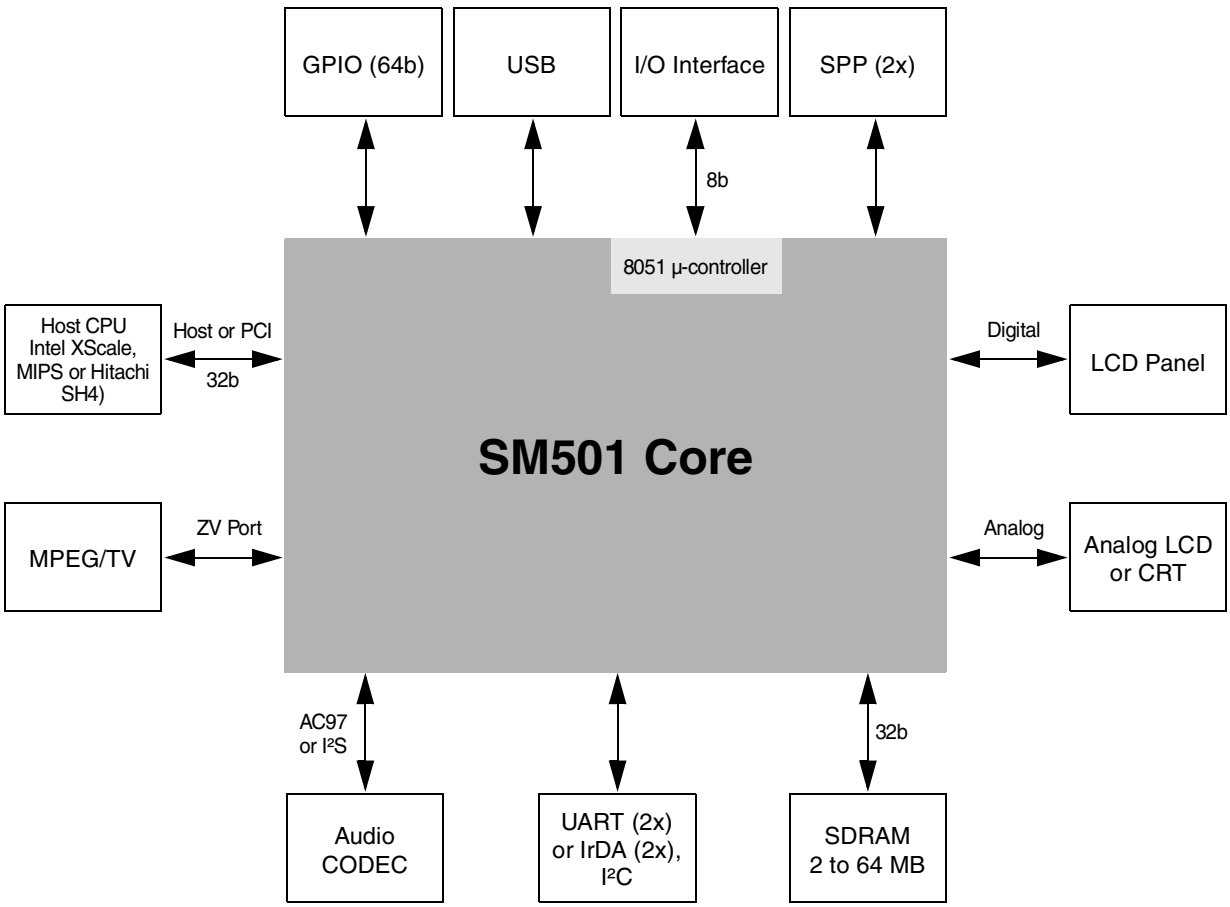
1 Introduction

Overview

SM501 is a Mobile Multimedia Companion Chip (MMCC™) device, packaged in a 297-pin BGA. Designed to complement needs for the embedded industry, it provides video and 2D capability. To help reduce system costs, it supports a wide variety of I/O, including analog RGB and digital LCD Panel interfaces, 8-bit parallel interface, USB, UART, IrDA, Zoom Video, AC97 or I^2S, SSP, PWM, and I^2C. There are additional GPIO bits that can be used to interface to external devices as well.

The 2D engine includes a front-end color space conversion with 4:1 and 1:8 scaling support. The video engine supports two different video outputs (Dual Monitor), at 8-bit, 16-bit, or 32-bit per pixel and a 3-color hardware cursor per video output. The LCD panel video pipe supports a back-end YUV color space conversion with 4:1 and $1:2^{12}$ scaling. A Zoom Video (ZV) port is also included to interface to external circuitry for MPEG decode or TV input.

Figure 1-1: System Block Diagram



Typical System Block Diagram

SM501 supports a variety of interfaces to the Host CPU. For systems with a PCI bus, the SM501 may be attached directly to the PCI bus. For systems with the Intel XScale, Hitachi SH-4 or NEC MIPS V_R4122/4131 CPU, the SM501 must be connected through a 32-bit SRAM-like memory interface directly to the CPU. The PCI bus interface and the memory interface use many of the same I/O pins, so their use is mutually exclusive.

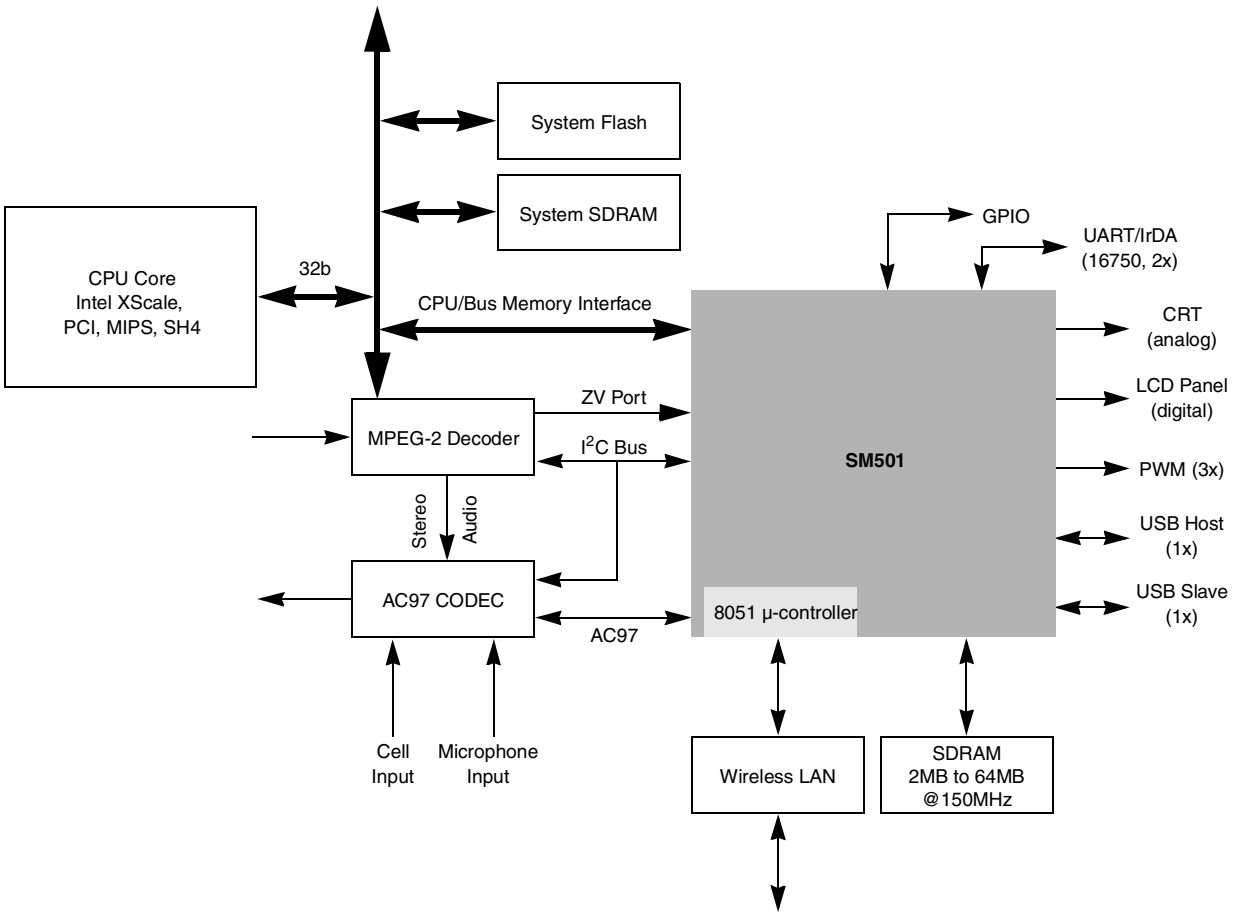
The SM501 will support the following host interface and memory configurations:

Table 1-1: SM501 Supported Host Interfaces and Memory Configurations

Memory Type / Interface	Host Interface (PCI *or* memory)	Frame Memory Interface
UMA_{CPU}	32-bit	None
UMA_{LOCAL} or Frame	32-bit	32-bit

Figure 1-2 shows a possible system configuration with the SM501 connected through a CPU SRAM-like memory interface.

Figure 1-2: Example System Diagram



UMA Architectures

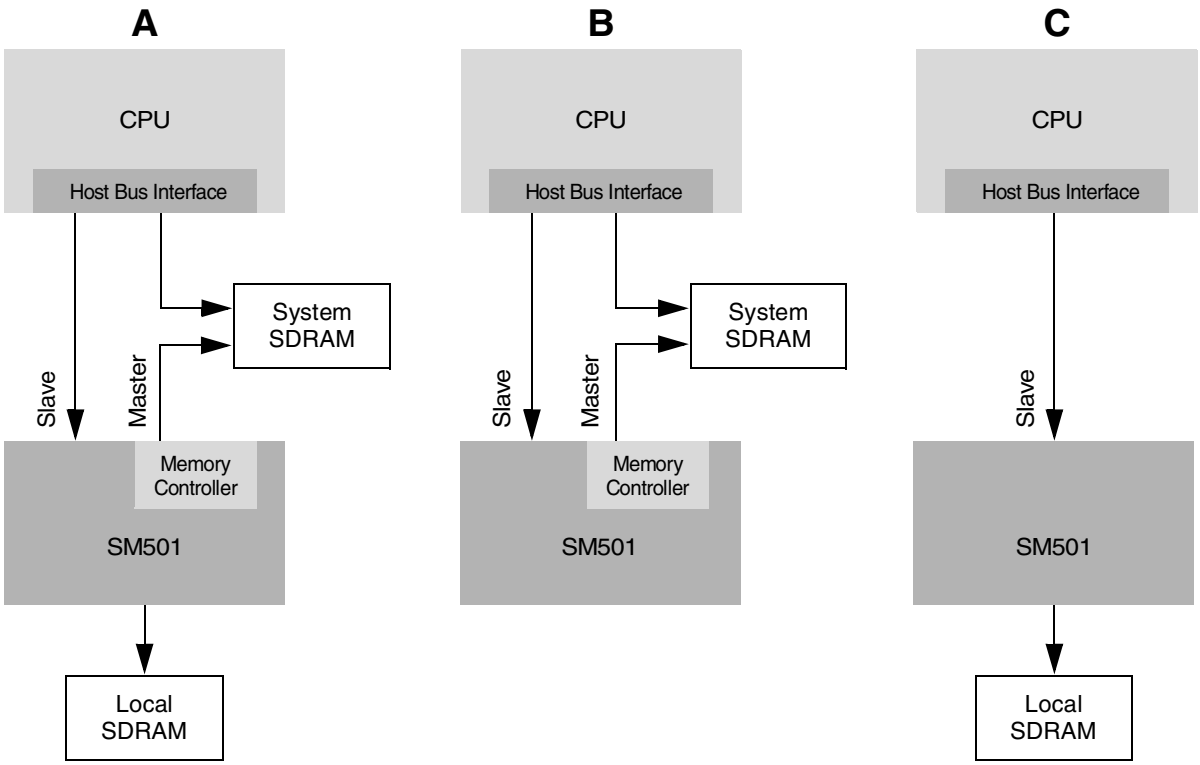
The SM501 supports three different system configurations, two of which are Unified Memory Architecture (UMA) where the system memory and frame buffer share the memory. Figure 1-3 shows the possible configurations.

Configuration A is a configuration without UMA and has the highest performance. The System SDRAM contains the system memory and the local SDRAM contains the frame buffer. The SM501 acts as a slave on the Host Bus Interface of the CPU. It controls the local SDRAM. The SM501 can also access the system SDRAM by acquiring the bus and act as a master.

Configuration B is an UMA architecture where the system SDRAM contains both the system memory and the frame buffer. The SM501 acts as a slave on the Host Bus Interface of the CPU. The CPU can burst to the system SDRAM as normal, and the SM501 can access the system SDRAM by acquiring the bus and act as a master. Bandwidth is very limited in this configuration since both the CPU and the SM501 compete for the same Host Bus Interface.

Configuration C is the preferred UMA architecture where the local SDRAM contains both the system memory and frame buffer. Since the local SDRAM bus runs at 150 MHz, there is more bandwidth in this configuration than in configuration B. The SM501 acts as a slave on the Host Bus Interface of the CPU.

Figure 1-3: System Configurations



NAND Tree Scan Testing

The SM501 NAND Tree scan test circuit is designed for verifying the device being properly soldered to the board. It detects opened/shorted traces of a signal pin with a simple test pattern which, for this particular case, only ~200 vectors in length. The NAND Tree scan test circuit uses Combinational logic; therefore, no clock pulses are required during the testing.

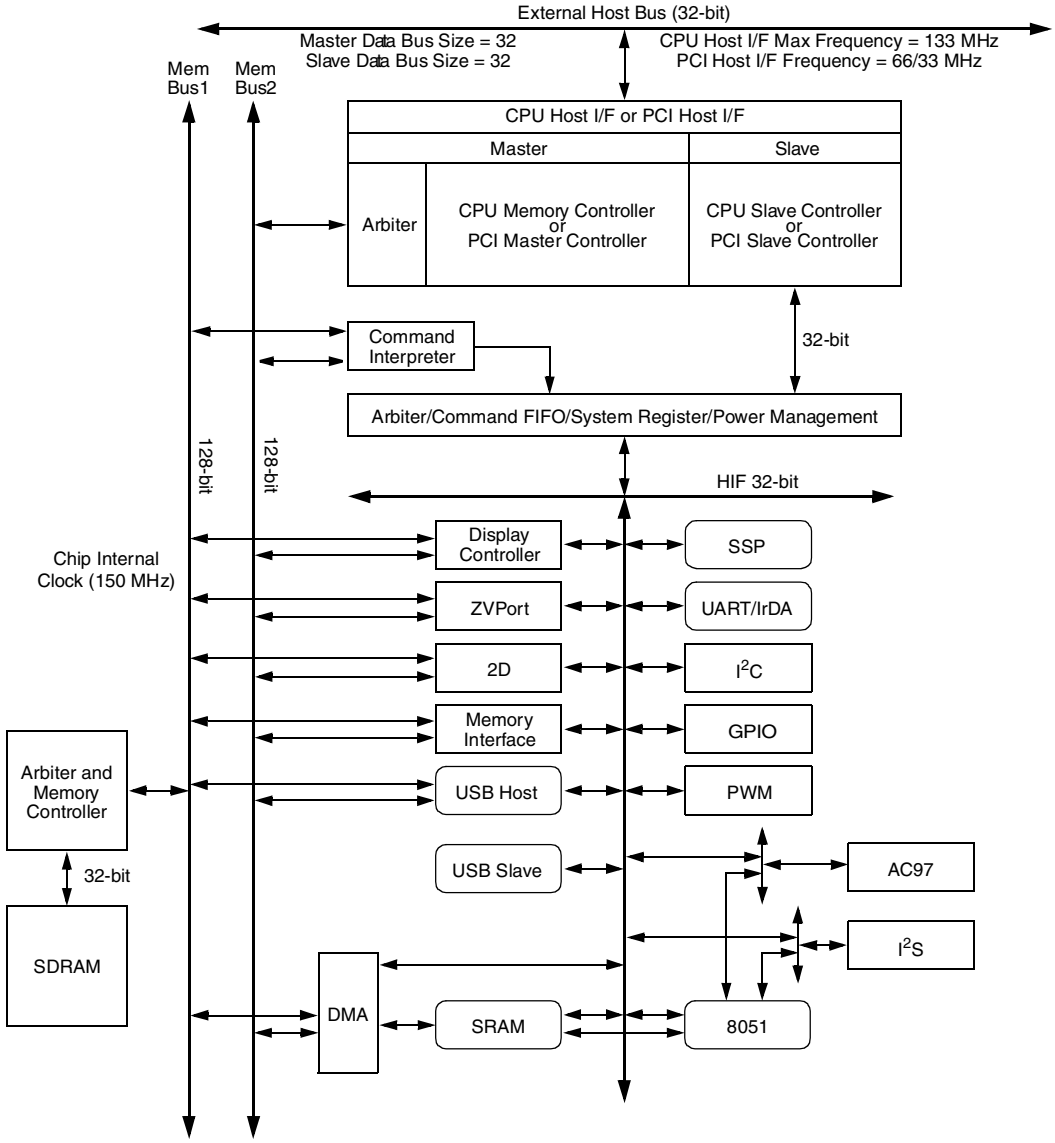
General Information

The SM501 NAND Tree scan test circuit is a long chain of 2-input NAND gates. The first pin of the NAND chain is an input, the last pin of the chain is an output. In order to set up the SM501 for NAND Tree scan testing, program the Test pin to 0x1. All VDD pins and Analog pins RED, GREEN, BLUE, IREF, and Test pins are not included in the scan chain.

Internal Block Description

As can be seen from Figure 1-4, the SM501 uses three major buses for internal communication. The HIF bus (32 bits wide) is used to read and write internal registers. A 128-bit wide data bus (MEM BUS2) is also provided to move data into and out of the Host CPU interface. A second 128-bit wide bus (MEM BUS1) is used to transfer data to and from the SM501's internal (or external) memory.

Figure 1-4: Internal Block Diagram



Host CPU Memory or PCI Interface

The SM501 supports two mutually exclusive modes of interfacing with the host CPU. The first option is to configure the SM501 as a memory-mapped device located off the host system's CPU to memory interface. In this case, the SM501 supports a 32-bit interface for commands/status and a 32-bit interface for data transfer. With a typical Intel XScale processor interface, this allows for a peak bandwidth of 400 MB/s.

For UMA designs, the SM501 supports an 8x32-bit memory cache at the host interface.

The second configuration option is to use the SM501 as a PCI device on a PCI bus. In this mode, the SM501 supports PCI-1X and 2X for a maximum bus throughput of 250 MB/s.

The following sections summarize the interface connections between the SM501 and a host CPU or PCI bus. See the *SM501 MMCC Design Guide* for more information.

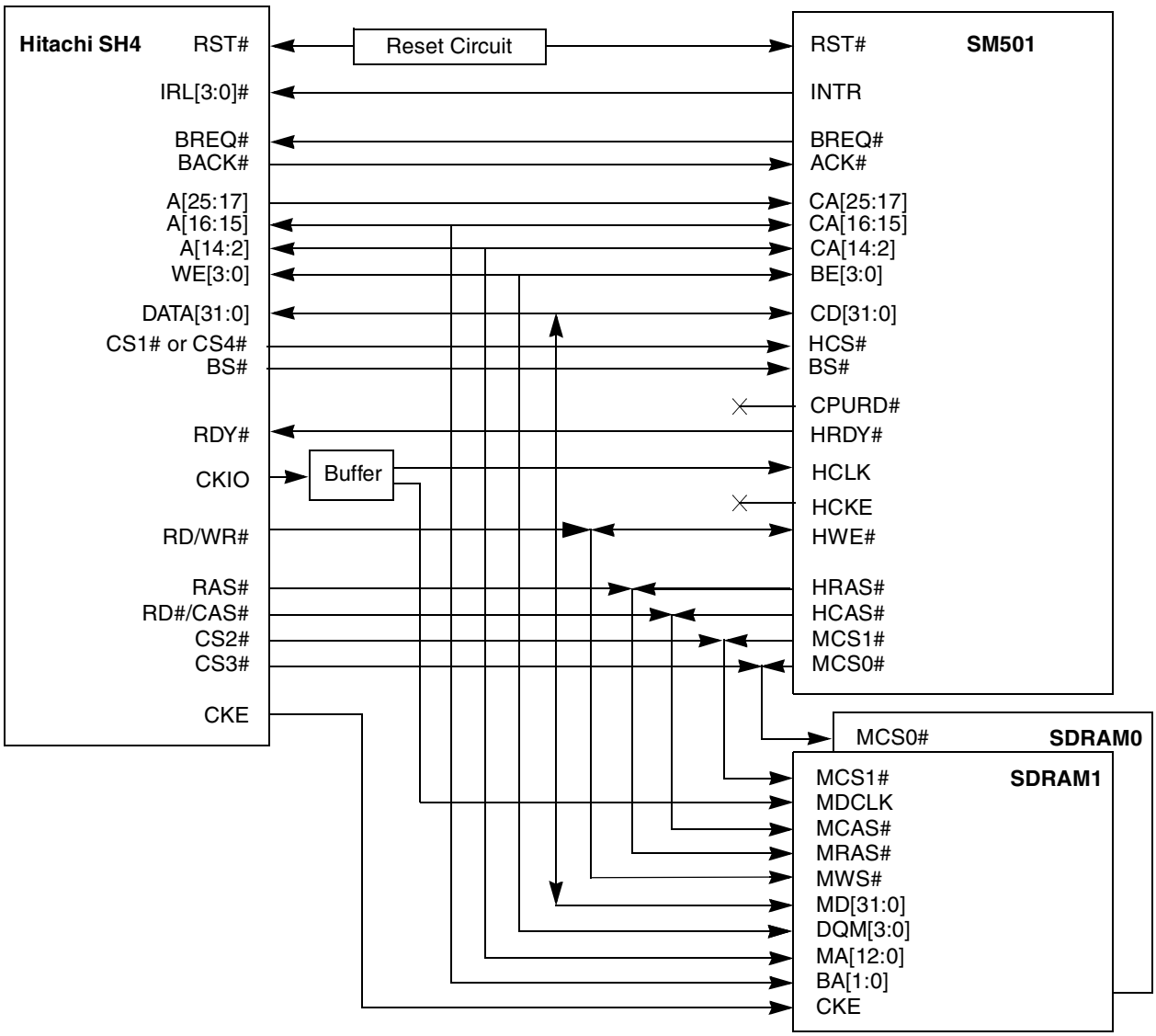
Hitachi SH4

When the SM501 is interfaced to the Hitachi SH4 host bus, it can run in two different host interface modes:

1. In Master mode: SDRAM mode.
2. In Slave mode: Byte Control SRAM mode.

Figure 1-5 shows a typical system-level hookup between the SM501 device and the Hitachi SH4.

Figure 1-5: Hitachi SH4 to the SM501 Bus Interface



Design Notes:

1. During the SM501 Bus Master mode, because the SH4 always drives the SDRAM clock, the timing for the SDRAM signals being driven by the SM501 (CAS#, RAS#, WE#, data, etc.) must reference the memory MDCLK input and meet the memory specifications.
2. In the SM501 Slave mode, the SH4 CPU drives RD/WR# (WE#) for the SM501 and SDRAM. In Bus Master mode, the SM501 drives this line.
3. The SH4 supports up to 64 MB with each chip select. Depending on the memory size and configuration, address [14:12] can be used for the bank address or the memory address. The design shown in Figure 1-5 supports two groups of 64MB memory.
4. The SM501 works as a byte-enable SRAM with the SH4. The SH4 supports byte-enable SRAM only with CS1 or CS4.
5. The SH4 can be programmed for 4-level or 16-level interrupts. The SM501 interrupt output can be connected to one of the 4-level interrupt inputs, depending on the system and software designs. To connect to a 16-level interrupt, an external circuit is required.
6. For a discussion of clock buffering, see the *SM501 MMCC Design Guide*.

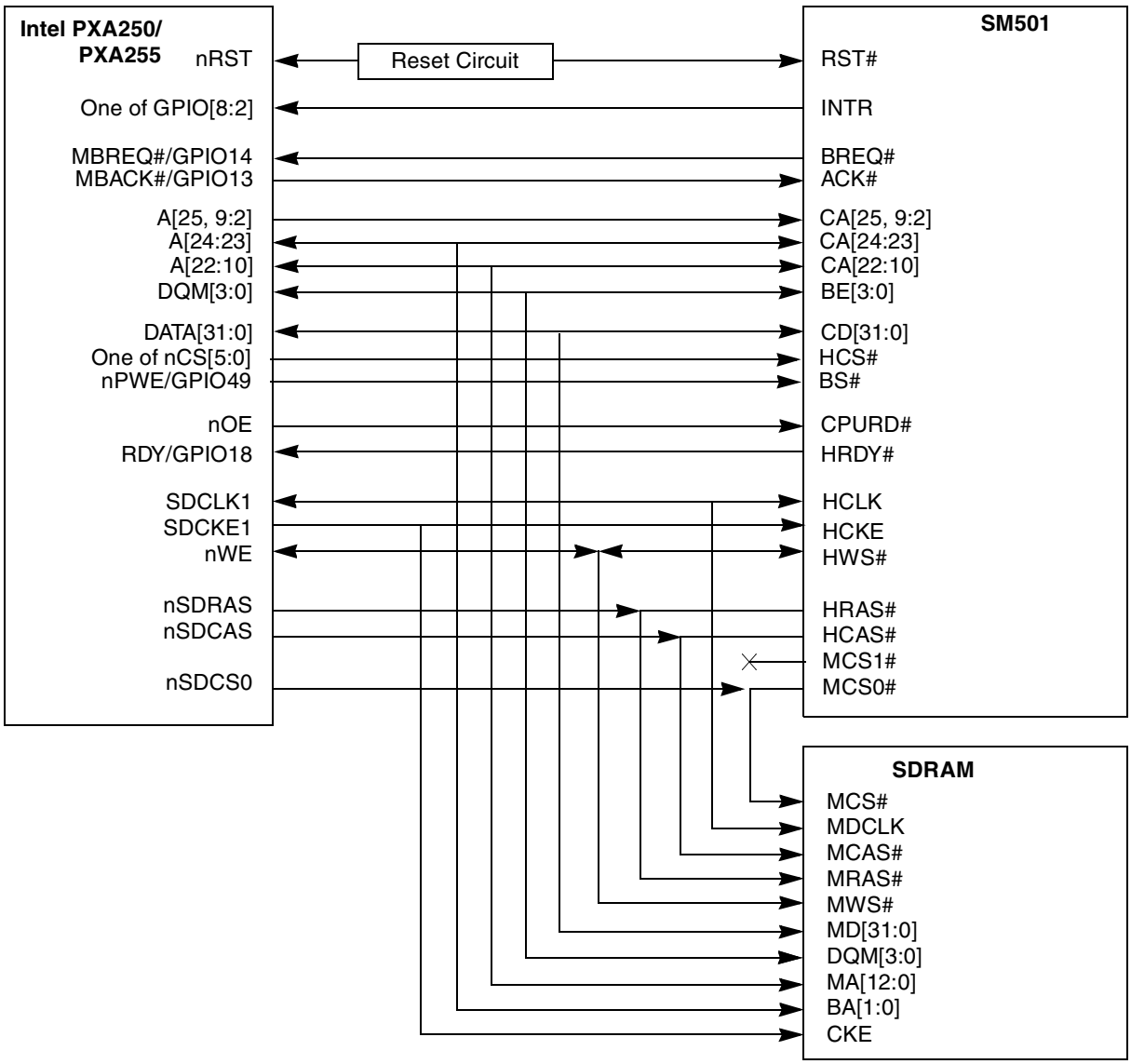
XScale

When the SM501 is interfaced to the XScale host bus, it can run in two different host interface modes:

1. In Master mode: SDRAM mode.
2. In Slave mode: SRAM-like Variable Latency I/O mode.

Figure 1-6 shows a typical system-level hookup between the SM501 device and the PXA250 or PXA255 processor.

Figure 1-6: Intel XScale (PXA250/255) to the SM501 Bus Interface



Design Notes:

1. In Bus Master mode, the SM501 drives the clock to SDRAM, and the CPU must 3-state the clock line. In Slave mode, the CPU drives the clock to both the SM501 and SDRAM.
2. The XScale processor only has one 3-stateable SDRAM CS line. Thus the SM501 can support only one group of SDRAMs in Bus Master mode.
3. According to the XScale specification, SDRAM must use SDCLK2 or SDCLK1. The variable latency I/O must use either SDCLK0 (synchronous) or no clock. Because the SM501 has to drive the SDRAM clock on the same line, SDCLK1 is used.
4. According to the XScale specification, the variable latency I/O device can use one of nCS[5:0] as the chip select.
5. The XScale processor does have a dedicated interrupt input. Use one of GPIO[8:2] and configure the selection as an interrupt via software.

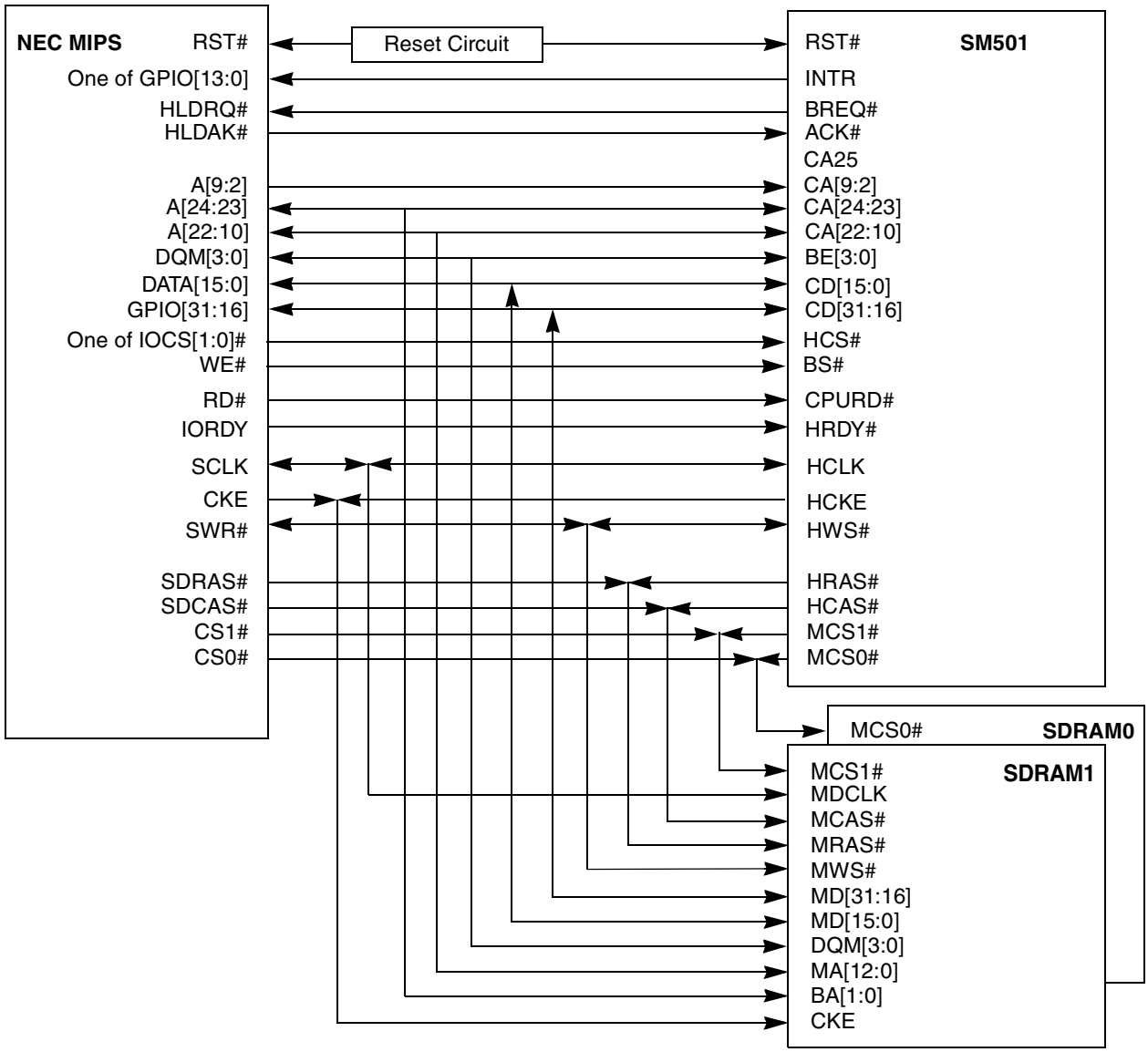
NEC V_R4122/4131 MIPS

When the SM501 is interfaced to the NEC V_R4122/31 MIPS host bus, it can run in two different host interface modes:

1. In Master mode: SDRAM mode.
2. In Slave mode: LCD Controller mode.

Figure 1-7 shows a typical system-level hookup between the SM501 device and the NEC V_R4122/31 MIPS processor.

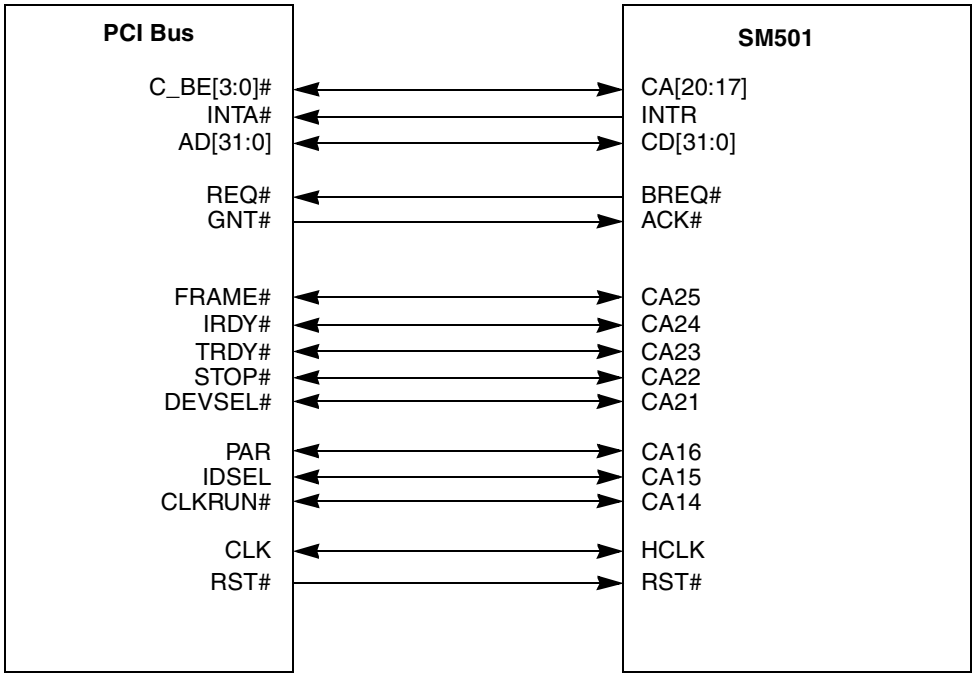
Figure 1-7: NEC MIPS to the SM501 Bus Interface



PCI Bus

Figure 1-8 shows a typical system-level hookup between the SM501 device and the PCI bus.

Figure 1-8: PCI Bus Interconnection



Command Interpreter / Command List Processor

This module is provided as an aid to the graphics controller and is used to move data from system memory to the graphics engine, executing a "Command List" placed in the shared memory. The Command Interpreter is capable of making decisions (such as a branch to another part of memory) and wait for events or conditions inside other modules.

Zoom Video Port

The SM501 includes a Zoom Video (ZV) port to allow the use of external MPEG decoders for DVD playback, external TV tuners, and other sources. This interface supports the YUV 4:2:2, YUV 4:2:2 with byte swap and RGB 5:6:5 data formats.

The standard ZV port uses an 8-bit interface at 50MHz. However, if desired, an extra 8 bits in the GPIO interface may be used to interface to ICs that only support a 16-bit ZV interface. ZV inputs up to 65 MHz are acceptable. The pins used for the ZV interface are designated GPIO pins. See the *SM501 MMCC Design Guide* for more information.

Note that the ZV input to video display path is as follows: ZV port → capture → frame buffer → video scalar → display. The capture portion supports a 1:1 or 2:1 reducing. In addition, the video scalar will allow arbitrary scaling from 1:1 to 4:1 on shrinking and 1:1 to $1:2^{12}$ on expansion; however, the quality of the expansion will be degraded beyond 1:8. This will easily allow 4:3 and 16:9 conversion, full screen PAL, and picture-in-picture.

See Chapter 10 for more information about the ZV Port registers. See the *SM501 MMCC Design Guide* for more information about interfacing the SM501 to external sources.

2D Engine

The SM501 provides industry-leading 2D acceleration through the combination of an optimized 128-bit 2D drawing engine and a high bandwidth link to local frame memory. The 2D engine also contains a command interpreter (an enhanced DMA engine) that can intelligently fetch operands out of the frame buffer at up to 600MB/s. The command interpreter can conditionally branch to another location in memory, wait for status from another module, etc. as it fetches and interprets commands.

The 2D drawing engine also contains a color space conversion unit. The color space conversion unit allows for direct translation from many YUV formats into RGB. The 2D drawing engine also contains a bi-linear scalar, which supports 4:1 shrink and $1:2^{16}$ stretch.

As noted previously, the SM501 supports frame memory in UMA, and local 32-bit modes. With 32 bits of SDRAM memory running at 150 MHz, the SM501's DMA engine has 600MB/s of memory bandwidth to use for fetching 2D operands and data. (The UMA solution's performance is dependent on the host system's topology.) This high memory bandwidth allows the 2D engine to run at full speed without costly waits or pipeline stalls from the frame buffer.

The 2D drawing engine understands the following commands:

1. BitBlt (from system/local memory to system/local memory) with 256 raster operations. Pattern is selectable between 8x8 monochrome pattern, 8x8 color pattern or another surface located in either system or local memory.
2. Transparent BitBlt with the same capabilities as the previous command, but only the source or destination can be transparent (either ColorKey or ChromaKey).
3. Alpha BitBlt with a constant alpha value.
4. Rotation BitBlt for any block size. This feature allows high speeds conversion between landscape and portrait display without the need for special software drivers. (90°, 180°, 270°.)
5. YUV to 16-bit/32-bit RGB Blt conversion with $1:2^{16}$ stretch or 4:1 shrink to provide high speeds video in common format.
6. Auto-wrapping for smooth scrolling support for navigational or other data.
7. Support for tiled memory to optimize performance for 2D operations and rotation.

As noted previously, the 2D Drawing Engine has a 150MHz clock and a 128-bit wide memory access path. With 8-bpp colors, the 2D engine can process 2400M pixels/s, and with 16-bpp the 2D engine can process 1200M pixels/s.

See Chapter 4 for more information about the 2D Drawing Engine.

Performance

By using a UMA architecture, the number of operations required by the graphics driver is less because there is no need to transfer data between host bitmaps and device bitmaps – they are the same in a UMA architecture.

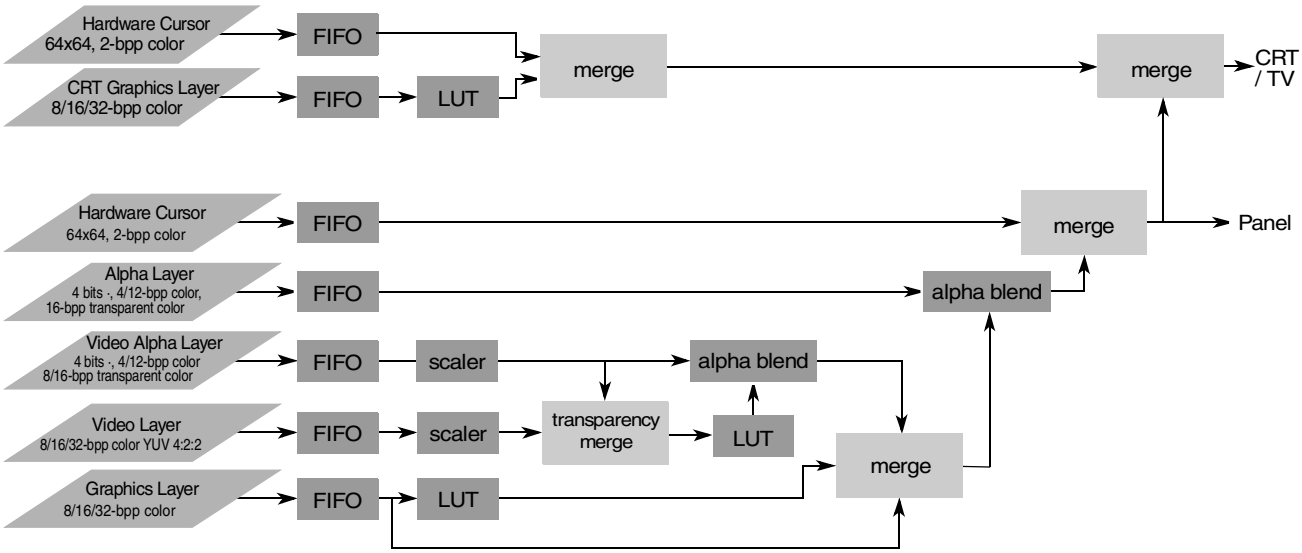
Also, both the Command Interpreter and the Color Space Conversion with Scaling will reduce the CPU utilization considerably – in the order of a 50% reduction. The reason behind this reduction is that the CPU does not have to convert the YUV color space into a RGB color space and the CPU does not have to wait until the drawing engine is finished.

Also, a sophisticated Command Interpreter algorithm can reduce the number of operations by looking into the command list for instructions still to be scheduled and combine certain instructions into one instruction.

Video Display Layers

As shown in Figure 1-9, the SM501 supports seven layers of display frames (2x hardware cursor, primary graphics, video, video alpha, alpha, and secondary graphics). (See Chapter 5 for more information about the Display Controller.)

Figure 1-9: Video Layers and Data Processing



Layer #7: Secondary Hardware Cursor

To display a cursor on the analog output for multi-monitor function.

- One single 64x64 pixel cursor 2-bpp color [1:0] (00=transparent, 01=color0, 10=color1, 11=color2), is mapped into a 32-bit RGB 8:8:8 color map.

Layer #6: Secondary Graphics

To display text or drawings on the analog output (CRT) for multi-monitor function.

- 8-bpp (index into RGB 8:8:8 lookup table).
- 16-bpp RGB 5:6:5.
- 32-bpp RGB 8:8:8.

Layer #5: Primary Hardware Cursor

To display a cursor on the digital output.

- One single 64x64 pixel cursor 2-bpp color [1:0] (00=transparent, 01=color0, 10=color1, 11=color2), is mapped into a 32-bit RGB 8:8:8 color map.

Layer #4: Alpha

To alpha-blend and/or color-key an image on top of the Primary Graphics and Video layer outputs.

- 16-bpp (4-bit alpha, 4-bit Red, 4-bit Green, 4-bit Blue) or RGB 5:6:5.
- 16-bit transparency register (with 16-bit RGB 5:6:5 mode); if a color matches the register's value it is transparent.

- 4-bit planar blending register (in 16-bit RGB 5:6:5 mode only) to blend all pixels on the plane (that are non-transparent) to one planar alpha value.

Layer #3: Video Alpha

To alpha-blend and/or color-key an image on top of the Video layer output.

- Supports bi-linear scale up or down.
- 8-bpp (4-bit alpha, 4-bit index color), or 8-bit index (index into RGB 8:8:8 lookup table).
- 16-bpp (4-bit alpha, 4-bit Red, 4-bit Green, 4-bit Blue) or RGB 5:6:5.
- 8-bit transparency register (with 8-bit index), or 16-bit transparency register (with 16-bit RGB 5:6:5 mode); if a color matches the register's value it is transparent.
- 4-bit planar blending register (in 16-bit RGB 5:6:5 mode only) to blend all pixels on the plane (that are non-transparent) to one planar alpha value.

Layer #2: Video

To overlay video image or graphics on top of Primary Graphics layer.

- Supports bi-linear scale up or down.
- 8-bit index (index into RGB 8:8:8 lookup table).
- 16-bpp RGB 5:6:5 or YUV 4:2:2.
- 32-bpp RGB 8:8:8.

Layer #1: Primary Graphics

To display text or drawings on the primary output (LCD panel).

- Support smooth scrolling and auto-wrapping.
- 8-bit index (index into RGB 8:8:8 lookup table).
- 16-bpp RGB 5:6:5.
- 32-bpp RGB 8:8:8.
- 8-bit (with 8-bit index), 16-bit (with 16-bit RGB 5:6:5), or 32-bit (with 32-bit RGB 8:8:8) color key register. If the color value matches the register's value, the color is transparent and the pixel from the Primary Video layer will be shown instead.

Terminology

Some definitions of terms used above:

bpp – bits per pixel.

RGB – Red, Green, Blue.

RGB 5:6:5 – 16-bit color mode, where Red has 5 bits, Green has 6 bits, and Blue has 5 bits.

RGB 8:8:8 – 32-bit color mode, where Red has 8 bits, Green has 8 bits, and Blue has 8 bits. The upper 8 bits are unused.

Alpha – A means of blending two layers together. The fundamental equation is (α * c) + (1 - α) * (original pixel), where c is the 4-bit color that points into the 18-bit lookup table. In practical terms, 4-bit alpha blending means that for two given display layers, one layer is dominant (e.g. video layer) and the other layer (e.g. video alpha) has 16 levels of transparency from 100% to 0% that may be applied to the colors in that layer.

LUT – Lookup table; a means to map an 8-bit color index into a 24-bit color space, thus a smaller number of simultaneous colors (8-bit) but with a wide range of colors (24-bit) to choose from.

Display Resolution

The SM501 supports display resolutions up to 1280 x 1024. 16:9 formats in this range (e.g. 800 x 480, 1024 x 600, and 1280 x 768) are supported. Note that there are trade-offs between the maximum resolution, the number of active video layers, and the frame memory choice.

Dual Display

The SM501 supports Dual Display, i.e. two different displays of the same or different resolutions.

As shown in Figure 1-9, only the panel pipe supports the video and alpha planes. However, since these planes fetch data from the frame buffer memory as well, there might not be enough bandwidth to enable the video and alpha planes in Dual Display mode.

In order to display video on the panel pipe in Dual Display mode and on the CRT pipe in any mode, the 2D Engine's Color Space Conversion and Stretching functionality should be used.

LCD Panel

The SM501's LCD logic block will drive an 18-bit or 24-bit TFT panel directly. 12-bit CSTN panels are also supported, and a dithering engine will support them to an effective 18-bit resolution. The maximum supported panel size is 1280 x 1024. Panel power sequencing is also provided through either hardware or software control.

Figure 1-10 shows a typical interface between the SM501 and a 24-bit TFT panel. Note the following with regards to this interface:

1. The timings of VDEN, FPEN, and BIAS are fully controlled by software.
2. The TFT panel does not use Vbias. The BIAS control used here controls the On/Off switch of the backlights. Program its timing so backlight is on after 12V is applied to the inverter.
3. The SM501 provides three PWM signals that can be used to control brightness.
4. To support a 24-bit TFT panel, use pins GPIO[63:58]. The pins are then limited to use as digital 8-bit TV data out and 8- or 16-bit video capture.

See the *SM501 MMCC Design Guide* for information about interfacing to LCD displays.

Figure 1-10: Typical 24-bit TFT Panel Interface

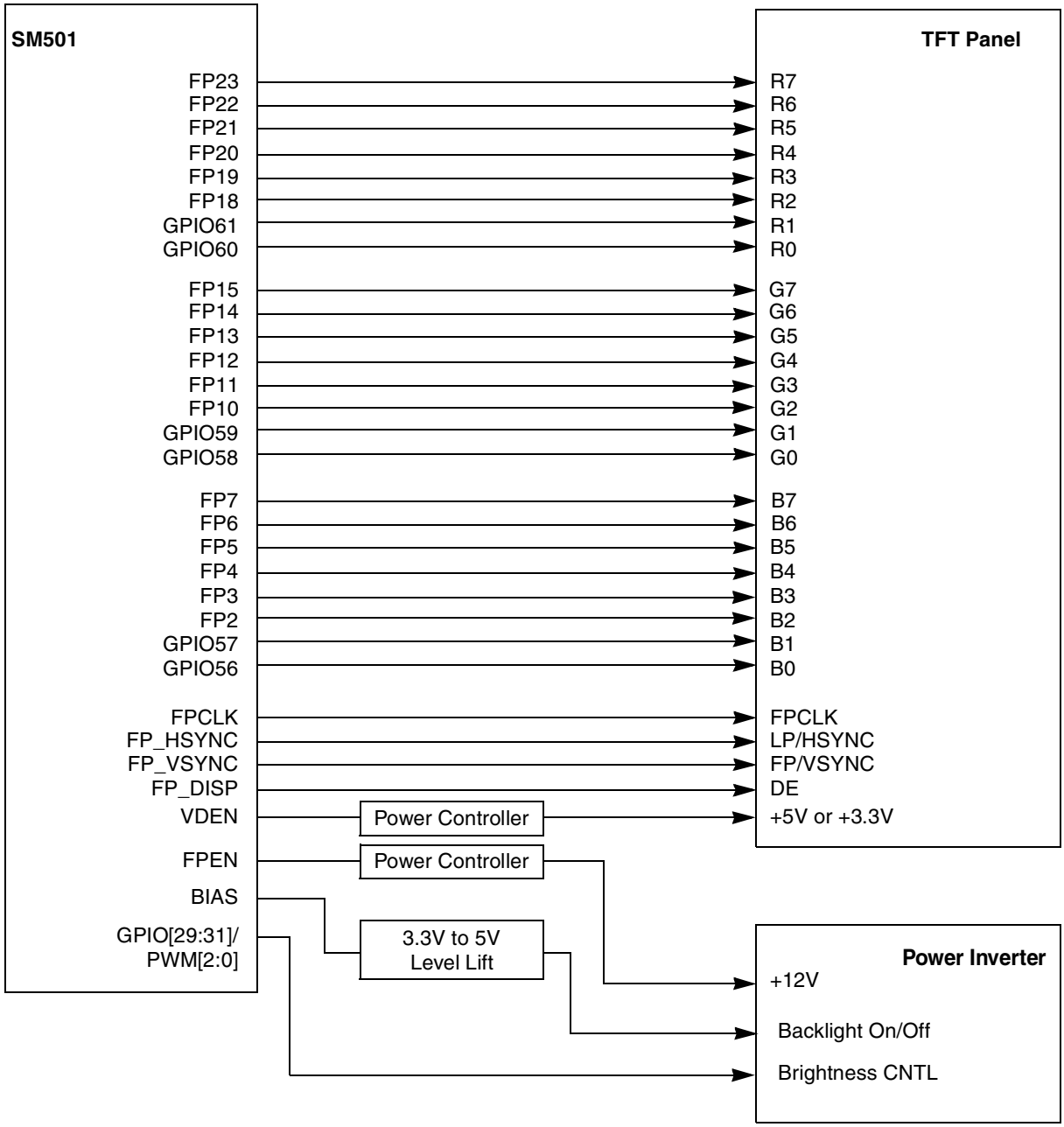


Figure 1-11 shows a typical interface between the SM501 and an 18-bit TFT panel.

Figure 1-11: Typical 18-bit TFT Panel Interface

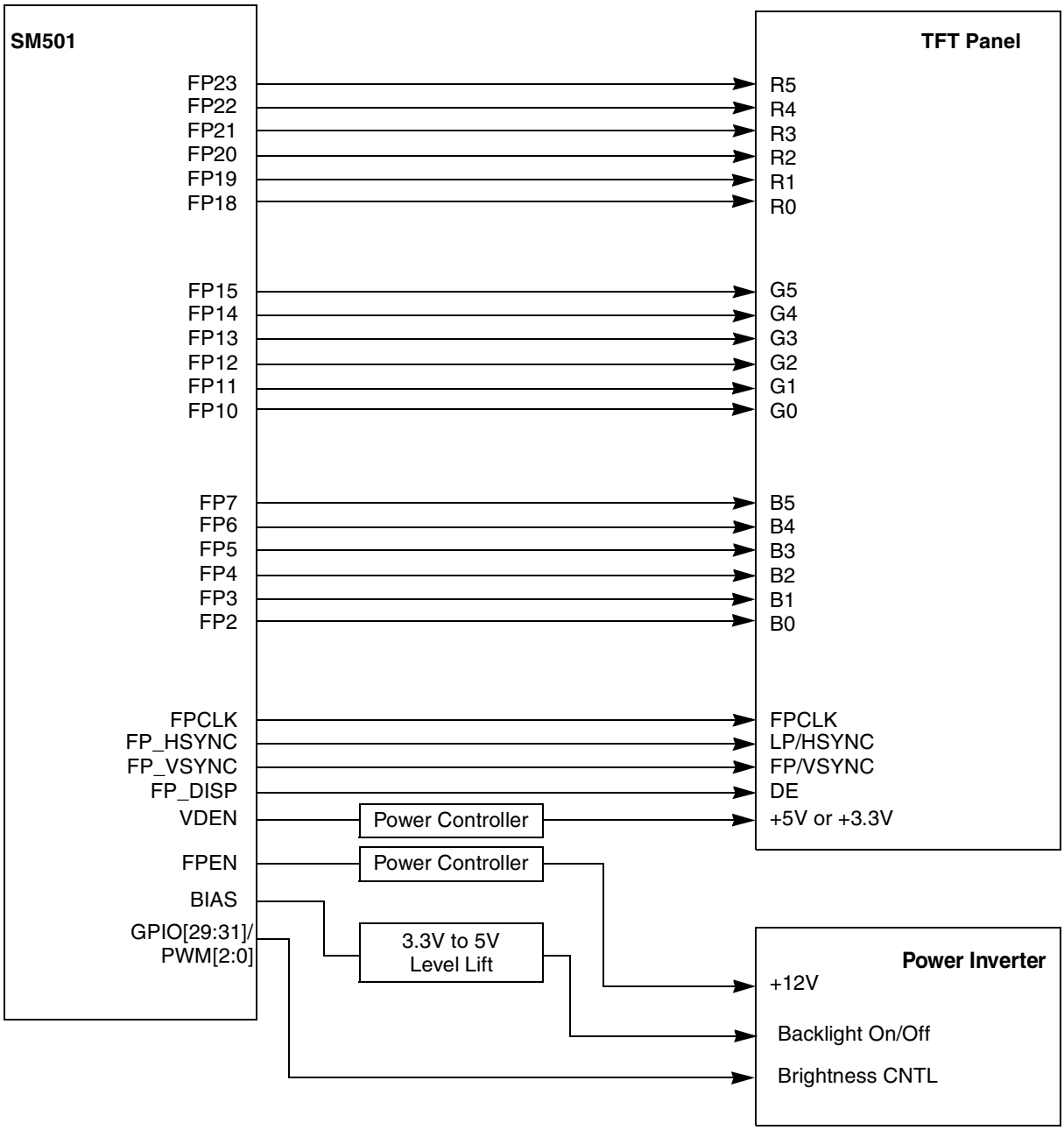


Figure 1-12 shows a typical interface between the SM501 and a 24-bit LVDS transmitter.

Figure 1-12: Typical 24-bit LVDS Interface (DS90C385)

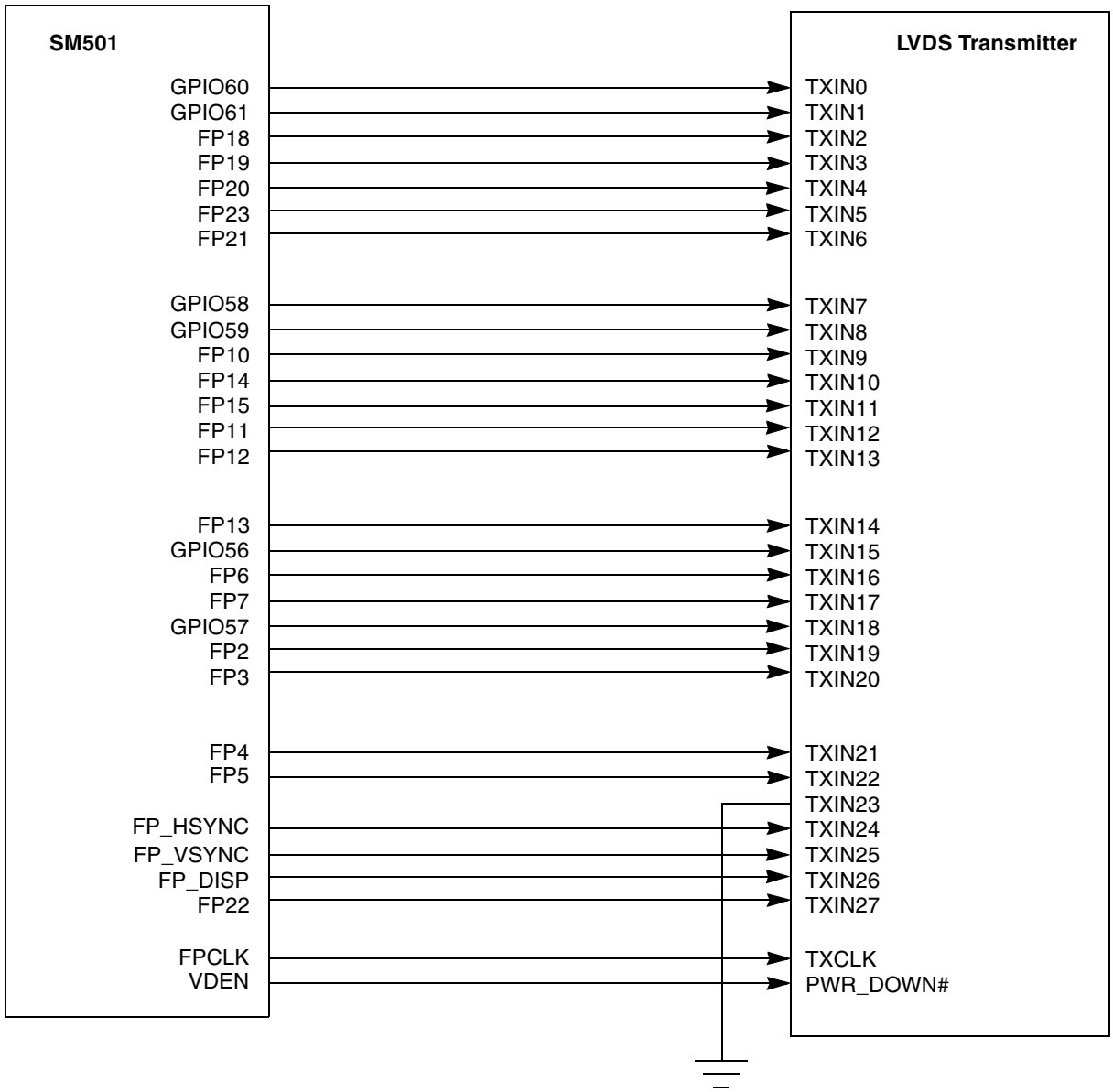


Table 1-2 compares the display data between the TFT panel, LVDS, and digital out.

Table 1-2: SM501 Display Data

	TFT Panel		LVDS	Digital Out[1]		CTSN (8 bit)			CTSN (12 bit)
	24-bit TFT[2]	18-bit TFT	DS90C385	1st Clock	2nd Clock	1st Clock	2nd Clock	3rd Clock	
GPIO63				G2	R4				
GPIO62				G1	R3				
FP23	R7	R5	TXIN5						R0
FP22	R6	R4	TXIN27						G0
FP21	R5	R3	TXIN6						B0
FP20	R4	R2	TXIN4						R1
FP19	R3	R1	TXIN3			R0	B2	G5	G1
FP18	R2	R0	TXIN2			G0	R3	B5	B1
GPIO61	R1		TXIN1	G0	R2				
GPIO60	R0		TXIN0	B4	R1				
FP15	G7	G5	TXIN11			B0	G3	R6	R2
FP14	G6	G4	TXIN10			R1	B3	G6	G2
FP13	G5	G3	TXIN14			G1	R4	B6	B2
FP12	G4	G2	TXIN13			B1	G4	R7	R3
FP11	G3	G1	TXIN12			R2	B4	G7	G3
FP10	G2	G0	TXIN9			G2	R5	B7	B3
GPIO59	G1		TXIN8	B3	R0				
GPIO58	G0		TXIN7	B2	G5				
FP7	B7	B5	TXIN17						
FP6	B6	B4	TXIN16						
FP5	B5	B3	TXIN22						
FP4	B4	B2	TXIN21						
FP3	B3	B1	TXIN20						
FP2	B2	B0	TXIN19						
GPIO57	B1		TXIN18	B1	G4				
GPIO56	B0		TXIN15	B0	G3				
GPIO55				2X Pixel Clock					

1. For Digital Out, program bit 25 of the Miscellaneous Control Register at offset 0x4 to 0 and bits [31:23] of the GPIO[63:32] Control Register at offset 0xC to 0x1FF.

2. For 24-bit TFT, program bit 25 of the Miscellaneous Control Register at offset 0x4 to 1 and bits [29:24] of the GPIO[63:32] Control Register at offset 0xC to 0x3F.

Analog RGB (Analog LCD or CRT)

The analog RGB block contains a 24-bit DAC (RGB 8:8:8) to drive an external analog RGB interface. The 200MHz DAC will easily support the maximum resolution of 1280 x 1024. See the *SM501 MMCC Design Guide* for information about interfacing the SM501 to analog RGB devices.

Internal or External Memory

The SM501's advanced 2D and video capabilities, when combined with a large fully-functioned display can require a high memory bandwidth. However, this higher graphics performance comes at a higher cost. Each system designer must carefully trade-off considerations of cost, performance, display resolution, number of display planes, etc. to arrive at the optimal system design for their application. To work well in a broad variety of applications, the SM501 supports the use of two different memory interfaces.

For lower cost applications willing to accept the trade-offs of lower performance, lower display resolution and/or fewer display planes, the SM501 supports a Unified Memory Architecture (UMA) model. In the UMA model, the host processor has no local memory. Therefore all host memory is shared with frame buffer memory with the commensurate performance impacts. However, for systems willing to accept lower resolution panels, the UMA design can provide for a very low cost solution.

For higher performance applications, the SM501 supports the use of an external 32-bit frame buffer. When using a 32-bit frame buffer, the SM501 uses a 2-64 MB external memory interface that will work with SDRAM. A data bus width of 32-bit provides a memory bandwidth of 600MB/s (SDRAM @ 150MHz). In this case, the SM501 supports a 32-bit interface to the host processor (either via PCI or the host memory interface).

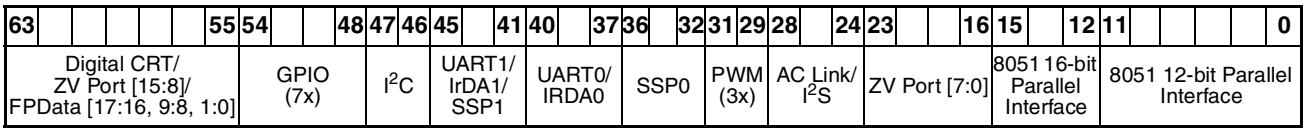
For designs using frame buffers, the SM501 also allows the use of 16MB to 64MB SDRAM internal to the SM501 MCB package. This allows minimal footprint impact to the system design as well as minimizing the layout and electrical constraints associated with using external SDRAM memory.

See the *SM501 MMCC Design Guide* for information about interfacing the SM501 to local memory.

GPIO

The SM501 provides 64 bits of GPIO. 57 of these bits are multiplexed and may be used to support special features or used to support standard GPIO under software control. Bits 48-54 of GPIO can be programmed as an interrupt input from external devices. Figure 1-13 shows the layout of the GPIO bits.

Figure 1-13: GPIO Layout



The 24-bit TFT panel interface, 16-bit ZV port interface and 8-bit Digital CRT interface are multiplexed. Three configurations can be supported with GPIO[63:56]:

- Configuration 1: 24-bit TFT panel interface and 8-bit ZV port interface
- Configuration 2: 18-bit TFT panel interface and 16-bit ZV port interface
- 18-bit TFT panel interface, 8-bit ZV port interface and 8-bit Digital CRT interface

Table 1-3 shows the function summary for pins GPIO[63:56].

Table 1-3: Function Summary, GPIO[63:56]

GPIO	ZV Port	Digital CRT	FP Data
55		CLOCK	
56	ZV8	DCRT0	FP0
57	ZV9	DCRT1	FP1
58	ZV10	DCRT2	FP8
59	ZV11	DCRT3	FP9
60	ZV12	DCRT4	FP16
61	ZV13	DCRT5	FP17
62	ZV14	DCRT6	
63	ZV15	DCRT7	

The following sections define the special features of the GPIO.

8051 µ-Controller

The SM501's 8051 µ-controller can be programmed to use its 8-bit parallel interface with control signals to follow any protocol (GPIO bits 0-15). The maximum speed of the 8051 µ-controller is 80 MHz.

ZV Port

Most ZV-compatible ICs support an 8-bit wide ZV port. The SM501 includes this functionality using GPIO pins 16-23. However, some ICs may only support a 16-bit ZV interface. To support these ICs, GPIO bits 56-63 may be used to add the extra 8 bits to the ZV interface.

Note: The upper 8 bits (bits 56-63) are multiplexed with the Digital CRT and Flat Panel Data [17:16,9:8,1:0].

AC Link / I^2S

The SM501 provides an AC Link interface on GPIO bits 24-28. This is a standard 48kHz AC Link interface that interfaces to an AC97 CODEC.

If a higher quality audio solution is required, the SM501 includes an I^2S interface that can be used instead of the AC97 interface. The GPIO bits are multiplexed with the AC Link interface.

For enhanced audio playback using DMA, the 8051 µ-controller can be used. The audio data is transferred into the 8051 µ-controller SRAM using the DMA engine, and the 8051 µ-controller will copy the data to the AC Link interface one per clock.

PWM Interface

Three independent PWM outputs (bits 29-31 of GPIO) are provided for generic use. Each output has its own control register. Two PWMs have each three independently selectable frequencies. The third PWM has three selectable frequency multiples that are synchronized to the video SYNC signal.

SSP Interface

The SM501 provides 2x5 special bits (bits 32-36 and 41-45 of GPIO) to act as an SPI-like interface to support external ICs such as CAN controllers.

Note: Bits 41-45 (SSP1) are multiplexed with UART1 bits.

UART Interface

Two generic 16750 compatible serial ports (bits 37-45 of GPIO) are provided. Both UARTs include RD, TD, CTS, and RTS.

Note: Bits 37-45 (UART[0-1]) are multiplexed with the SSP1 and IrDA bits.

IrDA

The IrDA controller (bits 37-45 of GPIO) is standalone and supports data transfer of up to 115kb/s. It adheres to the SIR protocol.

Note: These bits are multiplexed with UART[0-1] and SSP1.

I^2C Interface

The SM501 supports one I^2C interface (bits 46-47 of GPIO). The interface is master mode with 7-bit addressing. Speeds up to 400kb/s (Fast mode) are supported.

Digital TV Encoder Interface

The SM501 supports an 8-bit digital RGB output (bits 55-63 of GPIO) to hook up to external TV encoders.

Note: These bits are multiplexed with ZV Port [15:8] and Flat Panel Data [17:16,9:8,1:0].

Strap Pins

GPIO pins 0 through 7 and 12 through 15 control the power-on configuration for the SM501 according to Table 1-4.

Table 1-4: GPIO Strap Pins

Pin	Default Strapping	Description
GPIO0	Pulled-down	Bus selection: 0: Host Bus. 1: PCI.
GPIO[2:1]	Pulled-down	Host Bus selection: 00: Hitachi SH3/SH4 host bus. 01: Intel XScale PXA250/PXA255 host bus. 11: NEC MIPS V_R4122/V_R4131 host bus.
GPIO3	Pulled-down	Clock select: 0: Internal PLL. 1: External test clock.
GPIO4	Pulled-down	Ready polarity for Hitachi SH series CPU: 0: Active low. 1: Active high.
GPIO[6:5]	Pulled-down	Local memory column size: 0x: 256 words. 10: 512 words. 11: 1024 words.

Table 1-4: GPIO Strap Pins

Pin	Default Strapping	Description
GPIO7	Pulled-down	Clock divider reset control: 0: Do not reset clock divider. 1: Reset clock divider.
GPIO12	Pulled-down	Chip select pins: 0: Drive MCS# pins. 1: Do not drive MCS# pins
GPIO[15:13]	Pulled-down	Local memory size select: 000: 4MB. 001: 8MB. 010: 16MB. 011: 32MB. 100: 64MB. 101: 2MB.
GPIO31, GPIO29	Pulled-down	XScale Host Clock Input Source 00: From internal clock generated by internal PLL 01: From HCLK pin. 1x: From GPIO30 pin.

Flat Panel Data

The SM501 supports a native 18-bit panel interface, which can be extended to 24-bit by using GPIO bits 55-60. See the section entitled “Video Layers and Data Processing” on page 12 for a detailed list of how the 24-bit panel interface is connected.

Note: These bits are multiplexed with Digital TV Encoder and ZV Port [15:0].

USB Controllers

The SM501 supports one USB 1.1 compliant port. It is shared between the host controller and the device controller (ownership is software programmable).

Host Controller: OHCI compliant USB host controller. This module is sourced from the leading supplier of USB cores and it is compatible with existing software drivers shipping on all modern Operating Systems. The controller has its own DMA engine and bus arbitration mechanisms for increased performance and ease of use.

Device Controller: USB device controller supports 3 endpoints. This Generic Device implementation can be fully configurable through a base driver/application running the host CPU. The main function for the device controller is connecting to a PC for file transfers/downloads.

DMA Controller

The SM501 supports a 2-channel DMA controller than can move data between 8051 SRAM and either memory bus or it can move data from one memory bus to another memory bus.

The two channels of the DMA controller are dedicated:

1. Channel 1 is used to transfer data between both memory busses and the 8051 µ-controller data SRAM.
2. Channel 2 is used to transfer data between or within memory buses.

Interrupt Controller

The SM501 has only one interrupt to the host bus. This means all internal interrupts are shared without priority.

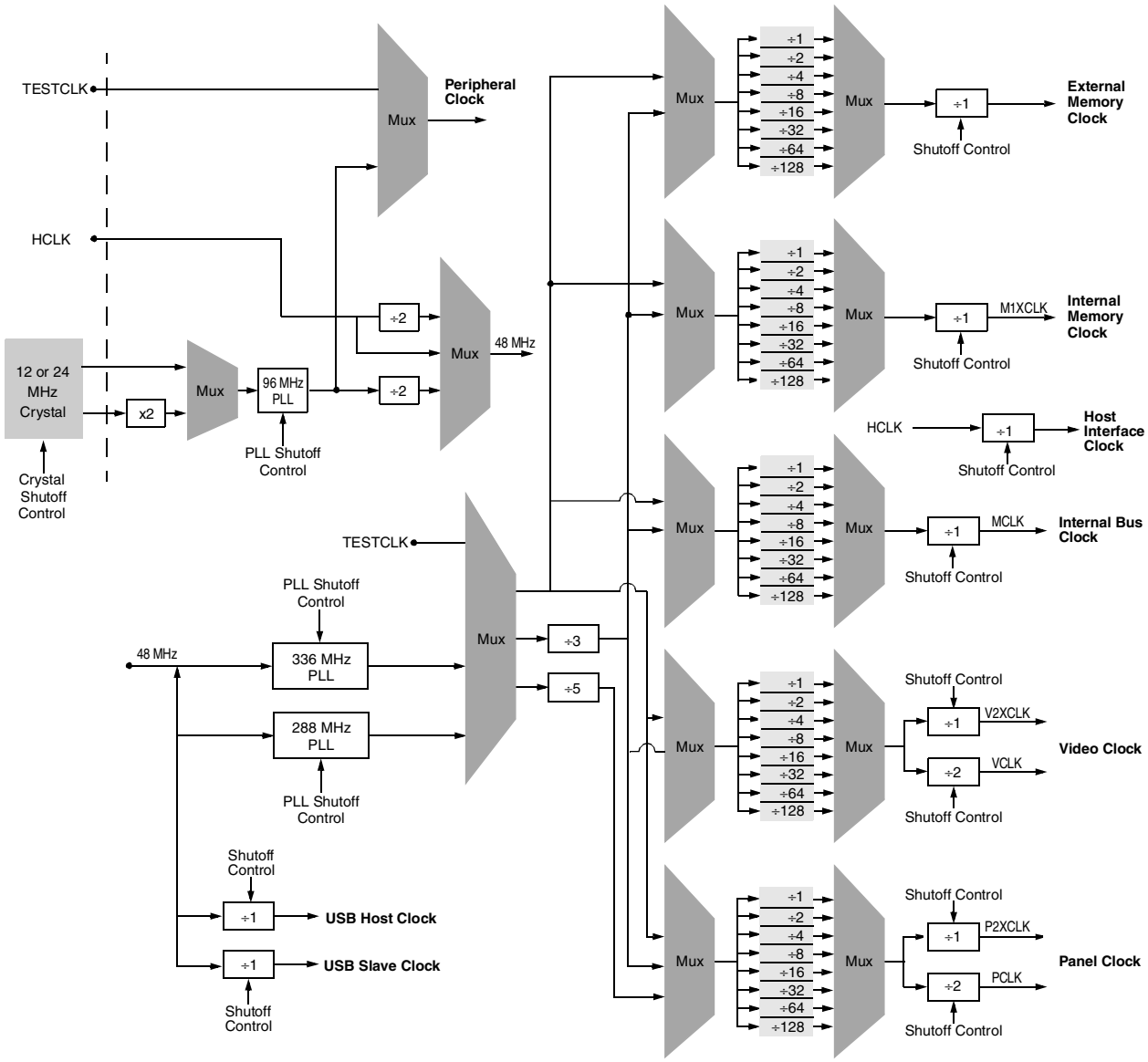
One interrupt status register specifies which module(s) generated the interrupts and the software drivers are responsible for clearing the interrupt at the source. The Host Interrupt remains asserted as long as there is any bit set in the System Interrupt Status register.

For example, if the USB Host generated an interrupt, then the USB Host Interrupt Pending bit is set in the System Interrupt Status register. If the USB Host Interrupt Mask bit is set in the System Interrupt Mask register, then the SM501 asserts the Host Interrupt line. The operating system finally redirects the interrupt routine to the USB Host driver. The USB Host Driver reads the System Interrupt Status register and determines it has to service the USB Host. It then reads the USB Host Interrupt Status register, performing the requested tasks, and clears the interrupt in the USB Host register. If there are no more interrupt bits set in the System Interrupt Status register, the SM501 will deassert the Host Interrupt line.

Clock Control

The SM501 has one oscillator input and two external clock inputs. Figure 1-14 shows the clock tree for the SM501.

Figure 1-14: Clock Tree



Power Management

Figure 1-15 shows the possible power states the SM501 supports.

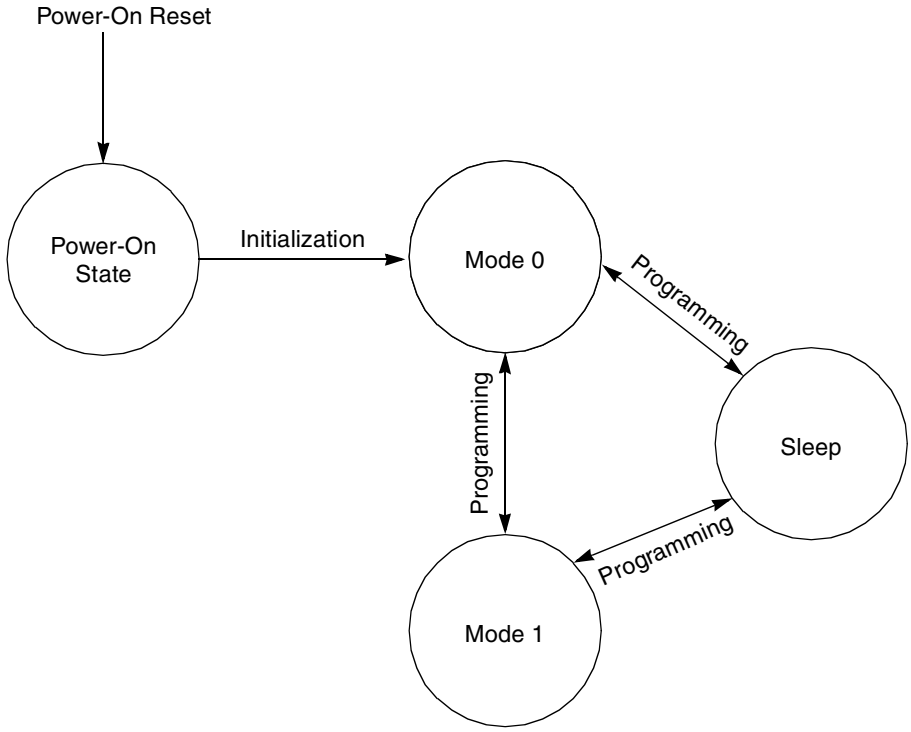
During power-on reset, the SM501 comes up in a predefined state with all I/O turned off, and running the lowest possible clock. The software is responsible for programming the Mode 0 power state to the requested state after power-on and transition into the Mode 0 power state.

The Mode 0 and Mode 1 power states are the same and fully under software control. Whenever the software decides that the SM501 must go into a different state, the software programs the non-active power state and transitions into that state. This way there are an infinite number of power states supported by software and makes power management very flexible.

The Sleep power state puts the SM501 into a sleep mode. In this mode the SDRAM is put into self-refresh mode, and the crystal and PLL circuits are turned off. The CLKOFF input pin is asserted by the system

integrator to turn off the host clock inside the SM501 in order to reduce power consumption even further if the system integrator decides to do so.

Figure 1-15: Power State Diagram

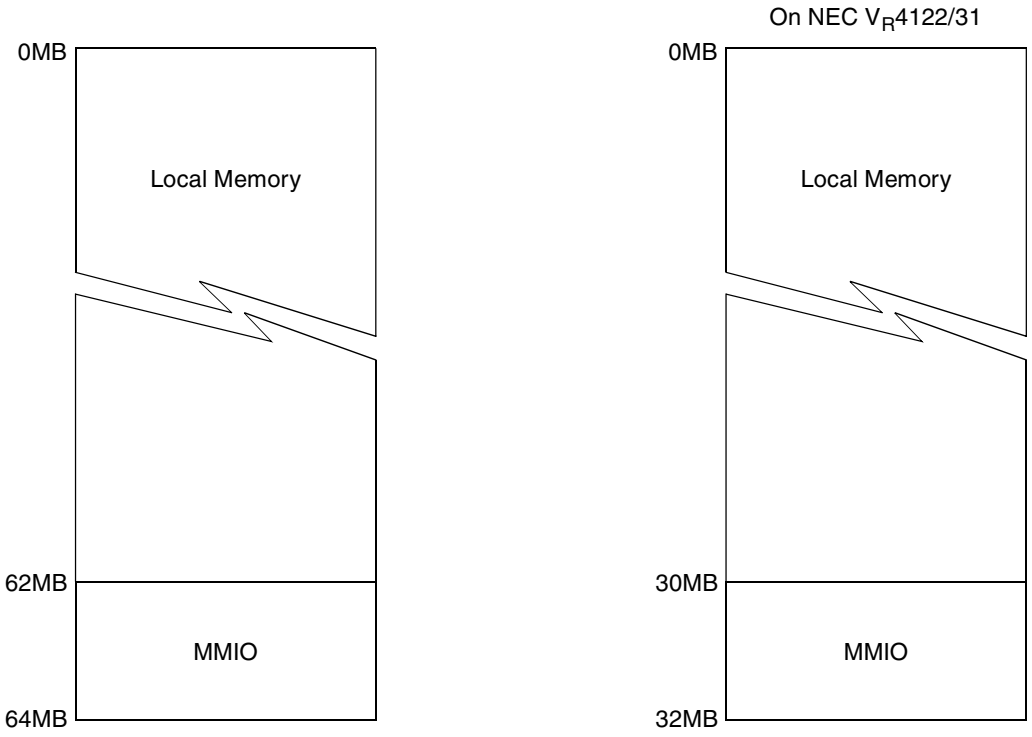


Note: The System Control registers are clocked in the Host clock domain and even shutting off the crystal and PLL circuits does not keep the System Control registers and the host bus from functioning. This way the software is always able to wake up the SM501 from sleep mode. In order to reduce even more power, the CLKOFF input pin should be used for gating the host clock.

Memory Map and Register Space

The memory map is a 64MB chunk divided into two chunks: a local memory space that includes the frame buffer and a memory mapped I/O space. The size of the local memory depends on the amount of internal or external memory attached, which may be in the range of 2MB to 64MB. MMIO space contains the SM501 register set. The Local Memory contains the actual frame buffer, 2D command lists, or any other data that can be directly accessed by tone of the SM501's functional blocks.

Figure 1-16: Memory Map

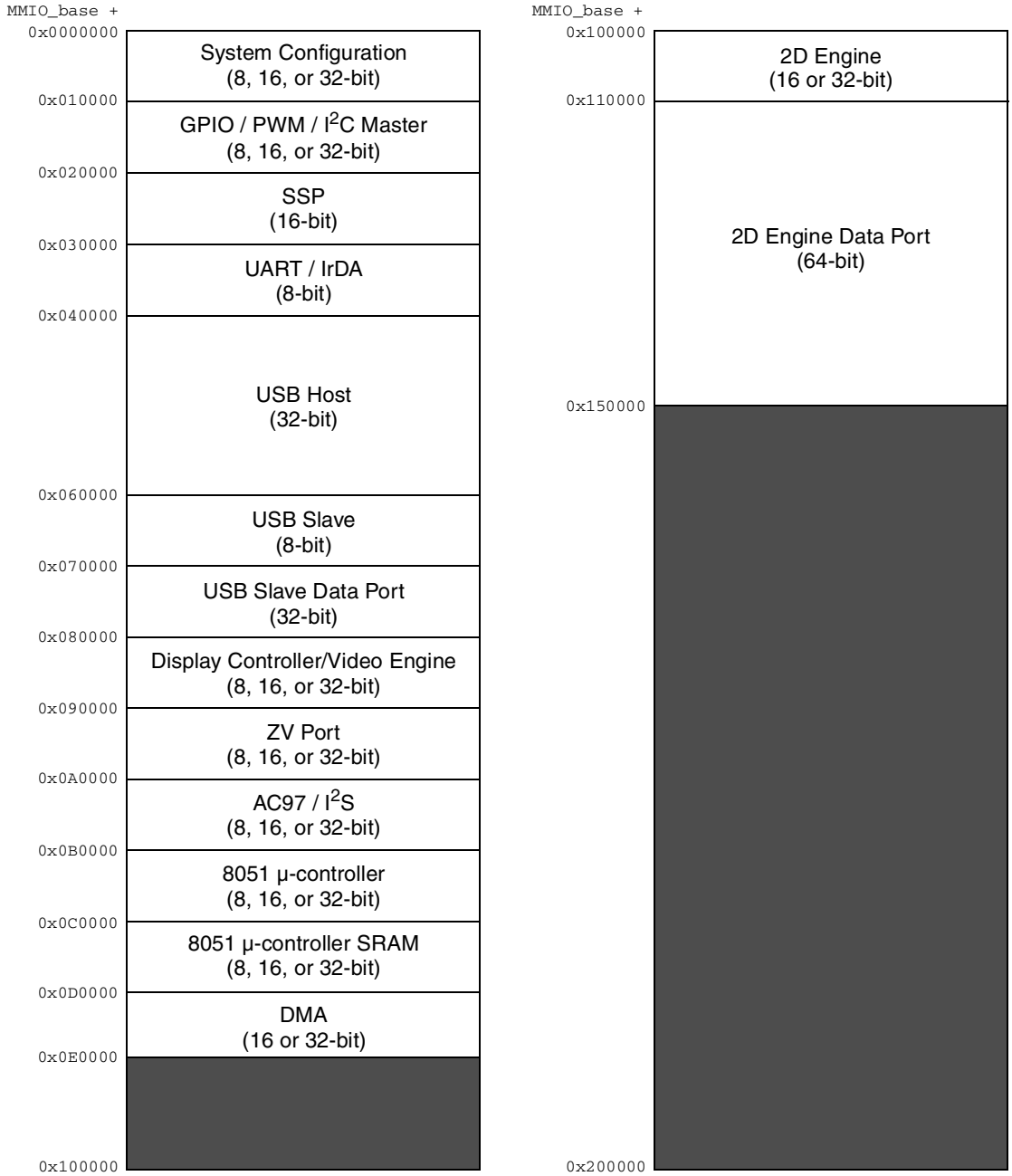


MMIO Space

The MMIO space contains the SM501 register set and is divided into separate 64kB blocks that hold the registers for each individual functional block of the SM501. If a functional block requires a data port to fill its FIFO, a separate 64kB block will be specified for the data port.

Figure 1-17 shows the MMIO space. Note that the addresses are offsets from the MMIO base address. The MMIO base address is dependent upon which host processor is being used. Refer to Table 1-5 on page 28 for the different addresses.

Figure 1-17: MMIO Space



MMIO Addressing

For different bus interfaces, the MMIO address is decoded differently. Table 1-5 lists the different MMIO addresses for all possible host interfaces. Note that for the NEC MIPS Host Interface, the MMIO address can

be programmed to be moved from 30MB (0x1E00000) to 62MB (0x3E00000) if a newer version of NEC MIPS supports 64MB I/O spaces instead of 32MB.

Table 1-5: MMIO Base Addresses for Host Interface

Host I/F	MMIO Address (MMIO_base address)	
	Power-Up	Programmable
Hitachi SH4	0x3E00000	N/A
Intel XScale	0x3E00000	N/A
NEC MIPS	0x1E00000	0x3E00000
PCI	N/A	PCI_CONFIG_14

2 System Configuration

Functional Overview

The SM501 has only one interrupt to the host bus. All internal interrupts are shared without priority.

Register Descriptions

The SM501 System Configuration Registers are located at base address `MMIO_base+0x0000000`, and contains 28 configuration registers. Every register is 32-bit DWORD aligned with offset addresses from 0x00 to 0x68. Not all of them are 32 bits wide; but if not specified, it should be reserved, and read as defaults. Most of them are software programmable, i.e. both write and read, but some of them are set by hardware and should be read only.

Figure 2-1 shows how this 64kB region in the MMIO space is laid out. It contains the System Configuration registers that are clocked from the Host Bus domain, so they are always available as long as there is a host clock, even when all other internal blocks are turned off.

Figure 2-1: System Configuration Register Space

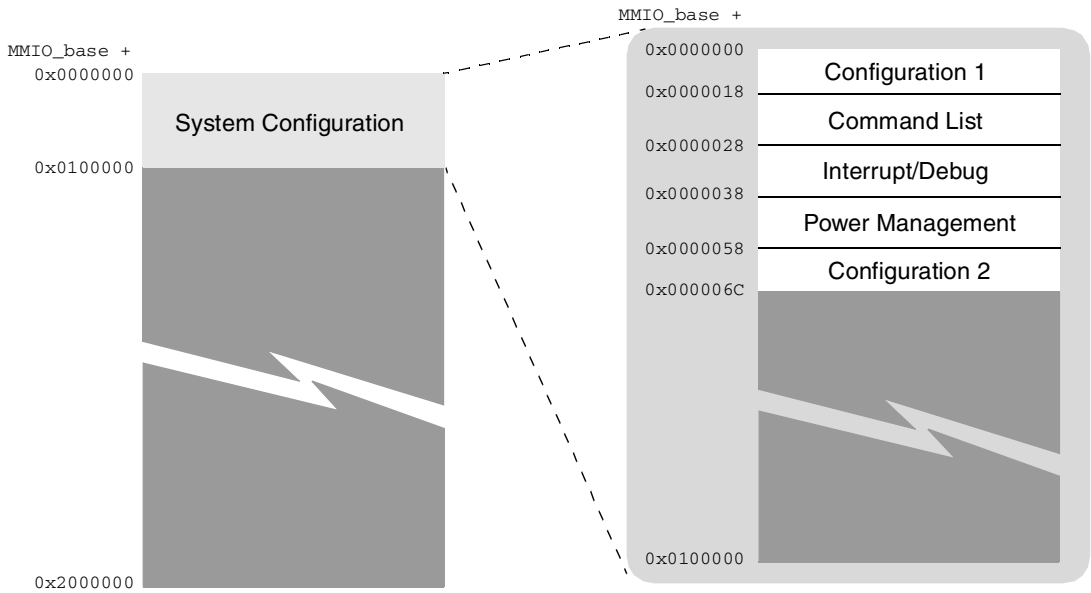


Table 2-1 shows the System Configuration Register offsets and general functions (Base Address: `MMIO_base`).

Table 2-1: SM501 System Configuration Register Summary

Address Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
Configuration 1				
0x000000	R/W	32	0b0000.0000.XX0X.X0XX. 0000.0000.0000.0000	System Control
0x000004	R/W	32	0b0000.0000.0000.00X0. 0001.0000.XXX0.0XXX	Miscellaneous Control
0x000008	R/W	32	0x00000000	$GPIO_{31:0}$ Control
0x00000C	R/W	32	0x00000000	$GPIO_{63:32}$ Control
0x000010	R/W	32	0bX000.0111.1111.0001. XXXX.X111.1100.0000	DRAM Control
0x000014	R/W	32	0x05146732	Arbitration Control
Command List				
0x000024	R	32	0000.0000.000X.XXXX. XXXX.X000.0000.0XXX	Command List Status
Interrupt/Debug				
0x000028	R	32	0x00000000	Raw Interrupt Status
0x000028	W	32	0x00000000	Raw Interrupt Clear
0x00002C	R	32	0x00000000	Interrupt Status
0x000030	R/W	32	0x00000000	Interrupt Mask
0x000034	R/W	32	0x00000000	Debug Control
Power Management				
0x000038	R	32	0x00021807	Current Gate
0x00003C	R	32	0x2A1A0A09	Current Clock
0x000040	R/W	32	0x00021807	Power Mode 0 Gate
0x000044	R/W	32	0x2A1A0A09	Power Mode 0 Clock
0x000048	R/W	32	0x00021807	Power Mode 1 Gate
0x00004C	R/W	32	0x2A1A0A09	Power Mode 1 Clock

Table 2-1: SM501 System Configuration Register Summary (Continued)

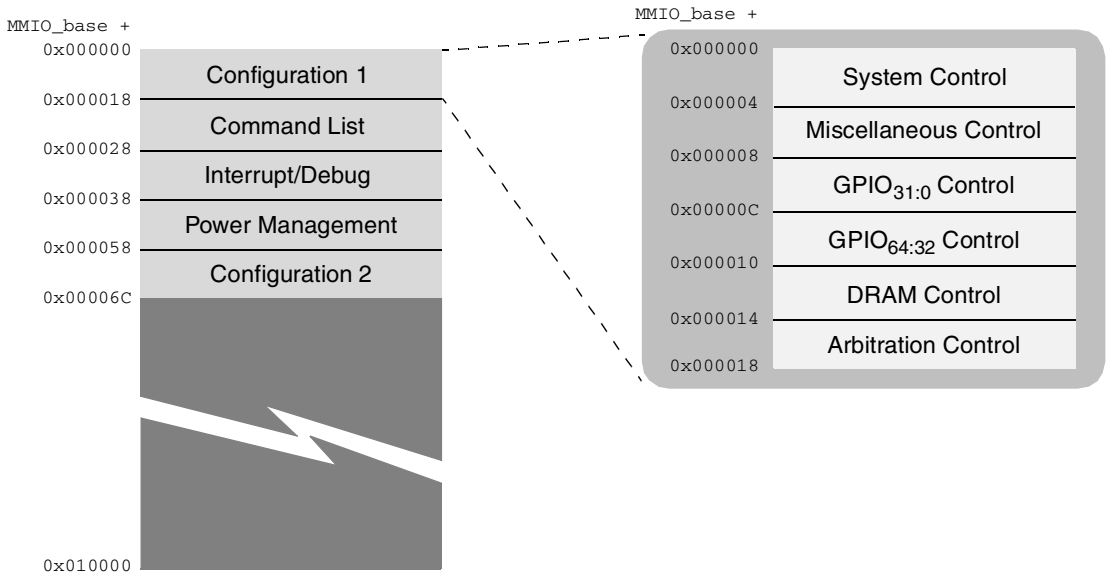
Address Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
0x000050	R/W	32	0x00018000	Sleep Mode Gate
0x000054	R/W	32	0x00000000	Power Mode Control
Configuration 2				
0x000058	R/W	32	0x00000000	PCI Master Base Address
0x00005C	R/W	32	0b0000.0000.0000.0000. 0000.0000.0000.0001	Endian Control
0x000060	R	32	0x050100A0	Device Id
0x000064	R	32	0x00000000	PLL Clock Count
0x000068	R/W	32	0x00090900	Miscellaneous Timing
0x00006C	R	32	0x00000009	Current System SDRAM Clock

1. Refer to Table 1-5 on page 1-28 for MMIO_base values depending on the CPU.
2. In the reset values, “X” indicates don’t care.

Configuration 1 Register Descriptions

The Configuration registers control the way the SM501 chips operates. Figure 2-2 shows the layout of the configuration registers in Configuration Register Space 1.

Figure 2-2: Configuration Register Space 1



System Control

Read/Write MMIO_base + 0x000000

Power-on Default 0b0000.0000.XX0X.X0XX.0000.0000.0000.0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DPMS R/W		BE R/W	CsB R	Res		BM R/W	Lat R/W	PS R	VS R	Res	2E R	2B R	Res	CS R	ZvS R
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BrE R/W	Res	Abort R/W		Lck R/W	Res			Rty R/W	Clk R/W	BrS R/W		Res	CT R/W	MT R/W	PT R/W

Bit(s)	Name	Description
31:30	DPMS	<table><tr><th>31:30</th><th>Vertical Sync</th><th>Horizontal Sync</th></tr><tr><td>00</td><td>Pulsing</td><td>Pulsing</td></tr><tr><td>01</td><td>Pulsing</td><td>Not pulsing</td></tr><tr><td>10</td><td>Not pulsing</td><td>Pulsing</td></tr><tr><td>11</td><td>Not pulsing</td><td>Not pulsing</td></tr></table>
29	BE	PCI Burst Enable. 0: Disable. 1: Enable.
28	CsB	Color Space Conversion Status. This bit is read-only. 0: Idle. 1: Busy.
27:26	Res	These bits are reserved.
25	BM	PCI Burst Master Control. 0: Stop PCI bus master. 1: Start PCI bus master.
24	Lat	PCI Latency Timer Enable. 0: Enable. 1: Disable.
23	PS	Panel Status. This bit is read-only. 0: Normal. 1: Flip pending.
22	VS	Video Status. This bit is read-only. 0: Normal. 1: Flip pending.
21	Res	This bit is reserved.
20	2E	2D Engine FIFO Status. This bit is read-only. 0: FIFO not empty. 1: FIFO empty.
19	2B	2D Engine Status. This bit is read-only. 0: Idle. 1: Busy.
18	Res	This bit is reserved.

Bit(s)	Name	Description
17	CS	CRT Status. This bit is read-only. 0: Normal. 1: Flip pending.
16	ZvS	ZV-Port Status. This bit is read-only. 0: Normal. 1: Vertical sync detected.
15	BrE	PCI Burst Read Enable. The BE bit must be enabled as well for this bit to take effect. 0: Disable. 1: Enable.
14	Res	This bit is reserved.
13:12	Abort	Drawing Engine Abort. 00: Normal. 11: Abort 2D engine.
11	Lck	Lock PCI Subsystem. 0: Unlocked. 1: Locked.
10:8	Res	These bits are reserved.
7	Rty	PCI Retry Enable. 0: Enable. 1: Disable.
6	Clk	PCI Clock Run Enable. 0: Disable. 1: Enable.
5:4	BrS	PCI Slave Burst Read Size. 00: 1 32-word. 01: 2 32-bit words. 10: 4 32-bit words. 11: 8 32-bit words.
3	Res	This bit is reserved.
2	CT	CRT Interface 3-State. 0: Normal. 1: 3-state.
1	MT	Local Memory Interface 3-State. 0: Normal. 1: 3-state.
0	PT	Panel Interface 3-State. 0: Normal. 1: 3-state.

Miscellaneous Control

Read/Write MMIO_base + 0x000004

Power-on Default 0b0000.0000.0000.00X0.0001.0000.XXX0.0XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Pad R/W		USBClk R/W		SSP R/W	Lat R/W	FP R/W	Freq R/W	Res	Refresh R/W		Hold R/W			SH R/W	II R/W
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PLL R/W	Gap R/W		DAC R/W	MC R/W	BL R/W	USB R/W	LB R/W	CDR R G7	Test R T1 T0		VR R/W	Clk R G3	Bus R G2 G1 G0		

Bit(s)	Name	Description
31:30	Pad	PCI Pad Drive Strength. 00: 24 ma. 01: 12 ma. 10: 8 ma.
29:28	USBClk	USB Clock Select. 00: Generated by crystal. 10: Generated by 96MHz host clock. 11: Generated by 48MHz host clock.
27	SSP	UART1/SSP1 Select. 0: UART1. 1: SSP1.
26	Lat	8051 Latch Enable for Address[11:8]. 0: Disable. 1: Enable.
25	FP	Flat Panel Data Select (controls function of GPIO[31:24]. 0: 18-bit (digital CRT). 1: 24-bit (24-bit panel).
24	Freq	Crystal Frequency Select. 0: 24MHz. 1: 12MHz.
23	Res	This bit is reserved.
22:21	Refresh	Internal Memory Refresh Control. 00: Refresh every 8µs. 01: Refresh every 16µs. 10: Refresh every 32µs. 11: Refresh every 64µs.
20:18	Hold	Bus Hold Time. 000: Hold bus until command FIFO is empty. 001: Hold bus for 8 transactions. 010: Hold bus for 16 transactions. 011: Hold bus for 24 transactions. 100: Hold bus for 32 transactions.
17	SH	Hitachi Ready Polarity. This bit is determined by the GPIO4 pin at reset. 0: (1) For SH series CPU, Active Low (2) For strapped SA1110 mode, the SRAM write shared with the SDRAM write. 1: (1) For SH series CPU, Active high. (2) For strapped SA1110 mode, the SRAM write is separated from the SDRAM write

Bit(s)	Name	Description
16	II	Interrupt Inverting. 0: Normal. 1: Inverted.
15	PLL	PLL Clock Count. 0: Disable. 1: Enable.
14:13	Gap	DAC Band Gap Control. 00: Default.
12	DAC	DAC Power Control. 0: Enable. 1: Disable.
11	MC	USB Slave Controller. 0: CPU. 1: 8051 µ-controller.
10	BL	CPU Master Burst Length Select. 0: Burst of 8. 1: Burst of 1.
9	USB	USB Port Select. 0: Master. 1: Slave.
8	LB	USB Loop Back Select. 0: Normal. 1: USB host to/from USB slave.
7	CDR	Clock Divider Reset Control. This bit is read-only and is determined by the GPIO7 pin at reset. 0: Do not reset clock divider. 1: Reset clock divider.
6:5	Test	Test Mode Select. These bits are read-only and are determined by the TEST0 and TEST1 pins at reset. 00: Normal mode. 01: Debugging mode. 10: NAND tree mode. 11: Memory test mode.
4	VR	NEC Memory Map Select. 0: MMIO located at 30MB (0x1E00000). 1: MMIO located at 62MB (0x3E00000).
3	Clk	Clock Select. This bit is read-only and is determined by the GPIO3 pin at reset. 0: PLL. 1: Test clock (14.31818MHz).
2:0	Bus	Host Bus Type. These bits are read-only and are determined by the GPIO2, GPIO1, and GPIO0 pins at reset. 000: Hitachi SH3/SH4. 001: PCI. 010: Intel XScale. 110: NEC VR4122/31.

$GPIO_{31:0}$ Control

Read/Write MMIO_base + 0x000008

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G_{31} R/W	G_{30} R/W	G_{29} R/W	G_{28} R/W	G_{27} R/W	G_{26} R/W	G_{25} R/W	G_{24} R/W	G_{23} R/W	G_{22} R/W	G_{21} R/W	G_{20} R/W	G_{19} R/W	G_{18} R/W	G_{17} R/W	G_{16} R/W
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G_{15} R/W	G_{14} R/W	G_{13} R/W	G_{12} R/W	G_{11} R/W	G_{10} R/W	G_{9} R/W	G_{8} R/W	G_{7} R/W	G_{6} R/W	G_{5} R/W	G_{4} R/W	G_{3} R/W	G_{2} R/W	G_{1} R/W	G_{0} R/W

Bit(s)	Name	Description
31	G_{31}	GPIO Pin 31 Control. 0: GPIO. 1: PWM2 or Test Data[10].
30	G_{30}	GPIO Pin 30 Control. 0: GPIO. 1: PWM1 or Test Data[9].
29	G_{29}	GPIO Pin 29 Control. 0: GPIO. 1: PWM0 or Test Data[8].
28	G_{28}	GPIO Pin 28 Control. 0: GPIO. 1: AC97 Serial Data In or I^2S.
27	G_{27}	GPIO Pin 27 Control. 0: GPIO. 1: AC97 Serial Data Out or I^2S.
26	G_{26}	GPIO Pin 26 Control. 0: GPIO. 1: AC97 Bit Clock or I^2S.
25	G_{25}	GPIO Pin 25 Control. 0: GPIO. 1: AC97 Sync 48kHz or I^2S.
24	G_{24}	GPIO Pin 24 Control. 0: GPIO. 1: AC97 Reset.
23	G_{23}	GPIO Pin 23 Control. 0: GPIO. 1: ZV-Port[7] or Test Data[7].
22	G_{22}	GPIO Pin 22 Control. 0: GPIO. 1: ZV-Port[6] or Test Data[6].
21	G_{21}	GPIO Pin 21 Control. 0: GPIO. 1: ZV-Port[5] or Test Data[5].
20	G_{20}	GPIO Pin 20 Control. 0: GPIO. 1: ZV-Port[4] or Test Data[4].

Bit(s)	Name	Description
19	G_{19}	GPIO Pin 19 Control. 0: GPIO. 1: ZV-Port[3] or Test Data[3].
18	G_{18}	GPIO Pin 18 Control. 0: GPIO. 1: ZV-Port[2] or Test Data[2].
17	G_{17}	GPIO Pin 17 Control. 0: GPIO. 1: ZV-Port[1] or Test Data[1].
16	G_{16}	GPIO Pin 16 Control. 0: GPIO. 1: ZV-Port[0] or Test Data[0].
15	G_{15}	GPIO Pin 15 Control. 0: GPIO. 1: 8051 Address[11].
14	G_{14}	GPIO Pin 14 Control. 0: GPIO. 1: 8051 Address[10].
13	G_{13}	GPIO Pin 13 Control. 0: GPIO. 1: 8051 Address[9].
12	G_{12}	GPIO Pin 12 Control. 0: GPIO. 1: 8051 Address[8].
11	G_{11}	GPIO Pin 11 Control. 0: GPIO. 1: 8051 Wait.
10	G_{10}	GPIO Pin 10 Control. 0: GPIO. 1: 8051 Address Latch Enable.
9	G_{9}	GPIO Pin 9 Control. 0: GPIO. 1: 8051 Write.
8	G_{8}	GPIO Pin 8 Control. 0: GPIO. 1: 8051 Read.
7	G_{7}	GPIO Pin 7 Control. 0: GPIO. 1: 8051 Address/Data[7].
6	G_{6}	GPIO Pin 6 Control. 0: GPIO. 1: 8051 Address/Data[6].
5	G_{5}	GPIO Pin 5 Control. 0: GPIO. 1: 8051 Address/Data[5].
4	G_{4}	GPIO Pin 4 Control. 0: GPIO. 1: 8051 Address/Data[4].
3	G_{3}	GPIO Pin 3 Control. 0: GPIO. 1: 8051 Address/Data[3].
2	G_{2}	GPIO Pin 2 Control. 0: GPIO. 1: 8051 Address/Data[2].

Bit(s)	Name	Description
1	G_1	GPIO Pin 1 Control. 0: GPIO. 1: 8051 Address/Data[1].
0	G_0	GPIO Pin 0 Control. 0: GPIO. 1: 8051 Address/Data[0].

$GPIO_{63:32}$ Control

Read/Write MMIO_base + 0x00000C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G_{63} R/W	G_{62} R/W	G_{61} R/W	G_{60} R/W	G_{59} R/W	G_{58} R/W	G_{57} R/W	G_{56} R/W	G_{55} R/W	Reserved						
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
G_{47} R/W	G_{46} R/W	G_{45} R/W	G_{44} R/W	G_{43} R/W	G_{42} R/W	G_{41} R/W	G_{40} R/W	G_{39} R/W	G_{38} R/W	G_{37} R/W	G_{36} R/W	G_{35} R/W	G_{34} R/W	G_{33} R/W	G_{32} R/W

Bit(s)	Name	Description
31	G_{63}	GPIO Pin 63 Control. 0: GPIO & ZV-Port[15]. 1: Digital CRT[7] or Test Data [19].
30	G_{62}	GPIO Pin 62 Control. 0: GPIO & ZV-Port[14]. 1: Digital CRT[6] or Test Data [18].
29	G_{61}	GPIO Pin 61 Control. 0: GPIO & ZV-Port[13]. 1: Digital CRT[5], Flat Panel[17], or Test Data [17].
28	G_{60}	GPIO Pin 60 Control. 0: GPIO & ZV-Port[12]. 1: Digital CRT[4], Flat Panel[16], or Test Data [16].
27	G_{59}	GPIO Pin 59 Control. 0: GPIO & ZV-Port[11]. 1: Digital CRT[3], Flat Panel[9], or Test Data [15].
26	G_{58}	GPIO Pin 58 Control. 0: GPIO & ZV-Port[10]. 1: Digital CRT[2], Flat Panel[8], or Test Data [14].
25	G_{57}	GPIO Pin 57 Control. 0: GPIO & ZV-Port[9]. 1: Digital CRT[1], Flat Panel[1], or Test Data [13].
24	G_{56}	GPIO Pin 56 Control. 0: GPIO & ZV-Port[8] . 1: Digital CRT[0], Flat Panel[0], or Test Data [12].
23	G_{55}	GPIO Pin 55 Control. 0: GPIO. 1: Digital CRT Clock or Test Data [11].

Bit(s)	Name	Description
22:16	Res	These bits are reserved.
15	G_{47}	GPIO Pin 47 Control. 0: GPIO. 1: I^2C Data.
14	G_{46}	GPIO Pin 46 Control. 0: GPIO. 1: I^2C Clock.
13	G_{45}	GPIO Pin 45 Control. 0: GPIO. 1: SSP1 Clock Out/In.
12	G_{44}	GPIO Pin 44 Control. 0: GPIO. 1: UART1 Request To Send or SSP1 Serial Frame Out.
11	G_{43}	GPIO Pin 43 Control. 0: GPIO. 1: UART1 Clear To Send or SSP1 Serial Frame In.
10	G_{42}	GPIO Pin 42 Control. 0: GPIO. 1: UART1 Receive or SSP1 Receive.
9	G_{41}	GPIO Pin 41 Control. 0: GPIO. 1: UART1 Transmit or SSP1 Transmit.
8	G_{40}	GPIO Pin 40 Control. 0: GPIO. 1: UART0 Request To Send.
7	G_{39}	GPIO Pin 39 Control. 0: GPIO. 1: UART0 Clear To Send.
6	G_{38}	GPIO Pin 38 Control. 0: GPIO. 1: UART0 Receive.
5	G_{37}	GPIO Pin 37 Control. 0: GPIO. 1: UART0 Transmit.
4	G_{36}	GPIO Pin 36 Control. 0: GPIO. 1: SSP0 Clock Out/In.
3	G_{35}	GPIO Pin 35 Control. 0: GPIO. 1: SSP0 Serial Frame In.
2	G_{34}	GPIO Pin 34 Control. 0: GPIO. 1: SSP0 Serial Frame Out.
1	G_{33}	GPIO Pin 33 Control. 0: GPIO. 1: SSP0 Receive.
0	G_{32}	GPIO Pin 32 Control. 0: GPIO. 1: SSP0 Transmit.

DRAM Control

Read/Write MMIO_base + 0x000010

Power-on Default 0bX000.0111.1111.0001.XXXX.X111.1100.0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
E R G12	Burst R/W			CL R/W	$Size_x$ R/W			$ColSize_x$ R/W		AP_x R/W	Rst_x R/W	Bks_x R/W	WP_x R/W	BwE R/W	Rfsh R/W
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Size R/W G15 G14 G13			ColSize R/W G6 G5		BwC R/W	BwP R/W	AP R/W	Rst R/W	RA R/W	Reserved				Bks R/W	WP R/W

Bit(s)	Name	Description
31	E	Embedded Memory Control. This bit is read-only and is determined by the GPIO12 pin during reset. 0: Enable. 1: Disable.
30:28	Burst	External Memory Burst Length. 000: 1 word. 001: 2 words. 010: 4 words. 011: 8 words.
27	CL	External Memory CAS Latency. 0: 2 clocks. 1: 3 clocks.
26:24	$Size_X$	External Memory Size. 000: 2MB 001: 4MB 100: 64MB 101: 32MB 110: 16MB 111: 8MB
23:22	$ColSize_X$	External Memory Column Size. 00: 1024 words. 10: 512 words. 11: 256 words.
21	AP_X	External Memory Active to Pre-charge Delay. 0: 6 clocks. 1: 7 clocks.
20	Rst_X	External Memory Reset. 0: Reset. 1: Normal.
19	Bks_X	External Memory Number of Banks. 0: 4 banks. 1: 2 banks.
18	WP_X	External Memory Write to Pre-change Delay. 0: 2 clocks. 1: 1 clock.

Bit(s)	Name	Description
17	BwE	Local Memory Block Write Enable. 0: Disabled. 1: Enabled.
16	Rfsh	Local Memory Refresh to Command Delay. 0: 10 clocks. 1: 12 clocks.
15:13	Size	Local Memory Size. These bits are determined by the GPIO15 through GPIO13 pins during reset, but can be changed by software. 000: 4MB. 001: 8MB. 010: 16MB. 011: 32MB. 100: 64MB. 101: 2MB.
12:11	ColSize	Local Memory Column Size. These bits are determined by the GPIO6 and GPIO5 pins at reset, but can be changed by software. 00: 256 words. 10: 512 words. 11: 1024 words.
10	BwC	Local Memory Block Write Cycle Time. 0: 1 clock. 1: 2 clocks.
9	BwP	Local Memory Block Write to Pre-charge Delay. 0: 4 clocks. 1: 1 clock.
8	AP	Internal Memory Active to Pre-charge Delay. 0: 6 clocks. 1: 7 clocks.
7	Rst	Internal Memory Reset. 0: Reset. 1: Normal.
6	RA	Internal Memory Remain in Active State. 0: Remain active. 1: Do not remain active.
5:2	Res	These bits are reserved.
1	Bks	Internal Memory Number of Banks. 0: 4 banks. 1: 2 banks.
0	WP	Internal Memory Write to Pre-charge Delay. 0: 2 clocks. 1: 1 clock.

Arbitration Control

Read/Write MMIO_base + 0x000014

Power-on Default 0x05146732

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res		Ext R/W	Int R/W	Res	USB R/W			Res	Panel R/W			Res	ZVPort R/W		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res	Command R/W			Res	DMA R/W			Res	Video R/W			Res	CRT R/W		

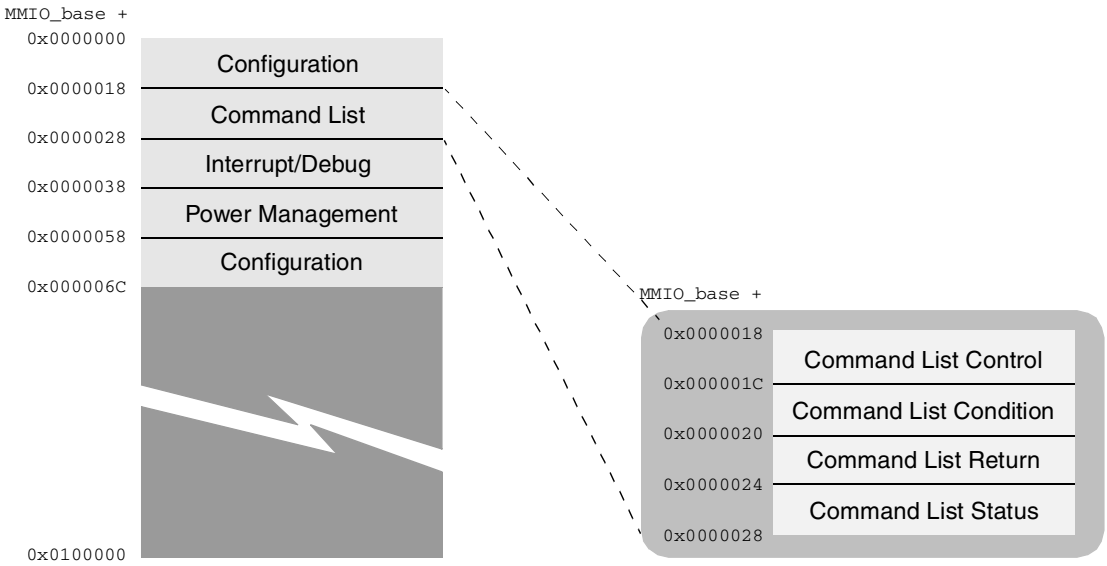
Bit(s)	Name	Description
31:30	Res	These bits are reserved.
29	Ext	External Memory Priority Scheme. 0: Fixed priority. 1: Revolving priority.
28	Int	Internal Memory Priority Scheme. 0: Fixed priority. 1: Revolving priority.
27	Res	This bit is reserved.
26:24	USB	USB FIFO Priority. 000 Off; 001 Priority 1 (highest).; 010 Priority 2.; 011 Priority 3.; 100 Priority 4.; 101 Priority 5.; 110 Priority 6.; 111 Priority 7 (lowest).
23	Res	This bit is reserved.
22:20	Panel	Panel FIFO Priority. 000 Off; 001 Priority 1 (highest).; 010 Priority 2.; 011 Priority 3.; 100 Priority 4.; 101 Priority 5.; 110 Priority 6.; 111 Priority 7 (lowest).
19	Res	This bit is reserved.
18:16	ZVPort	ZV Port FIFO Priority. 000 Off; 001 Priority 1 (highest).; 010 Priority 2.; 011 Priority 3.; 100 Priority 4.; 101 Priority 5.; 110 Priority 6.; 111 Priority 7 (lowest).
15	Res	This bit is reserved.

<table>
<tr><th>Bit(s)</th><th>Name</th><th>Description</th></tr>
<tr><td>14:12</td><td>Command</td><td>Command List Interpreter FIFO Priority.
<table>
<tr><td>000</td><td>Off</td><td>100</td><td>Priority 4.</td></tr>
<tr><td>001</td><td>Priority 1 (highest).</td><td>101</td><td>Priority 5.</td></tr>
<tr><td>010</td><td>Priority 2.</td><td>110</td><td>Priority 6.</td></tr>
<tr><td>011</td><td>Priority 3.</td><td>111</td><td>Priority 7 (lowest).</td></tr>
</table></td></tr>
<tr><td>11</td><td>Res</td><td>This bit is reserved.</td></tr>
<tr><td>10:8</td><td>DMA</td><td>DMA FIFO Priority.
<table>
<tr><td>000</td><td>Off</td><td>100</td><td>Priority 4.</td></tr>
<tr><td>001</td><td>Priority 1 (highest).</td><td>101</td><td>Priority 5.</td></tr>
<tr><td>010</td><td>Priority 2.</td><td>110</td><td>Priority 6.</td></tr>
<tr><td>011</td><td>Priority 3.</td><td>111</td><td>Priority 7 (lowest).</td></tr>
</table></td></tr>
<tr><td>7</td><td>Res</td><td>This bit is reserved.</td></tr>
<tr><td>6:4</td><td>Video</td><td>Video FIFO Priority.
<table>
<tr><td>000</td><td>Off</td><td>100</td><td>Priority 4.</td></tr>
<tr><td>001</td><td>Priority 1 (highest).</td><td>101</td><td>Priority 5.</td></tr>
<tr><td>010</td><td>Priority 2.</td><td>110</td><td>Priority 6.</td></tr>
<tr><td>011</td><td>Priority 3.</td><td>111</td><td>Priority 7 (lowest).</td></tr>
</table></td></tr>
<tr><td>3</td><td>Res</td><td>This bit is reserved.</td></tr>
<tr><td>2:0</td><td>CRT</td><td>CRT FIFO Priority.
<table>
<tr><td>000</td><td>Off</td><td>100</td><td>Priority 4.</td></tr>
<tr><td>001</td><td>Priority 1 (highest).</td><td>101</td><td>Priority 5.</td></tr>
<tr><td>010</td><td>Priority 2.</td><td>110</td><td>Priority 6.</td></tr>
<tr><td>011</td><td>Priority 3.</td><td>111</td><td>Priority 7 (lowest).</td></tr>
</table></td></tr>
</table>

Command List Register Descriptions

The Command List registers control the Command List Interpreter. Figure 2-3 shows the layout of the registers that control the Command List Interpreter.

Figure 2-3: Command List Register Space



Command List Status

Read MMIO_base + 0x000024

Power-on Default 0b0000.0000.000X.XXXX.XXXX.X000.0000.0XXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved											2_M R	C_F R	2_C R	D_M R	C_S R
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
V_F R	V_S R	P_S R	S_C R	S_P R	Reserved								2_S R	2_F R	2_E R

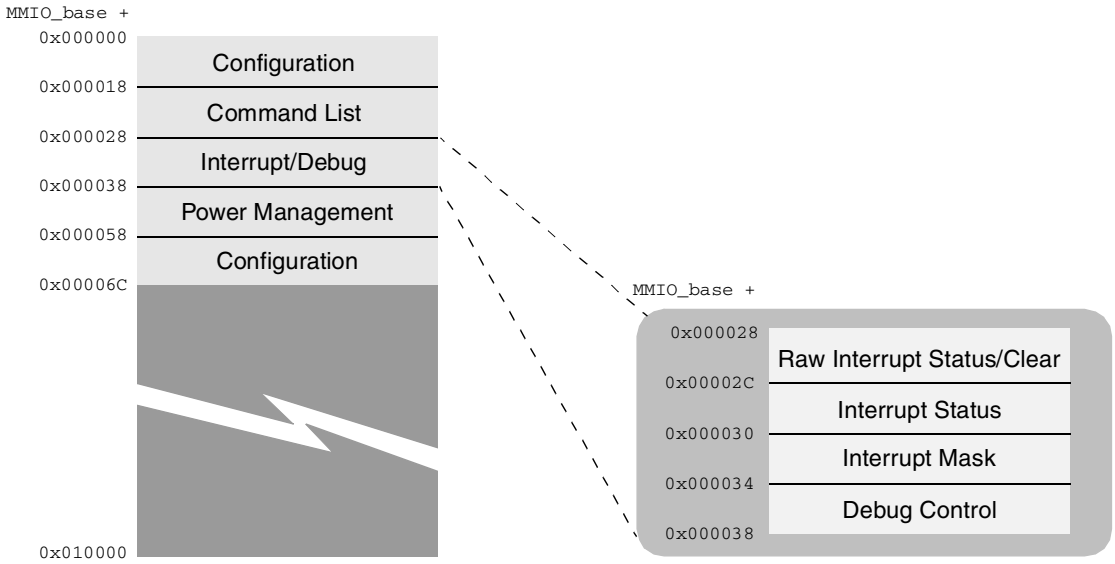
Bit(s)	Name	Description
31:21	Reserved	These bits are reserved.
20	2_M	2D Memory FIFO Status. 0: Not empty. 1: Empty.
19	C_F	Command FIFO on HIF bus. 0: Not empty. 1: Empty.

Bit(s)	Name	Description
18	2_C	2D Color Space Conversion Status. 0: Idle. 1: Busy.
17	D_M	Memory DMA Status. 0: Idle. 1: Busy.
16	C_S	CRT Graphics Layer Status. 0: No flip pending. 1: Flip in progress.
15	V_F	Current Video Field. 0: Odd. 1: Even.
14	V_S	Video Layer Status. 0: No flip pending. 1: Flip in progress.
13	P_S	Panel Graphics Layer Status. 0: No flip pending. 1: Flip in progress.
12	S_C	CRT Vertical Sync Status. 0: Not active. 1: Active.
11	S_P	Panel Vertical Sync Status. 0: Not active. 1: Active.
10:3	Reserved	These bits are reserved.
2	2_S	2D Setup Engine Status. 0: Idle. 1: Busy.
1	2_F	2D Command FIFO Status. 0: Not empty. 1: Empty.
0	2_E	2D Engine Status. 0: Idle. 1: Busy.

Interrupt / Debug Register Descriptions

The Interrupt / Debug registers reflect the status of interrupts, allow for enabling and disabling different interrupts and control which area of the chip is to be debugged in the Debugging Test Mode. Figure 2-4 lists the registers available in the Interrupt and Debug register space.

Figure 2-4: Interrupt / Debug Register Space



Raw Interrupt Status

Read MMIO_base + 0x000028

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved										UP R/W	ZV R/W	CV R/W	US R/W	PV R/W	CI R/W

Note: Write a 1 to clear; writing a 0 has no effect.

Bit(s)	Name	Description
31:6	Reserved	These bits are reserved.
5	UP	USB Slave Plug-in Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
4	ZV	ZV-Port Interrupt Status. 0: Interrupt not active. 1: Interrupt active.

Bit(s)	Name	Description
3	CV	CRT Vertical Sync Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
2	US	USB Slave Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
1	PV	Panel Vertical Sync Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
0	CI	Command Interpreter Interrupt Status. 0: Interrupt not active. 1: Interrupt active.

Raw Interrupt Clear

Write MMIO_base + 0x000028

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved										UP W	ZV W	CV W	US W	PV W	CI W

Note: Write a 1 to clear; writing a 0 has no effect.

Bit(s)	Name	Description
31:6	Reserved	These bits are reserved.
5	UP	USB Slave Plug-in Interrupt Clear. 0: No action. 1: Clear interrupt.
4	ZV	ZV-Port Interrupt Clear. 0: No action. 1: Clear interrupt.
3	CV	CRT Vertical Sync Interrupt Clear. 0: No action. 1: Clear interrupt.
2	US	USB Slave Interrupt Clear. 0: No action. 1: Clear interrupt.
1	PV	Panel Vertical Sync Interrupt Clear. 0: No action. 1: Clear interrupt.
0	CI	Command Interpreter Interrupt Clear. 0: No action. 1: Clear interrupt.

Interrupt Status

Read MMIO_base + 0x00002C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UP R	G_{54} R	G_{53} R	G_{52} R	G_{51} R	G_{50} R	G_{49} R	G_{48} R	I^2C R	PW R	Res	DMA R	PCI R	I^2S R	AC R	US R
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res		U1 R	U0 R	CV R	MC R	S1 R	S0 R	Res	UH R	Res		2D R	ZV R	PV R	CI R

Bit(s)	Name	Description
31	UP	USB Slave Plug-in Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
30	G_{54}	GPIO Pin 54 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
29	G_{53}	GPIO Pin 53 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
28	G_{52}	GPIO Pin 52 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
27	G_{51}	GPIO Pin 51 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
26	G_{50}	GPIO Pin 50 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
25	G_{49}	GPIO Pin 49 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
24	G_{48}	GPIO Pin 48 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
23	I^2C	I^2C Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
22	PW	PWM Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
21	Res	This bit is reserved.
20	DMA	DMA Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
19	PCI	PCI Interrupt Status. 0: Interrupt not active. 1: Interrupt active.

Bit(s)	Name	Description
18	I^2S	I^2S Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
17	AC	AC97 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
16	US	USB Slave Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
15:14	Res	These bits are reserved.
13	U1	UART1 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
12	U0	UART0 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
11	CV	CRT Vertical Sync Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
10	MC	8051 µ-Controller Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
9	S1	SSP1 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
8	S0	SSP0 Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
7	Res	These bits are reserved.
6	UH	USB Host Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
5:4	Res	These bits are reserved.
3	2D	2D Engine Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
2	ZV	ZV-Port Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
1	PV	Panel Vertical Sync Interrupt Status. 0: Interrupt not active. 1: Interrupt active.
0	CI	Command Interpreter Interrupt Status. 0: Interrupt not active. 1: Interrupt active.

Interrupt Mask

Read/Write MMIO_base + 0x000030

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UP R/W	G_{54} R/W	G_{53} R/W	G_{52} R/W	G_{51} R/W	G_{50} R/W	G_{49} R/W	G_{48} R/W	I^2C R/W	PW R/W	Res	DMA R/W	PCI R/W	I^2S R/W	AC R/W	US R/W
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res		U1 R/W	U0 R/W	CV R/W	MC R/W	S1 R/W	S0 R/W	Res	UH R/W	Res		2D R/W	ZV R/W	PV R/W	CI R/W

Bit(s)	Name	Description
31	UP	USB Slave Plug-in Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
30	G_{54}	GPIO Pin 54 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
29	G_{53}	GPIO Pin 53 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
28	G_{52}	GPIO Pin 52 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
27	G_{51}	GPIO Pin 51 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
26	G_{50}	GPIO Pin 50 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
25	G_{49}	GPIO Pin 49 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
24	G_{48}	GPIO Pin 48 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
23	I^2C	I^2C Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
22	PW	PWM Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
21	Res	This bit is reserved.
20	DMA	DMA Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
19	PCI	PCI Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.

Bit(s)	Name	Description
18	I^2S	I^2S Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
17	AC	AC97 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
16	US	USB Slave Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
15:14	Res	These bits are reserved.
13	U1	UART1 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
12	U0	UART0 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
11	CV	CRT Vertical Sync Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
10	MC	8051 µ-Controller Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
9	S1	SSP1 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
8	S0	SSP0 Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
7	Res	This bit is reserved.
6	UH	USB Host Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
5:4	Res	These bits are reserved.
3	2D	2D Engine Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
2	ZV	ZV-Port Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
1	PV	Panel Vertical Sync Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.
0	CI	Command Interpreter Interrupt Mask. 0: Interrupt disabled. 1: Interrupt enabled.

Debug Control

Read/Write MMIO_base + 0x000034

Power-on Default 0x00000000

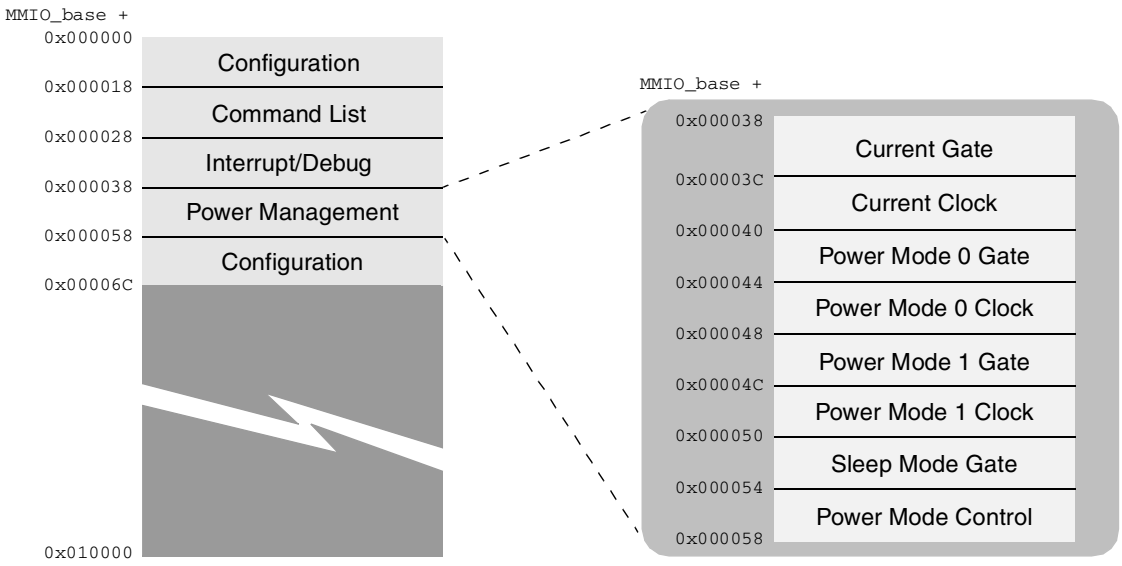
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								Module R/W			Partition R/W				

Bit(s)	Name	Description
31:8	Reserved	These bits are reserved.
7:5	Module	Module Select for Partition.
4:0	Partition	Partition select for Debugging Test Mode. 00000: HIF controller 00001: External memory controller 00010: PCI controller 00011: Command Interpreter 00100: Display controller 00101: ZV-Port 00110: 2D Engine 01000: Local memory interface 01010: USB Host 01100: SSP0 01101: SSP1 10011: UART0 10100: UART1 10101: I^2C 10111: 8051 µ-controller interface 11000: AC97 11001: I^2S 11010: Local memory controller 11011: DMA 11100: Simulation test mode

Power Management Register Descriptions

The Power Management registers control which areas of the SM501 are running and at which speed. Figure 2-5 lists the registers available in the Power Management register space.

Figure 2-5: Power Management Register Space



Current Gate

Read MMIO_base + 0x000038

Power-on Default 0x00021807

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res													AC R	µP R	P R
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
O R	R R		US R	UH R	S R	Res	U1 R	U0 R	G R	ZV R	C R	2D R	D R	M R	H R

Bit(s)	Name	Description
31:19	Res	These bits are reserved.
18	AC	AC97 and I^2S Clock Control. 0: Disable. 1: Enable.
17	µP	8051 µ-Controller and SRAM Clock Control. 0: Disable. 1: Enable.

Bit(s)	Name	Description
16	P	PLL Control. 0: Enable. 1: Disable.
15	O	Oscillator Control. 0: Enable. 1: Disable.
14:13	R	PLL Recovery Time. 00: 1ms. 01: 2ms. 10: 3ms. 11: 4ms.
12	US	USB Slave Clock Control. 0: Disable. 1: Enable.
11	UH	USB Host Clock Control. 0: Disable. 1: Enable.
10	S	SSP0 and SSP1 Clock Control. 0: Disable. 1: Enable.
9	Res	This bit is reserved.
8	U1	UART1 Clock Control. 0: Disable. 1: Enable.
7	U0	UART0 Clock Control. 0: Disable. 1: Enable.
6	G	GPIO, PWM, and I^2C Clock Control. 0: Disable. 1: Enable.
5	Z	ZV-Port Clock Control. 0: Disable. 1: Enable.
4	C	Color Space Conversion Clock Control. 0: Disable. 1: Enable.
3	2D	2D Engine Clock Control. 0: Disable. 1: Enable.
2	D	Display Controller Clock Control. 0: Disable. 1: Enable.
1	M	Memory Controller Clock Control. 0: Disable. 1: Enable.
0	H	Host Interface, Command Interpreter, and DMA Clock Control. 0: Disable. 1: Enable.

Current Clock

Read MMIO_base + 0x00003C

Power-on Default 0x2A1A0A09

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		$P2_S$ R	P2 R					Reserved			$V2_S$ R	V2 R			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			M_S R	M R				Reserved			$M1_S$ R	M1 R			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29	$P2_S$	P2XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
28:24	P2	P2XCLK Frequency Divider.
23:21	Reserved	These bits are reserved.
20	$V2_S$	V2XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
19:16	V2	V2XCLK Frequency Divider.
15:13	Reserved	These bits are reserved.
12	M_S	MCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).

00000	÷ 1	01000	÷ 3	10000	÷ 5		
00001	÷ 2	01001	÷ 6	10001	÷ 10		
00010	÷ 4	01010	÷ 12	10010	÷ 20		
00011	÷ 8	01011	÷ 24	10011	÷ 40		
00100	÷ 16	01100	÷ 48	10100	÷ 80		
00101	÷ 32	01101	÷ 96	10101	÷ 160		
00110	÷ 64	01110	÷ 192	10110	÷ 320		
00111	÷ 128	01111	÷ 384	10111	÷ 640		

0000	÷ 1	0100	÷ 16	1000	÷ 3	1100	÷ 48
0001	÷ 2	0101	÷ 32	1001	÷ 6	1101	÷ 96
0010	÷ 4	0110	÷ 64	1010	÷ 12	1110	÷ 192
0011	÷ 8	0111	÷ 128	1011	÷ 24	1111	÷ 384

Bit(s)	Name	Description
11:8	M	MCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384
7:5	Reserved	These bits are reserved.
4	$M1_S$	M1XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
3:0	M1	M1XCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384

Power Mode 0 Gate

Read/Write MMIO_base + 0x000040

Power-on Default 0x00021807

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res													AC R/W	µP R/W	Res
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res			US R/W	UH R/W	S R/W	Res	U1 R/W	U0 R/W	G R/W	ZV R/W	C R/W	2D R/W	D R/W	M R/W	H R/W

Bit(s)	Name	Description
31:19	Res	These bits are reserved.
18	AC	AC97 and I^2S Clock Control. 0: Disable. 1: Enable.
17	µP	8051 µ-Controller and SRAM Clock Control. 0: Disable. 1: Enable.
16:13	Res	These bits are reserved.
12	US	USB Slave Clock Control. 0: Disable. 1: Enable.

Bit(s)	Name	Description
11	UH	USB Host Clock Control. 0: Disable. 1: Enable.
10	S	SSP0 and SSP1 Clock Control. 0: Disable. 1: Enable.
9	Res	This bit is reserved.
8	U1	UART1 Clock Control. 0: Disable. 1: Enable.
7	U0	UART0 Clock Control. 0: Disable. 1: Enable.
6	G	GPIO, PWM, and I^2C Clock Control. 0: Disable. 1: Enable.
5	Z	ZV-Port Clock Control. 0: Disable. 1: Enable.
4	C	Color Space Conversion Clock Control. 0: Disable. 1: Enable.
3	2D	2D Engine Clock Control. 0: Disable. 1: Enable.
2	D	Display Controller Clock Control. 0: Disable. 1: Enable.
1	M	Memory Controller Clock Control. 0: Disable. 1: Enable.
0	H	Host Interface, Command Interpreter, and DMA Clock Control. 0: Disable. 1: Enable.

Power Mode 0 Clock

Read/Write MMIO_base + 0x000044

Power-on Default 0x2A1A0A09

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		$P2_S$ R/W	P2 R/W					Reserved			$V2_S$ R/W	V2 R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			M_S R/W	M R/W				Reserved			$M1_S$ R/W	M1 R/W			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29	$P2_S$	P2XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
28:24	P2	P2XCLK Frequency Divider.
23:21	Reserved	These bits are reserved.
20	$V2_S$	V2XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
19:16	V2	V2XCLK Frequency Divider.
15:13	Reserved	These bits are reserved.
12	M_S	MCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).

00000	÷ 1	01000	÷ 3	10000	÷ 5		
00001	÷ 2	01001	÷ 6	10001	÷ 10		
00010	÷ 4	01010	÷ 12	10010	÷ 20		
00011	÷ 8	01011	÷ 24	10011	÷ 40		
00100	÷ 16	01100	÷ 48	10100	÷ 80		
00101	÷ 32	01101	÷ 96	10101	÷ 160		
00110	÷ 64	01110	÷ 192	10110	÷ 320		
00111	÷ 128	01111	÷ 384	10111	÷ 640		

0000	÷ 1	0100	÷ 16	1000	÷ 3	1100	÷ 48
0001	÷ 2	0101	÷ 32	1001	÷ 6	1101	÷ 96
0010	÷ 4	0110	÷ 64	1010	÷ 12	1110	÷ 192
0011	÷ 8	0111	÷ 128	1011	÷ 24	1111	÷ 384

Bit(s)	Name	Description
11:8	M	MCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384
7:5	Reserved	These bits are reserved.
4	$M1_S$	M1XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
3:0	M21	M1XCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384

Power Mode 1 Gate

Read/Write MMIO_base + 0x000048

Power-on Default 0x00021807

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res													AC R/W	µP R/W	Res
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res			US R/W	UH R/W	S R/W	Res	U1 R/W	U0 R/W	G R/W	ZV R/W	O R/W	2D R/W	D R/W	M R/W	H R/W

Bit(s)	Name	Description
31:19	Res	These bits are reserved.
18	AC	AC97 and I^2S Clock Control. 0: Disable. 1: Enable.
17	µP	8051 µ-Controller and SRAM Clock Control. 0: Disable. 1: Enable.
16:13	Res	These bits are reserved.
12	US	USB Slave Clock Control. 0: Disable. 1: Enable.

Bit(s)	Name	Description
11	UH	USB Host Clock Control. 0: Disable. 1: Enable.
10	S	SSP0 and SSP1 Clock Control. 0: Disable. 1: Enable.
9	Res	This bit is reserved.
8	U1	UART1 Clock Control. 0: Disable. 1: Enable.
7	U0	UART0 Clock Control. 0: Disable. 1: Enable.
6	G	GPIO, PWM, and I^2C Clock Control. 0: Disable. 1: Enable.
5	Z	ZV-Port Clock Control. 0: Disable. 1: Enable.
4	C	Color Space Conversion Clock Control. 0: Disable. 1: Enable.
3	2D	2D Engine Clock Control. 0: Disable. 1: Enable.
2	D	Display Controller Clock Control. 0: Disable. 1: Enable.
1	M	Memory Controller Clock Control. 0: Disable. 1: Enable.
0	H	Host Interface, Command Interpreter, and DMA Clock Control. 0: Disable. 1: Enable.

Power Mode 1 Clock

Read/Write MMIO_base + 0x00004C

Power-on Default 0x2A1A0A09

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		$P2_S$ R/W	P2 R/W					Reserved			$V2_S$ R/W	V2 R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			M_S R/W	M R/W				Reserved			$M1_S$ R/W	M1 R/W			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29	$P2_S$	P2XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
28:24	P2	P2XCLK Frequency Divider. 00000 ÷ 1 | 01000 ÷ 3 | 10000 ÷ 5 00001 ÷ 2 | 01001 ÷ 6 | 10001 ÷ 10 00010 ÷ 4 | 01010 ÷ 12 | 10010 ÷ 20 00011 ÷ 8 | 01011 ÷ 24 | 10011 ÷ 40 00100 ÷ 16 | 01100 ÷ 48 | 10100 ÷ 80 00101 ÷ 32 | 01101 ÷ 96 | 10101 ÷ 160 00110 ÷ 64 | 01110 ÷ 192 | 10110 ÷ 320 00111 ÷ 128 | 01111 ÷ 384 | 10111 ÷ 640
23:21	Reserved	These bits are reserved.
20	$V2_S$	V2XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
19:16	V2	V2XCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384
15:13	Reserved	These bits are reserved.
12	M_S	MCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).

Bit(s)	Name	Description
11:8	M	MCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384
7:5	Reserved	These bits are reserved.
4	$M1_S$	M1XCLK Frequency Input Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68 on page 2-41).
3:0	M1	M1XCLK Frequency Divider. 0000 ÷ 1 | 0100 ÷ 16 | 1000 ÷ 3 | 1100 ÷ 48 0001 ÷ 2 | 0101 ÷ 32 | 1001 ÷ 6 | 1101 ÷ 96 0010 ÷ 4 | 0110 ÷ 64 | 1010 ÷ 12 | 1110 ÷ 192 0011 ÷ 8 | 0111 ÷ 128 | 1011 ÷ 24 | 1111 ÷ 384

Sleep Mode Gate

Read/Write MMIO_base + 0x000050

Power-on Default 0x00018000

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

Bit(s)	Name	Description
31:23	Reserved	These bits are reserved.
22:19	D	PLL Recovery Clock Divider. 0000 ÷ 4096 | 0100 ÷ 256 | 1000 ÷ 16 0001 ÷ 2048 | 0101 ÷ 128 | 1001 ÷ 8 0010 ÷ 1024 | 0110 ÷ 64 | 1010 ÷ 4 0011 ÷ 512 | 0111 ÷ 32 | 1011 ÷ 2 Internally, the PLL recovery time counters are based on a 32µs clock. So you have to program the *D* field to make the host clock come as close to 32µs as possible.

Bit(s)	Name	Description
18:15	Reserved	These bits are reserved.
14:13	R	PLL Recovery. 00: 1ms (32 counts). 01: 2ms (64 counts). 10: 3ms (96 counts). 11: 4ms (128 counts).
12:0	Reserved	These bits are reserved.

Power Mode Control

Read/Write MMIO_base + 0x000054

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:3	Reserved	These bits are reserved.
2	S	Current Sleep Status. 0: Not in sleep mode. 1: In sleep mode. When the SM501 is transitioning back from sleep mode to a normal power mode (Modes 0 or 1), the software needs to poll this bit until it becomes “0” before writing any other commands to the chip.
1:0	Mode	Power Mode Select. 00: Power Mode 0. 01: Power Mode 1. 10: Sleep Mode.

Clock Multiply PLLs

An external crystal combined with the on-chip oscillator provides the SM501 clock source. The external crystal frequency should be fixed at 24 MHz ± 5%.

The SM501 contains three clock multiply PLLs:

- PLL0

  Input frequency = 24 MHz oscillator
  Output frequency = Input frequency x 4 = 96 MHz fixed

- PLL1

  Input frequency = Output of PLL0 / 2 = 48 MHz
  Output frequency = Input frequency x 6 = 288 MHz fixed

- PLL2

  Input frequency = Output of PLL0 / 2 = 48 MHz
  The output frequency is controlled by the Miscellaneous Timing register, MMIO_base + 0x000068 as follows:

  Bits [5:4] = 00, the output of PLL2 = 48 MHz x 7 = 336 MHz, power on default
  Bits [5:4] = 01, the output of PLL2 = 48 MHz x 6 = 288 MHz
  Bits [5:4] = 10, the output of PLL2 = 48 MHz x 5 = 240 MHz
  Bits [5:4] = 11, the output of PLL2 = 48 MHz x 4 = 192 MHz

Power Mode 0, Power Mode 1, and Sleep Mode

There are two operational power modes (Power Mode 0 and Power Mode 1) and Sleep Mode. Each operational power mode has its own Power Mode Gate register to control the clock gating for each functional block. Each operational power mode also has its own Power Mode Clock register that selects the different clock frequencies for each clock branch. In Sleep Mode, all clocks are shut down.

The power mode switch should be controlled by power management software; for example, Power Mode 0 can be programmed to a high clock rate for maximum performance. Power Mode 1 can be programmed to a lower clock rate for sustaining operation.

The Power Mode Control Register at MMIO_base + 0x000054 determines the power mode selection as follows:

- Bits [1:0] = 00, select Power Mode 0 (power on default)
- Bits [1:0] = 01, select Power Mode 1.
- Bits [1:0] = 10, Sleep Mode. The PLLs and the oscillator are shut down in this mode.
- Bits [1:0] = 11, Reserved.

In Power Mode 0, the Power Mode 0 Clock Register at MMIO_base + 0x000044 controls the clock settings. The default of the Power Mode 0 Clock Register is 0x2A1A0A09. The Power Mode 0 Gate Register at MMIO_base + 0x000040 controls clock gating. The default of the Power Mode 0 Gate Register is 0x00021807.

In Power Mode 1, Power Mode 1 Clock Register at MMIO_base + 0x00004C controls the clock settings. The default of the Power Mode 1 Clock Register is 0x2A1A0A09. The Power Mode 1 Gate Register at MMIO_base + 0x000048 controls clock gating. The default of the Power Mode 1 Gate Register is 0x00021807.

Adjusting the Clock Frequency

There are five clock branches that can be programmed through the Power Mode 0 Clock, Power Mode 1 Clock, and Miscellaneous Timing registers:

Table 2-2: Programmable Clock Branches

Clock	Description
P2XCLK	2X clock source for the Panel interface timing. The actual rate at which the pixels are shifted out is P2XCLK divided by two.
V2XCLK	2X clock source for the CRT interface timing. The actual rate at which the pixels are shifted out is V2XCLK divided by two.
M1XCLK	Clock source for the local SDRAM controller.
MCLK	Main clock source for all functional blocks, such as the 2D engine, GPIO, Video Engine, DMA Engine.
SYSCLK[1]	Clock source for the system memory (CPU memory) controller. This clock also can be selected as the XScale CPU interface logic clock.

1. The definition of SYSCLK applies to the SM501, Rev AA only. The M1XCLK signal on Rev A and B test chips is used to drive the system memory controller.

In the Miscellaneous Timing Register (MMIO_base + 0x000068):

- Bits [20:16]: SYSCLK control if Power Mode 1 is selected. The power-on default for these bits is 0x09.
- Bits [12:8]: SYSCLK control if Power Mode 0 is selected. The power-on default for these bits is 0x09.

Current Gate and Current Clock Registers

The Current Gate (MMIO_base + 0x000038) and Current Clock (MMIO_base + 0x00003C) registers are read-only registers that reflect the current clock control selection. When Power Mode 0 is selected, the value read from this register should be the same as the value in the Power Mode 0 Clock register.

Rules to Program the Power Mode Clock Registers for Clock Selection

1. There should be only one clock source changed at a time. To change clock source for P2XCLK, V2XCLK, MCLK, M1XCLK simultaneously may cause the internal logic normal operation to be disrupted. There should be a minimum of **16mS wait** from change one clock source to another.
2. When adjusting the clock rate, the PLL selection bit should be programmed first before changing the divider value for each clock source. For example, to change the P2XCLK clock rate:
   - bit 29 should be set first
   - wait for a minimum of 16ms (about one Vsync time)
   - adjust bits [28:24].

   The minimum 16 ms wait is necessary for logic to settle down before the clock rate is changed.
3. There should be a minimum 16 ms wait after a clock source is changed before any operation that could result in a bus transaction.

Power Down (Sleep Mode)

There are three ways to power down the SM501 chip (sleep mode):

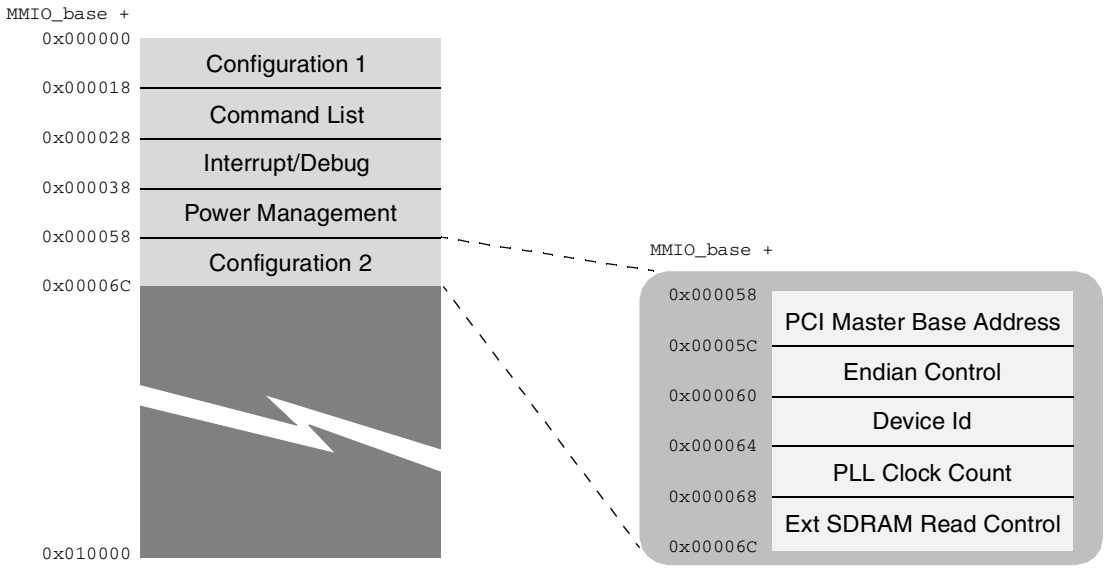
1. For CPU local bus interface only:
   MMIO_base + 0x000068, bit 6 = 0; disable PCI ACPI mode
   MMIO_base + 0x000054, bits [1:0] = 10
   PCI Config register 44, bits [1:0] = xx (don't care)

2. For PCI bus interface only, without support PCI ACPI protocol (this method applies only to SM501 Rev AA and later):
   MMIO_base + 0x000068, bit 6 = 0; disable PCI ACPI mode
   MMIO_base + 0x000054 bits [1:0] = 10
   PCI Config register 44, bits [1:0] = xx (don't care)
3. For PCI bus interface only, with support of PCI ACPI protocol:
   MMIO_base + 0x000068, bit 6 = 0; enable PCI ACPI mode
   MMIO_base + 0x000054, bits [1:0] = xx (don't care)
   PCI Config register 44, bits [1:0] = 1x (ACPI power down mode)

Configuration 2 Register Descriptions

The Configuration registers control the way the SM501 chips operates. Figure 2-2 shows the layout of the configuration registers in Configuration Register Space 2.

Figure 2-6: Configuration Register Space 2



PCI Master Base Address

Read/Write MMIO_base + 0x000058

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Base_{31:20}$ R/W												Reserved			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved															

Bit(s)	Name	Description
31:20	$Base_{31:20}$	PCI Master Base Address Bits [31:20].
19:0	Reserved	These bits are reserved.

Endian Control

Read/Write MMIO_base + 0x00005C

Power-on Default 0b0000.0000.0000.0000.0000.0000.0000.0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved															E G4

Bit(s)	Name	Description
31:1	Reserved	These bits are reserved.
0	E	Endian Select. To program the correct endianess for the CPU, the CPU should either write 0x00000000 (for little endian) or 0xFFFFFFFF (for big endian) into this register before touching any other register on the SM501. 0: Little endian. 1: Big endian.

Device Id

Read MMIO_base + 0x000060

Power-on Default 0x050100A0

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

Bit(s)	Name	Description
31:16	DeviceId	Device Identification: 0x0501.
15:8	Reserved	These bits are reserved.
7:0	RevisionId	Revision Identification: 0xA0.

PLL Clock Count

Read MMIO_base + 0x000064

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	ClockCount	Number of clocks since enabling. These clock counts verify if there is a PLL function.

Miscellaneous Timing

Read/Write MMIO_base + 0x000068

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Ex				Reserved		Xc R/W		Reserved			S_{SM1} R/W	S_{M1} R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			S_{SM0} R/W	S_{M0} R/W				Deb R/W	A R/W	Divider R/W		U R/W	Delay R/W		

Bit(s)	Name	Description
31:28	Ex	Extend the bus holding when the SM501 is a host bus master to access system SDRAM. The extension period starts to count down when the SM501 gets an asserted bus acknowledge signal from the CPU.

0000	No Extension	1000	Extend 128 Host Clock
0001	Extend 16 Host Clock	1001	Extend 144 Host Clock
0010	Extend 32 Host Clock	1010	Extend 160 Host Clock
0011	Extend 48 Host Clock	1011	Extend 176 Host Clock
0100	Extend 64 Host Clock	1100	Extend 192 Host Clock
0101	Extend 80 Host Clock	1101	Extend 208 Host Clock
0110	Extend 96 Host Clock	1110	Extend 224 Host Clock
0111	Extend 112 Host Clock	1111	Extend 240 Host Clock

27:26	Reserved	These bits are reserved.
25:24	Xc	XScale Clock Input Source. 00: From clock generated by internal PLL. 01: From HCLK pin. 1x: From GPIO30 pin. **Note:** Bit 25 is strapped by GPIO31, and bit 24 is strapped by GPIO29.
23:21	Us	USB host over current detection. 0: Disable over current detection. 1: Enable over current detection.
22:21	Reserved	These bits are reserved.
20	S_{SM1}	System SDRAM Clock Select for PW Mode 1. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 below).
19:16	S_{M1}	System SDRAM Clock Frequency Divider for PW Mode 1.

0000	÷ 1	0100	÷ 16	1000	÷ 3	1100	÷ 48
0001	÷ 2	0101	÷ 32	1001	÷ 6	1101	÷ 96
0010	÷ 4	0110	÷ 64	1010	÷ 12	1110	÷ 192
0011	÷ 8	0111	÷ 128	1011	÷ 24	1111	÷ 384

<table>
<tr><th>Bit(s)</th><th>Name</th><th>Description</th></tr>
<tr><td>15:13</td><td>Reserved.</td><td>These bits are reserved.</td></tr>
<tr><td>12</td><td>S_{SM0}</td><td>System SDRAM Clock Select for PW Mode 0.
0: 288 MHz.
1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 below).</td></tr>
<tr><td>11:8</td><td>S_{M0}</td><td>System SDRAM Clock Frequency Divider for PW Mode 0.
<table>
<tr><td>0000</td><td>÷ 1</td><td>0100</td><td>÷ 16</td><td>1000</td><td>÷ 3</td><td>1100</td><td>÷ 48</td></tr>
<tr><td>0001</td><td>÷ 2</td><td>0101</td><td>÷ 32</td><td>1001</td><td>÷ 6</td><td>1101</td><td>÷ 96</td></tr>
<tr><td>0010</td><td>÷ 4</td><td>0110</td><td>÷ 64</td><td>1010</td><td>÷ 12</td><td>1110</td><td>÷ 192</td></tr>
<tr><td>0011</td><td>÷ 8</td><td>0111</td><td>÷ 128</td><td>1011</td><td>÷ 24</td><td>1111</td><td>÷ 384</td></tr>
</table></td></tr>
<tr><td>7</td><td>Deb</td><td>96 MHz PLL Debug Control.
0: Check input reference frequency.
1: Check output frequency.</td></tr>
<tr><td>6</td><td>A</td><td>ACPI Control.
0: No ACPI control.
1: ACPI control.</td></tr>
<tr><td>5:4</td><td>Divider</td><td>Output Frequency Selection for Second PLL.
00: 336 MHz.
01: 288 MHz.
10: 240 MHz.
11: 192 MHz.</td></tr>
<tr><td>3</td><td>U</td><td>USB Host Mode.
0: Normal mode.
1: Simulation mode – don’t generate 1ms pulse.</td></tr>
<tr><td>2:0</td><td>Delay</td><td>Delay time to latch read data for external SDRAM memory controller.
000: No delay.
001: Delay ½ns.
010: Delay 1ns.
011: Delay 1½ns.
100: Delay 2ns.
101: Delay 2½ns.</td></tr>
</table>

Current System SDRAM Clock

Read MMIO_base + 0x00006C

Power-on Default 0x00000009

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

Bit(s)	Name	Description
31:5	Reserved	These bits are reserved.
4	Ss	System SDRAM Clock Select. 0: 288 MHz. 1: 336 MHz/288 MHz/240 MHz/192 MHz (see bits 5:4 in the Miscellaneous Timing register at offset 0x68).
3:0	S	System SDRAM Clock Frequency Divider.

0000	÷ 1	0100	÷ 16	1000	÷ 3	1100	÷ 48
0001	÷ 2	0101	÷ 32	1001	÷ 6	1101	÷ 96
0010	÷ 4	0110	÷ 64	1010	÷ 12	1110	÷ 192
0011	÷ 8	0111	÷ 128	1011	÷ 24	1111	÷ 384

3 PCI Configuration Space

Register Descriptions

The PCI specification defines the configuration space for auto-configuration (plug-and-play), and device and memory relocation.

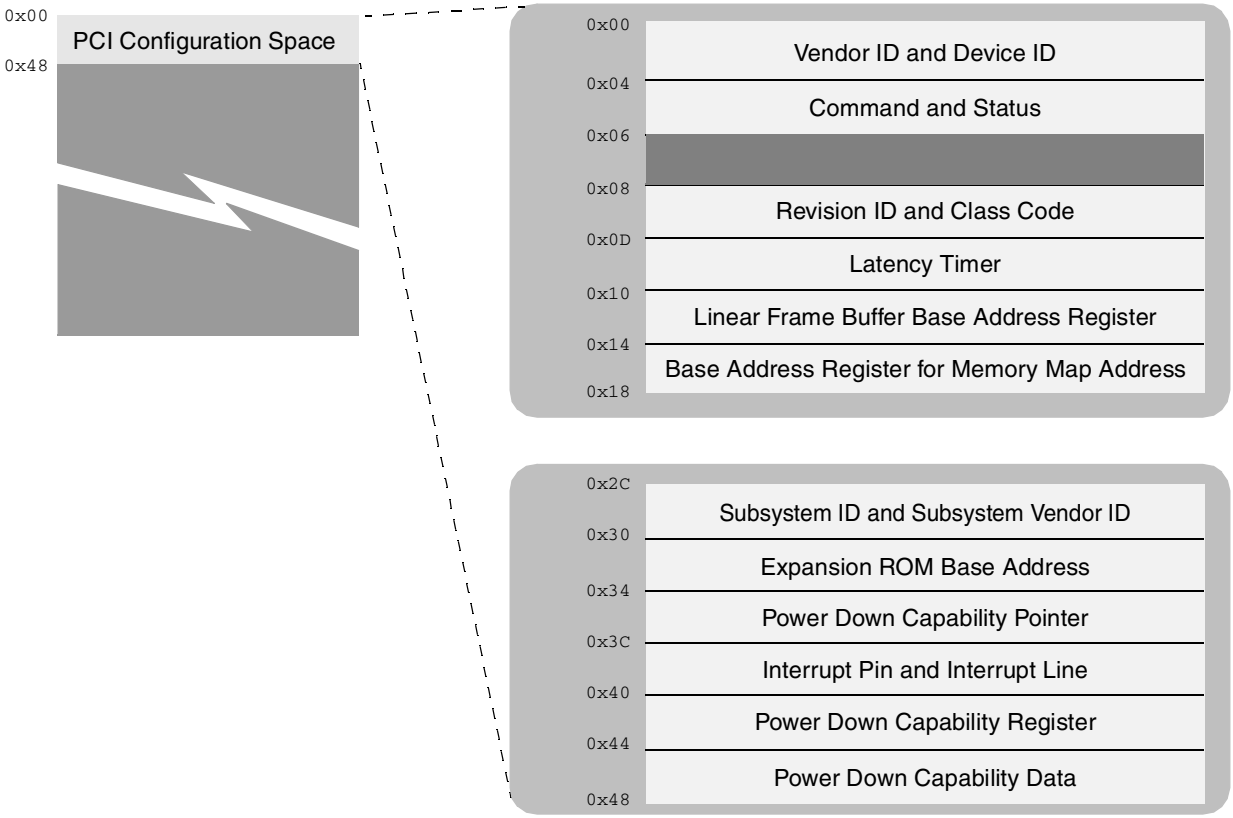
Table 3-1 summarizes the PCI Configuration Space Registers.

Table 3-1: PCI Configuration Space Register Summary

Address	Type	Width	Reset Value	Register Name
0x00	R	32	0x0501126F	CSR00: Vendor ID and Device ID
0x04	R/W	32	0x23000000	CSR04: Command and Status
0x08	R	32	0x380000A0	CSR08: Revision ID and Class Code
0x0D	R	8	0x00	CSR0C: Latency Timer
0x10	R/W	32	0x00000000	CSR10: Linear Frame Buffer Base Address Register
0x14	R/W	32	0x00000000	CSR14: Base Address Register for Memory Map Address
0x2C	R	32	0x00000000	CSR2C: Subsystem ID and Subsystem Vendor ID
0x30	R/W	32	0x00000000	CSR30: Expansion ROM Base Address
0x34	R	32	0x00000040	CSR34: Power Down Capability Pointer
0x3C	R/W	32	0x00000000	CSR3C: Interrupt Pin and Interrupt Line
0x40	R	32	0x06010001	CSR40: Power Down Capability Register
0x44	R/W	32	0x00000000	CSR44: Power Down Capability Data

Figure 3-1 lists the registers available in the PCI Configuration register space.

Figure 3-1: PCI Configuration Register Space



CSR00: Vendor ID and Device ID

Read Address `0x00`

Power-on Default `0x0501126F`

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

This register specifies the device and vendor IDs.

Bit(s)	Name	Description
31:16	Device ID	These bits are hardwired to `0x0501` to identify the SM501 device.
15:0	Vendor ID	These bits are hardwired to `0x126F` to identify the vendor as Silicon Motion®, Inc.

CSR04: Command and Status

Read/Write Address 0x04
Power-on Default 0x02300000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DPE R	Res		DTA R	Res	DEVSEL R		Res			66C R	NCD R	Res			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res										PSE R/W	MWR R/W	Res	PBM R/W	MS R/W	IO R

This register controls which types of PCI command cycles are supported by the SM501.

Note: Reserved bits are read only.

Bit(s)	Name	Description
31	DPE	Data Parity Error Detected. 0: Correct. 1: Error detected.
30:29	Res	These bits are reserved.
28	DTA	Received Target Abort. 0: Correct. 1: Abort detected.
27	Res	This bit is reserved.
26:25	DEVSEL	Timing Select Medium.
24:22	Res	These bits are reserved.
21	66C	66 MHz Capable.
20	NCD	New Capability Definition.
19:6	Res	These bits are reserved.
5	PSE	Palette Snooping Enable. 0: Disable. 1: Enable.
4	MWR	Memory Write and Invalidate Enable. 0: Disable. 1: Enable.
3	Res	This bit is reserved.
2	PBM	PCI Bus Master Enable.
1	MS	Memory Space Access Enable. 0: Disable. 1: Enable.
0	IO	I/O Space Access Enable. 0: Disable.

CSR08: Revision ID and Class Code

Read Address 0x08

Power-on Default 0x038000A0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Base Class Code R								Subclass Code R							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Register Level Programming Interface R								Revision ID R							

This register specifies the silicon revision ID and the Class Code that the silicon supports.

Bit(s)	Name	Description
31:24	Base Class Code	0x03 = for Video Controller
23:16	Subclass Code	0x80 = Other Display Controller
15:8	Register Level Programming Interface	0x00 = hardwired setting
7:0	Revision ID	For example, 0xA0 = revision A; 0xB0 = revision B

CSR0C: Latency Timer

Read Address 0x0D

Power-on Default 0x00

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

This register specifies the latency timer that the SM501 supports for burst master mode.

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:8	LT	Latency Timer. The default for this field is 0x00.
7:0	Reserved	These bits are reserved.

CSR10: Linear Frame Buffer Base Address Register

Read/Write Address 0x10
Power-on Default 0x0000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Linear Addressing Memory Base R/W/R											Linear Frame Buffer Base Address R				
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Linear Frame Buffer Base Address R															MB R

This register specifies the PCI configuration space for address relocation

Note: Reserved bits are read only.

Bit(s)	Name	Description
31:21	Linear Address Memory Base Address	Memory segment allocated within 64 MB boundary. • If 2 MB: **Bits 26:21** = FBA (Read/Write) • If 4 MB: **Bits 26:22** = FBA (Read/Write) **Bit 21** = 0b • If 8 MB: **Bits 26:23** = FBA (Read/Write) **Bits 22:21** = 00b (Read Only) • If 16 MB: **Bits 26:24** = FBA (Read/Write) **Bits 23:21** = 000b (Read Only) • If 32 MB: **Bit 26** = FBA (Read/Write) **Bits 24:21** = 0000b (Read Only) • If 64 MB: **Bit 26** = FBA (Read/Write) **Bits 25:21** = 00000b (Read Only)
20:1	Linear Frame Buffer Base Address	The default for this read-only field is 0x000000.
0	MB	Memory Base Read. The default for this bit is 0.

CSR14: Base Address Register for Memory Map Address

Read/Write Address 0x14
Power-on Default 0x0000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Memory Map Address Base Address R/W/R											Memory Map Address Base Address R				
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Memory Map Address Base Address R															MB R

This register specifies the PCI configuration space for address relocation.

Note: Reserved bits are read only.

Bit(s)	Name	Description
31:21	FBA	Memory Map Address Base Address. • If One Endian: **Bits [31:21]** = FBA (Read/Write) • If Big and Small Endian: **Bit [31:22]** = FBA (Read/Write) **Bit [21]** = 0b (Read Only)
20:1	ABA	Memory Map Address Base Address. The default for this read-only field is 0x000000.
0	MB	Memory Base. The default for this read-only bit is 0.

CSR2C: Subsystem ID and Subsystem Vendor ID

Read Address 0x2C

Power-on Default 0x00000000

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

This register specifies both the Subsystem device ID and the Subsystem Vendor ID.

Bit(s)	Name	Description
31:16	Subsystem ID	This System ID is written by the system BIOS during POST.
15:0	Subsystem Vendor ID	This field contains the Subsystem Vendor ID.

CSR30: Expansion ROM Base Address

Read/Write Address 0x30

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ROM Base Address R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved															BIOS R/W

This register specifies the expansion ROM base address.

Note: Reserved bits are read only.

Bit(s)	Name	Description
31:16	ROM Base Address	Memory segment allocated for BIOS ROM in 64KB boundary [15:0].
15:1	Reserved	These bits are reserved.
0	BIOS Address Decode Enable	This bit is valid only if memory space access is enabled (CSR04 bit 1 = 1). 0: Disable. 1: Enable.

CSR34: Power Down Capability Pointer

Read Address 0x34

Power-on Default 0x00000040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Power Down Capability Pointer															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Power Down Capability Pointer															

This register contains the address where PCI power down management registers are located.

Bit(s)	Name	Description
31:0	Power Down Capability Pointer	The Capability pointer contains the address where the PCI Power Down Management Register is located.

CSR3C: Interrupt Pin and Interrupt Line

Read/Write Address 0x3C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Interrupt Pin R								Interrupt Line R/W							

This register specifies the PCI interrupt pin and interrupt line.

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:8	Interrupt Pin	
7:0	Interrupt Line	

CSR40: Power Down Capability Register

Read Address 0x40

Power-on Default 0x06010001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PCI Power Down Management Capability (0x0601) R															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Next Capability Pointer Link List R								PCI Power Down Mgmt Capability (0x01) R							

This register contains the address for PCI power-down management capabilities.

Bit(s)	Name	Description
31:16	PCI Power Down Management Capability	Offset 2. This field is hardwired to 0x0601.
15:8	No More Extra Capability Pointer	This field is hardwired to 0x00.
7:0	PCI Power Down Management Capability ID	Offset 0. This field is hardwired to 0x01.

CSR44: Power Down Capability Data

Read/Write Address 0x44

Power-on Default 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Data R								Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PCI Power Down Mgmt Control/Status R/W														PDS R/W	

This register contains the address for PCI power-down management Control, Status and Data.

Bit(s)	Name	Description
31:24	Data	Offset 7. This data field is read-only.
23:16	Reserved	Offset 6.
15:2	PCI Power Down Management Control/Status	Offset 4.
1:0	PDS	Power Down Management Control and Status. 00: Power Down Management State D0. 01: Power Down Management State D1. 10: Power Down Management State D2. 11: Power Down Management State D3.

4 Drawing Engine

Functional Overview

The SM501's Drawing Engine is designed to accelerate Microsoft's DirectDraw and Direct3D applications. The engine contains a 3-operand ALU with 256 raster operations, source and destination FIFOs, as well as a host data FIFO. The drawing engine pipeline allows single cycle operations and runs at the memory clock speed.

The Drawing Engine includes several key functions to achieve the high GUI performance. The device supports color expansion with packed mono font, color pattern fill, host BLT, stretch BLT, short stroke, line draw, and others. Dedicated pathways are designed to transfer data between the host interface (HIF) bus and the Drawing Engine, and memory interface (MIF) bus and the Drawing Engine. In addition, the Drawing Engine supports rotation BIBLT for any block size, and automatic self activate rotation BLIT. This feature allows conversion between landscape and portrait display without the need for special software drivers.

Programmer's Model

The Drawing Engine supports various drawing functions, including Bresenham line draw, short stroke line draw, BITBLT, rectangle fill, HOSTBLT, Rotation Blit, and others. Hardware clipping is supported by 4 registers, DPR2C-DPR32, which define a rectangular clipping area.

The drawing engine supports two types of addressing formats for its source and destination locations. In XY addressing mode, the location is specified in X-Y coordinates, where the upper left corner of the screen is defined to be (0,0). In linear addressing mode, the location is specified based on its position in the display memory sequentially from the first pixel of the visible data. The addressing mode is set by the Addressing field of the 2D Stretch & Format register. The Command field of the 2D Control register selects other drawing functions, for example, Bresenham line draw, host write and short stroke.

Register Descriptions

All Drawing Engine control registers can be accessed via memory mapping. The address is at DP_Base + XXXh, where DP_Base is at PCI graphics base address + 4MB + 32K. Figure 4-1 shows how this 64kB region in the MMIO space is laid out. It controls the Drawing Engine registers.

Figure 4-1: Drawing Engine Register Space

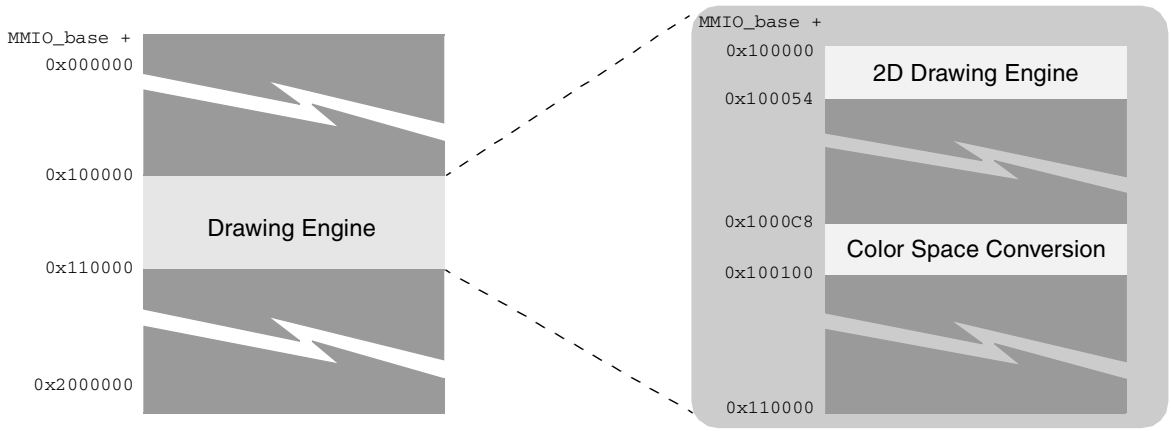


Table 4-1 summarizes the Drawing Engine registers.

Table 4-1: Drawing Engine Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Register Name
0x100000	R/W	32	0x00000000	2D Source
0x100004	R/W	32	0x00000000	2D Destination
0x100008	R/W	32	0x00000000	2D Dimension
0x10000C	R/W	32	0x00000000	2D Control
0x100010	R/W	32	0x00000000	2D Pitch
0x100014	R/W	32	0x00000000	2D Foreground
0x100018	R/W	32	0x00000000	2D Background
0x10001C	R/W	32	0x00000000	2D Stretch & Format
0x100020	R/W	32	0x00000000	2D Color Compare
0x100024	R/W	32	0x00000000	2D Color Compare Mask
0x100028	R/W	32	0x00000000	2D Mask
0x10002C	R/W	32	0x00000000	2D Clip TL
0x100030	R/W	32	0x00000000	2D Clip BR
0x100034	R/W	32	0x00000000	2D Mono Pattern Low
0x100038	R/W	32	0x00000000	2D Mono Pattern High
0x10003C	R/W	32	0x00000000	2D Window Width
0x100040	R/W	32	0x00000000	2D Source Base
0x100044	R/W	32	0x00000000	2D Destination Base
0x100048	R/W	32	0x00000000	2D Alpha
0x10004C	R/W	32	0x00000000	2D Wrap
0x100050	R/W	32	0x00000000	2D Status
0x1000C8	R/W	32	0x00000000	CSC Y Source Base
0x1000CC	R/W	32	0x00000000	CSC Constants
0x1000D0	R/W	32	0x00000000	CSC Y Source X

Table 4-1: Drawing Engine Register Summary (Continued)

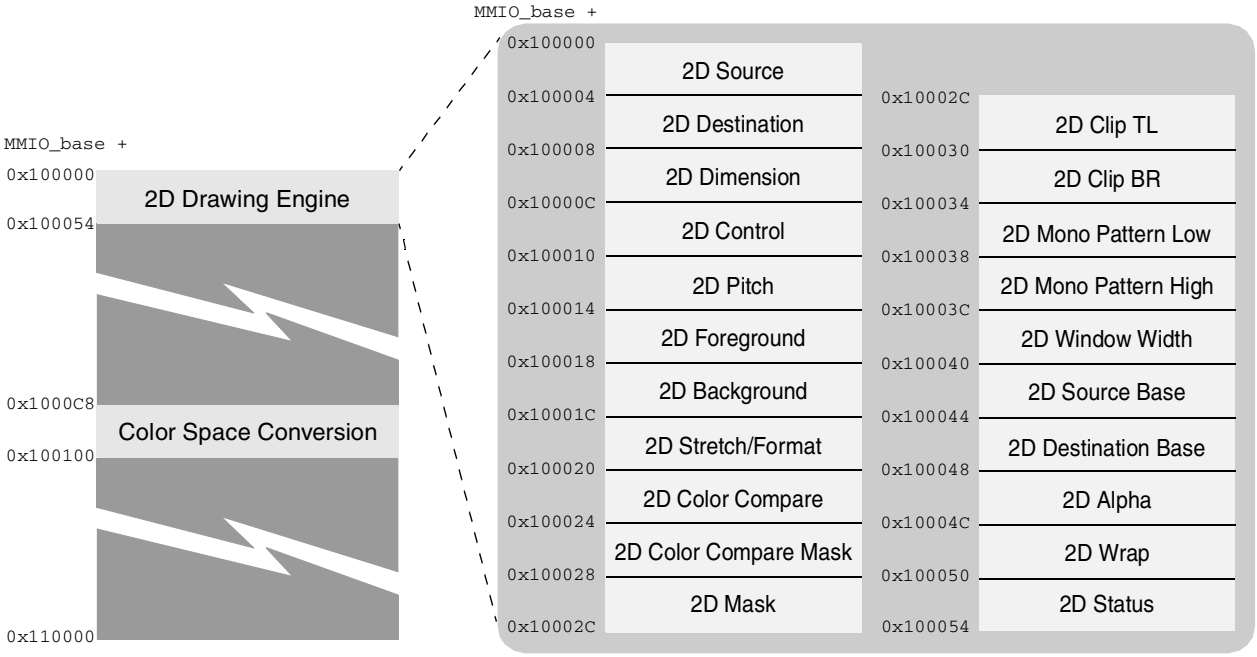
Offset from MMIO_base[1]	Type	Width	Reset Value	Register Name
0x1000D4	R/W	32	0x00000000	CSC Y Source Y
0x1000D8	R/W	32	0x00000000	CSC U Source Base
0x1000DC	R/W	32	0x00000000	CSC V Source Base
0x1000E0	R/W	32	0x00000000	CSC Source Dimension
0x1000E4	R/W	32	0x00000000	CSC Source Pitch
0x1000E8	R/W	32	0x00000000	CSC Destination
0x1000EC	R/W	32	0x00000000	CSC Destination Dimension
0x1000F0	R/W	32	0x00000000	CSC Destination Pitch
0x1000F4	R/W	32	0x00000000	CSC Scale Factor
0x1000F8	R/W	32	0x00000000	CSC Destination Base
0x1000FC	R/W	32	0x00000000	CSC Control

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.

2D Drawing Engine Registers

The 2D Drawing Engine register space is shown in Figure 4-2.

Figure 4-2: 2D Drawing Register Space



2D Source

Read/Write MMIO_base + 0x100000

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
W R/W	Res	X_K1 R/W													
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Y_K2 R/W															

Bit(s)	Name	Description
31	W	Wrap Control. 0: Disable. 1: Enable.
30	Res	This bit is reserved.

Bit(s)	Name	Description
29:16	X_K1	In XY addressing mode, the 12-bit X-coordinate of source (bits 27:16). In linear addressing mode, the lower 12-bits of the source address (bits 27:16). In host write mode, the 5-bit mono source for alignment (bits 20:16) In Bresenham line drawing mode, the 14-bit K1 constant for line drawing (bits 29:16): K1 = 2 * min(|dx|, |dy|).
15:0	Y_K2	In XY addressing mode, the 12-bit Y-coordinate of source (bits 11:0). In linear addressing mode, the higher 12-bits of the source address (bits 11:0). In host write mode, this field is not used. In Bresenham line drawing mode, the 14-bit K2 constant for line drawing (bits 13:0): K2 = 2 * (min(|dx|, |dy|) – max(|dx|, |dy|).

2D Destination

Read/Write MMIO_base + 0x100004

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31	W	Wrap Control. 0: Disable. 1: Enable.
30:29	Reserved	These bits are reserved.
28:16	X	In XY addressing mode, the 12-bit X-coordinate of source (bits 27:16). In linear addressing mode, the lower 12-bits of the source address (bits 27:16). In Bresenham mode, the vector X start address (bits 27:16)
15:0	Y	In XY addressing mode, the 12-bit Y-coordinate of source (bits 11:0). In linear addressing mode, the higher 12-bits of the source address (bits 11:0). In Bresenham mode, the vector Y start address (bits 11:0)

2D Dimension

Read/Write MMIO_base + 0x100008

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:29	Reserved	These bits are reserved.
28:16	X_VL	In XY addressing mode, the X dimension in pixels. In Bresenham mode, the vector length. In short stroke mode, horizontal length
15:0	Y_ET	In XY addressing mode, the Y dimension in pixels. In Bresenham mode, the vector error term given by: ET = 2 * min(|dx|, |dy|) – max(|dx|, |dy|) if start X > end X, or ET = 2 * min(|dx|, |dy|) – max(|dx|, |dy|) – 1 if start X <= end X. In short stroke mode, the non-horizontal length.

2D Control

Read/Write MMIO_base + 0x10000C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	P R/W	U R/W	Q R/W	D R/W	M R/W	X R/W	Y R/W	St R/W	H R/W	LP R/W	Command R/W				
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
R R/W	R2 R/W	Mono R/W		RR R/W	TM R/W	TS R/W	T R/W	ROP R/W							

Bit(s)	Name	Description
31	S	Drawing Engine Status. 0: Stop. 1: Start.
30	P	Pattern Select. 0: Monochrome. 1: Color.

<table>
<tr><th>Bit(s)</th><th>Name</th><th>Description</th></tr>
<tr><td>29</td><td>U</td><td>Update Destination X after Operation Control.
0: Disable.
1: Enable.</td></tr>
<tr><td>28</td><td>Q</td><td>Quick Start Control. Quick start will start the drawing engine after the X field in the 2D Dimension register has been written to.
0: Disable.
1: Enable.</td></tr>
<tr><td>27</td><td>D</td><td>Direction Control for Operation.
0: Left to right.
1: Right to left.</td></tr>
<tr><td>26</td><td>M</td><td>Major Axis for Line Drawing.
0: X axis.
1: Y axis.</td></tr>
<tr><td>25</td><td>X</td><td>X Step Control for Line Drawing.
0: Positive.
1: Negative.</td></tr>
<tr><td>24</td><td>Y</td><td>Y Step Control for Line Drawing.
0: Positive.
1: Negative.</td></tr>
<tr><td>23</td><td>St</td><td>Stretch in Y Direction Control.
0: Disable.
1: Enable.</td></tr>
<tr><td>22</td><td>H</td><td>Host BitBlt Select.
0: Color.
1: Monochrome.</td></tr>
<tr><td>21</td><td>LP</td><td>Draw Last Pixel Control for Line Drawing.
0: Don’t draw last pixel.
1: Draw last pixel.</td></tr>
<tr><td>20:16</td><td>Command</td><td>Command Code.
<table>
<tr><td>00000</td><td>BitBlt</td><td>00111</td><td>Line Draw</td></tr>
<tr><td>00001</td><td>Rectangle Fill</td><td>01000</td><td>Host Write</td></tr>
<tr><td>00010</td><td>De-Tile</td><td>01001</td><td>Host Read</td></tr>
<tr><td>00011</td><td>Trapezoid Fill</td><td>01010</td><td>Host Write bottom-to-top</td></tr>
<tr><td>00100</td><td>Alpha Blend</td><td>01011</td><td>Rotate</td></tr>
<tr><td>00101</td><td>RLE Strip</td><td>01100</td><td>Font</td></tr>
<tr><td>00110</td><td>Short Stroke</td><td>01111</td><td>Texture Load</td></tr>
</table></td></tr>
<tr><td>15</td><td>R</td><td>ROP Control.
0: ROP3.
1: ROP2.</td></tr>
<tr><td>14</td><td>R2</td><td>ROP2 Control.
0: ROP2 source is bitmap.
1: ROP2 source is pattern.</td></tr>
<tr><td>13:12</td><td>Mono</td><td>Monochrome Data Pack Control.
00: Not packed.
01: Packed at 8-bit.
10: Packed at 16-bit.
11: Packed at 32-bit.</td></tr>
</table>

Bit(s)	Name	Description
11	RR	Repeat Rotation Control. Only valid when *Command* is 01011 (Rotate). When enabled, the drawing engine is started again at every vertical sync. 0: Disable. 1: Enable.
10	TM	Transparency Match Select. 0: Matching pixel is opaque. 1: Matching pixel is transparent.
9	TS	Transparency Select. 0: Transparency is controlled by source. 1: Transparency is controlled by destination.
8	T	Transparency Control. 0: Disabled. 1: Enabled.
7:0	ROP	ROP2 or ROP3 code (see tables below).

Binary Raster Operations (ROP2)

Each raster-operation code represents a Boolean operation in which the values of the pixels in the selected pen and the destination bitmap are combined. The operands used in these operations are:

- P = selected pen
- D = destination bitmap

		bit[2:0]							
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Bit3	0	0	~(D+S)	D*~S	~S	~D*S	~D	D^S	(~D*S)
	1	D*S	(~D^S)	D	D+~S	S	~D+S	D+S	1

Ternary Raster Operations (ROP3)

Each raster-operation code represents a Boolean operation in which the values of the pixels in the source, the selected brush, and the destination are combined. The operands and operands are:

- D = destination bitmap
- P = selected brush (pattern)
- S = source bitmap
- a = bitwsie AND
- n = bitwise NOT (inverve)
- o = bitwise OR
- x = bitwise XOR (exclusive OR)

All Boolean operations are presented in reverse Polish notation. For example, the following operation replaces the pixel values in the destination bitmap with a combination of the pixel values of the source and brush:

PSo = (P+S)

The following operation combines the values of the pixels in the source and brush with the pixel values of the destination bitmap (there are alternative spellings of the same function, so although a particular spelling may not be in the list, an equivalent form would be):

DPSoo = (P+S) + D

VoyagerGX supports all the 256 operations. However, Pattern must be monochrome.

Table 4-2 lists all the possible ROP3 operations.

Table 4-2: ROP3 Operations

		Bits [7:4]							
		0	1	2	3	4	5	6	7
bit [3:0]	0	0	PDSona	DPSnaa	PSna	PSDnaa	PDna	PDSxa	PDSana
	1	DPSoon	DSon	SDPxon	SDPnaon	DPSxon	DSPnaon	DSPDSaoxxn	SSDxPDxaxn
	2	DPSona	SDPxnon	DSna	SDPSoox	SDxPDxa	DPSDaox	DSPDoax	SDPSxox
	3	PSon	SDPaon	SPDnaon	Sn	SPDSanaxn	SPDSxaxn	SDPnox	SDPnoan
	4	SDPona	DPSxnon	SPxDSxa	SPDSaox	SDna	DPSonon	SDPSoax	DSPDxox
	5	DPon	DPSaon	PDSPanaxn	SPDSxnox	DPSnaon	Dn	DSPnox	DSPnoan
	6	PDSxnon	PSDPSanaxx	SDPSaox	SDPox	DSPDaox	DPSox	DSx	SDPSnaox
	7	PDSaon	SSPxDSxaxn	SDPSxnox	SDPoan	PSDPxaxn	DPSoan	SDPSonox	DSan
	8	SDPnaa	SPxPDxa	DPSxa	PSDPoax	SDPxa	PDSPoax	DSPDSonoxxn	PDSax
	9	PDSxon	SDPSanaxn	PSDPSaoxxn	SPDnox	PDSPDaoxxn	DPSnox	PDSxxn	DSPDSoaxxn
	A	DPna	PDSPaox	DPSana	SPDSxox	DPSDoax	DPx	DPSax	DPSDnoax
	B	PSDnaon	SDPSxaxn	SSPxPDxaxn	SPDnoan	PDSnox	DPSDonox	PSDPSoaxxn	SDPxnan
	C	SPna	PSDPaox	SPDSoax	PSx	SDPana	DPSDxox	SDPax	SPDSnoax
	D	PDSnaon	DSPDxaxn	PSDnox	SPDSonox	SSPxDSxoxn	DPSnoan	PDSPDoaxxn	DPSxnan
	E	PDSonon	PDSox	PSDPxox	SPDSnaox	PDSPxox	DPSDnaox	SDPSnoax	SPxDSxo
	F	Pn	PDSoan	PSDnoan	PSan	PDSnoan	DPan	PDSxnan	DPSaan

		Bits [7:4]							
		8	9	A	B	C	D	E	F
bit [3:0]	0	DPSaa	PDSxna	DPa	PDSnoa	PSa	PSDnoa	PDSoa	P
	1	SPxDSxon	SDPSnoaxn	PDSPnaoxn	PDSPxoxn	SPDSnaoxn	PSDPxoxn	PDSoxn	PDSono
	2	DPSxna	DPSDPoaxx	DPSnoa	SSPxDSxox	SPDSonoxn	PDSnax	DSPDxax	PDSnao
	3	SPDSnoaxn	SPDaxn	DPSDxoxn	SDPanan	PSxn	SPDSoaxn	PSDPaoxn	PSno
	4	SDPxna	PSDPSoaxx	PDSPonoxn	PSDnax	SPDnoa	SSPxPDxax	SDPSxax	PSDnao
	5	PDSPnoaxn	DPSaxn	PDxn	DPSDoaxn	SPDSxoxn	DPSanan	PDSPaoxn	PDno
	6	DSPDSoaxx	DPSxx	DSPnax	DPSDPaoxx	SDPnax	PSDPSaoxx	SDPSanax	PDSxo
	7	PDSaxn	PSDPSonoxx	PDSPoaxn	SDPxan	PSDPoaxn	DPSxan	SPxPDxan	PDSano
	8	DSa	SDPSonoxn	DPSoa	PSDPxax	SDPoa	PDSPxax	SSPxDSxax	PDSao
	9	SDPSnaoxn	DSxn	DPSoxn	DSPDaoxn	SPDoxn	SDPSaoxn	DSPDSanaxxn	PDSxno
	A	DSPnoa	DPSnax	D	DPSnao	DPSDxax	DPSDanax	DPSao	DPo
	B	DSPDxoxn	SDPSoaxn	DPSono	DSno	SPDSaoxn	SPxDSxan	DPSxno	DPSnoo
	C	SDPnoa	SPDnax	SPDSxax	SPDSanax	S	SPDnao	SDPao	PSo
	D	SDPSxoxn	DSPDoaxn	DPSDaoxn	SDxPDxan	SDPono	SDno	SDPxno	PSDnoo
	E	SSDxPDxax	DSPDSaoxx	DSPnao	DPSxo	SDPnao	SDPxo	DSo	DPSoo
	F	PDSanan	PDSxan	DPno	DPSano	SPno	SDPano	SDPnoo	1

Rotate Command

For the Rotate command, the *X* and *Y* bits determine the rotation angle. For the Short Stroke command, the D, M, X, and Y bits determine the direction of the vector.

X	Y	Rotation Direction
0	0	0 degrees
0	1	270 degrees
1	0	90 degrees
1	1	180 degrees

D	M	X	Y	Vector Direction
0	0	0	0	225 degrees
0	0	0	1	135 degrees
0	0	1	0	315 degrees
0	0	1	1	45 degrees
0	1	0	0	270 degrees
0	1	0	1	90 degrees
1	0	0	0	180 degrees
1	0	1	0	0 degrees

2D Pitch

Read/Write MMIO_base + 0x100010

Power-on Default 0x00000000

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

Bit(s)	Name	Destination
31:29	Reserved	These bits are reserved.
28:16	Destination	Pitch of destination specified in pixels.
15:13	Reserved	These bits are reserved.
12:0	Source	Pitch of source specified in pixels.

2D Foreground

Read/Write MMIO_base + 0x100014

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:0	Foreground	

Bits Per Pixel	Foreground Color
8	Index color.
16	RGB565 color.
32	RGBx888 color.

2D Background

Read/Write MMIO_base + 0x100018

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:0	Background	

Bits Per Pixel	Background Color
8	Index color.
16	RGB565 color.
32	RGBx888 color.

In monochrome transparency, the *Background* must be programmed with the invert of the *Foreground* pixels in the *2D Foreground* register.

2D Stretch & Format

Read/Write MMIO_base + 0x10001C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	XY R/W	Y R/W			Res	X R/W			Res	Format R/W		Addressing R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res				Height R/W											

Bit(s)	Name	Description
31	Res	This bit is reserved.
30	XY	Pattern XY Select. 0: Only use *X* and *Y* fields in linear mode. 1: Use *X* and *Y* fields in XY mode.
29:27	Y	Pattern Y Origin. This field is only valid in linear mode (*Addressing* = 1111) or when *XY* is enabled.
26	Res	This bit is reserved.
25:23	X	Pattern X Origin. This field is only valid in linear mode (*Addressing* = 1111) or when *XY* is enabled.
22	Res	This bit is reserved.
21:20	Format	Pixel Format. 00: 8-bits per pixel. 01: 16-bits per pixel. 10: 32-bits per pixel.
19:16	Addressing	Addressing Mode. 0000: XY mode. 1111: Linear mode.
15:12	Res	These bits are reserved.
11:0	Height.	Source height when stretch is enabled.

2D Color Compare

Read/Write MMIO_base + 0x100020

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:0	Color	**Bits Per Pixel** / **Color Compare** 8 / Index color. 16 / RGB565 color. 32 / RGB888 color.

In monochrome transparency, the *Color* must be programmed with the same value as the *Foreground* pixels in the *2D Foreground* register.

2D Color Compare Mask

Read/Write MMIO_base + 0x100024

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:0	Mask	Mask Bits for Color Compare. 0: Color compare always matches. 1: Color compare only matches when bits are equal.

2D Mask

Read/Write MMIO_base + 0x100028

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Byte	Byte mask for each of the 16 bytes on the 128-bit memory bus. 0: Disable write. 1: Enable write.
15:0	Bit	Bit mask for 8- and 16-bits per pixel modes. 0: Disable write. 1: Enable write.

2D Clip TL

Read/Write MMIO_base + 0x10002C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Top R/W															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		E R/W	S R/W	Left R/W											

Bit(s)	Name	Description
31:16	Top	Top Coordinate of Clipping Rectangle.
15:14	Reserved	These bits are reserved.
13	E	Clipping Control. 0: Disable. 1: Enable.
12	S	Clipping Select Control. 0: Write outside clipping rectangle disabled. 1: Write inside clipping rectangle disabled.
11:0	Left	Left Coordinate of Clipping Rectangle.

2D Clip BR

Read/Write MMIO_base + 0x100030

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Bottom	Bottom Coordinate of Clipping Rectangle.
15:13	Reserved	These bits are reserved.
12:0	Right	Right Coordinate of Clipping Rectangle.

2D Mono Pattern Low

Read/Write MMIO_base + 0x100034

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:0	Pattern	Bits [31:0] of monochrome pattern.

2D Mono Pattern High

Read/Write MMIO_base + 0x100038

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:0	Pattern	Bits [63:32] of monochrome pattern.

2D Window Width

Read/Write MMIO_base + 0x10003C

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:29	Reserved	These bits are reserved.
28:16	Destination	Width of destination window specified in pixels.
15:13	Reserved	These bits are reserved.
12:0	Source	Width of source window specified in pixels.

2D Source Base

Read/Write MMIO_base + 0x100040

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Destination
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of source window with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

2D Destination Base

Read/Write MMIO_base + 0x100044

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.

Bit(s)	Name	Description
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of destination window with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

2D Alpha

Read/Write MMIO_base + 0x100048

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:8	Reserved	These bits are reserved.
7:0	Alpha	Alpha Value for Alpha Blend.

2D Wrap - Width and Height

Read/Write MMIO_base + 0x10004C

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Width	Horizontal pitch (H_Pitch) in pixels.
15:0	Height	Vertical pitch (V_Pitch) in lines.

2D Status

Read/Write MMIO_base + 0x100050

Power-on Default 0x00000000

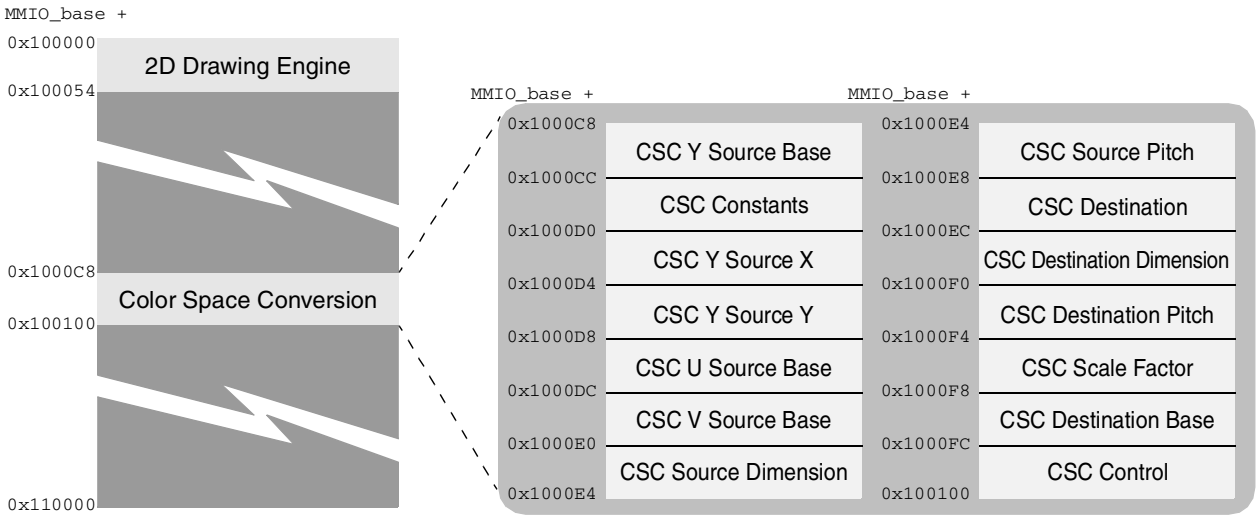
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved														CSC R/W	2D R/W

Bit(s)	Name	Description
31:2	Reserved	These bits are reserved.
1	CSC	Color Space Conversion Interrupt Status. Write a 0 into this field to clear the interrupt status. 0: CSC not active or job not done. 1: CSC interrupt.
0	2D	2D Engine Interrupt Status. Write a 0 into this field to clear the interrupt status. 0: 2D not active or job not done. 1: 2D interrupt.

Color Space Conversion Registers

The Color Space Conversion register space is shown in Figure 4-3.

Figure 4-3: Color Space Conversion Register Space



CSC Y Source Base (UV Source Base in 420i)

Read/Write MMIO_base + 0x1000C8

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory Address of Source Y-plane with 128-bit Alignment.
3:0	0000	These bits are hardwired to zeros.

CSC Constants

Read/Write MMIO_base + 0x1000CC

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:24	Y	Y Conversion Constant (luminosity).
23:16	R	Red Conversion Constant.
15:8	G	Green Conversion Constant.
7:0	B	Blue Conversion Constant.

CSC Y Source X

Read/Write MMIO_base + 0x1000D0

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	X_I	Integer Part of Starting X-coordinate into Y-plane.
15:3	X_F	Fractional Part of Starting X-coordinate into Y-plane.
2:0	Reserved	These bits are reserved.

CSC Y Source Y

Read/Write MMIO_base + 0x1000D4

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	Y_I	Integer Part of Starting Y-coordinate into Y-plane.
15:3	Y_F	Fractional Part of Starting Y-coordinate into Y-plane.
2:0	Reserved	These bits are reserved.

CSC U Source Base

Read/Write MMIO_base + 0x1000D8

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory Address of source Y-plane with 128-bit Alignment.
3:0	0000	These bits are hardwired to zeros.

CSC V Source Base

Read/Write MMIO_base + 0x1000DC

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0000			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory Address of source Y-plane with 128-bit Alignment.
3:0	0000	These bits are hardwired to zeros.

CSC Source Dimension

Read/Write MMIO_base + 0x1000E0

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Width R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Height R/W															

Bit(s)	Name	Description
31:16	Width	Width of Source in Pixels.
15:0	Height	Height of Source in Lines.

CSC Source Pitch

Read/Write MMIO_base + 0x1000E4

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Y R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
UV R/W															

Bit(s)	Name	Destination
31:16	Y	Pitch of Y-plane specified in bytes ÷ 16.
15:0	UV	Pitch of UV-planes specified in bytes ÷ 16.

CSC Destination

Read/Write MMIO_base + 0x1000E8

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31	W	Wrap Control. 0: Disable. 1: Enable.
30:28	Reserved	These bits are reserved.
27:16	X	X-coordinate of Destination.
15:12	Reserved	These bits are reserved.
11:0	Y	Y-coordinate of Destination.

CSC Destination Dimension

Read/Write MMIO_base + 0x1000EC

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Width R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Height R/W															

Bit(s)	Name	Description
31:16	Width	Width of Destination.
15:0	Height	Height of Destination.

CSC Destination Pitch

Read/Write MMIO_base + 0x1000F0

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
X R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Y R/W															

Bit(s)	Name	Description
31:16	X	Horizontal Pitch of Destination specified in bytes ÷ 16.
15:0	Y	Vertical Pitch of Destination specified in pixels.

CSC Scale Factor

Read/Write MMIO_base + 0x1000F4

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
X R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Y R/W															

Bit(s)	Name	Description
31:16	X	Horizontal scale factor specified in 3.13 format. Scale factor = 2^{13} * ($Width_S$ – 1) / ($Width_D$ – 1).
15:0	Y	Vertical scale factor specified in 3.13 format. Scale factor = 2^{13} * ($Height_S$ – 1) / ($Height_D$ – 1).

CSC Destination Base

Read/Write MMIO_base + 0x1000F8

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0000			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory Address of Destination Window with 128-bit Alignment.
3:0	0000	These bits are hardwired to zeros.

CSC Control

Read/Write MMIO_base + 0x1000FC

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	$Format_S$ R/W			$Format_D$ R/W		H R/W	V R/W	B R/W	Reserved						
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved															

Bit(s)	Name	Description
31	S	Color Space Conversion Control. 0: Stop. 1: Start.
30:28	$Format_S$	Source Pixel Format. 000 YUV422 — 100 Reserved 001 YUV420I — 101 Reserved 010 YUV420 — 110 RGB565 011 Reserved — 111 RGBx888
27:26	$Format_D$	Destination Pixel Format. 00: RGB565. 01: RGBx888.
25	H	Horizontal Linear Filter Control. 0: Disable. 1: Enable.
24	V	Vertical Linear Filter Control. 0: Disable. 1: Enable.
23	B	Byte Order for YUV422 and YUV420I. 0: YUV422 = YUYV, YUV420I = UVUV 1: YUV422 = UYVY, YUV420I = VUVU
22:0	Reserved	These bits are reserved.

5 Display Controller

Programmer's Model

The SM501 integrates a concurrent video processor to control LCD display. Some of the features are:

- Background graphic supports from 4-bit index color, 8-bit index color, 15-bit direct color, and 16-bit direct color. Background graphic can be programmed to pan to the left/right and to up/down automatically according to number of VSYNC.
- Support 1 independent video surface using hardware scaling for any size of video windows at any location of the screen display and using hardware YUV to RGB color space conversion.
- Support 1 Alpha blend surface at any location of the screen display. It can use as hardware cursor or popup icon or sub picture for the video. Data format is 4-bit alpha and 4-bit color.
- The LCD module manages data flow and generate timing to select LCD display. It provides support for 18-bit and 24-bit TFT and 8-bits and 12-bits DSTN panel up to SXGA.

The Video Processor Control Registers specify the control registers for Video Processor. The Video Processor Control Registers can only be accessed through memory-mapping.

Register Descriptions

Table 5-1 summarizes the Display Controller registers.

Table 5-1: Display Controller Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
Panel Graphics Control				
0x080000	R/W	32	0x00010000	Panel Display Control
0x080004	R/W	32	Undefined	Panel Panning Control
0x080008	R/W	32	Undefined	Panel Color Key
0x08000C	R/W	32	Undefined	Panel FB Address
0x080010	R/W	32	Undefined	Panel FB Offset/Window Width
0x080014	R/W	32	Undefined	Panel FB Width
0x080018	R/W	32	Undefined	Panel FB Height
0x08001C	R/W	32	Undefined	Panel Plane TL Location
0x080020	R/W	32	Undefined	Panel Plane BR Location
0x080024	R/W	32	Undefined	Panel Horizontal Total
0x080028	R/W	32	Undefined	Panel Horizontal Sync
0x08002C	R/W	32	Undefined	Panel Vertical Total
0x080030	R/W	32	Undefined	Panel Vertical Sync
0x080034	R	32	0b0000.0000.0000.0000. 0000.0XXX.XXXX.XXXX	Panel Current Line
Video Control				
0x080040	R/W	32	0b0000.0000.0000.0001. X000.0000.0000.0000	Video Display Control
0x080044	R/W	32	Undefined	Video FB 0 Address
0x080048	R/W	32	Undefined	Video FB Width
0x08004C	R/W	32	Undefined	Video FB 0 Last Address
0x080050	R/W	32	Undefined	Video Plane TL Location
0x080054	R/W	32	Undefined	Video Plane BR Location

Table 5-1: Display Controller Register Summary (Continued)

Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
0x080058	R/W	32	0x00000000	Video Scale
0x08005C	R/W	32	0x00000000	Video Initial Scale
0x080060	R/W	32	0x00EDEDED	Video YUV Constants
0x080064	R/W	32	Undefined	Video FB 1 Address
0x080068	R/W	32	Undefined	Video FB 1 Last Address
Video Alpha Control				
0x080080	R/W	32	0x00000000	Video Alpha Display Control
0x080084	R/W	32	Undefined	Video Alpha FB Address
0x080088	R/W	32	Undefined	Video Alpha FB Offset/Window Width
0x08008C	R/W	32	Undefined	Video Alpha FB Last Address
0x080090	R/W	32	Undefined	Video Alpha Plane TL Location
0x080094	R/W	32	Undefined	Video Alpha Plane BR Location
0x080098	R/W	32	0x00000000	Video Alpha Scale
0x08009C	R/W	32	0x00000000	Video Alpha Initial Scale
0x0800A0	R/W	32	Undefined	Video Alpha Chroma Key
0x0800A4 - 0x0800C0	R/W	32	Undefined	Video Alpha Color Lookup
Panel Cursor Control				
0x0800F0	R/W	32	Undefined	Panel HWC Address
0x0800F4	R/W	32	Undefined	Panel HWC Location
0x0800F8	R/W	32	Undefined	Panel HWC Color 1 & 2
0x0800FC	R/W	32	Undefined	Panel HWC Color 3
Alpha Control				
0x080100	R/W	32	0x00010000	Alpha Display Control
0x080104	R/W	32	Undefined	Alpha FB Address
0x080108	R/W	32	Undefined	Alpha FB Offset/Window Width

Table 5-1: Display Controller Register Summary (Continued)

Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
0x08010C	R/W	32	Undefined	Alpha Plane TL Location
0x080110	R/W	32	Undefined	Alpha Plane BR Location
0x080114	R/W	32	Undefined	Alpha Chroma Key
0x080118 - 0x080134	R/W	32	Undefined	Alpha Color Lookup
CRT Graphics Control				
0x080200	R/W	32	0x00010000	CRT Display Control
0x080204	R/W	32	Undefined	CRT FB Address
0x080208	R/W	32	Undefined	CRT FB Offset/Window Width
0x08020C	R/W	32	Undefined	CRT Horizontal Total
0x080210	R/W	32	Undefined	CRT Horizontal Sync
0x080214	R/W	32	Undefined	CRT Vertical Total
0x080218	R/W	32	Undefined	CRT Vertical Sync
0x08021C	R/W	32	Undefined	CRT Signature Analyzer
0x080220	R	32	0b0000.0000.0000.0000. 0000.0XXX.XXXX.XXXX	CRT Current Line
0x080224	R/W	32	0b0000.0000.XXXX.XXXX. XXXX.XXXX.XXXX.XXXX	CRT Monitor Detect
CRT Cursor Control				
0x080230	R/W	32	Undefined	CRT HWC Address
0x080234	R/W	32	Undefined	CRT HWC Location
0x080238	R/W	32	Undefined	CRT HWC Color 1 & 2
0x08023C	R/W	32	Undefined	CRT HWC Color 3
Palette RAM				
0x080400 - 0x0807FC	R/W	32	Undefined	Panel Palette RAM
0x080800 - 0x080BFC	R/W	32	Undefined	Video Palette RAM

Table 5-1: Display Controller Register Summary (Continued)

Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
0x080C00 - 0x080FFC	R/W	32	Undefined	CRT Palette RAM

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. In the reset values, "X" indicates don't care.

Figure 5-1 shows how this 64kB region in the MMIO space is laid out. It controls the backend of the display controller as shown in Figure 5-2.

Figure 5-1: Display Controller Register Space

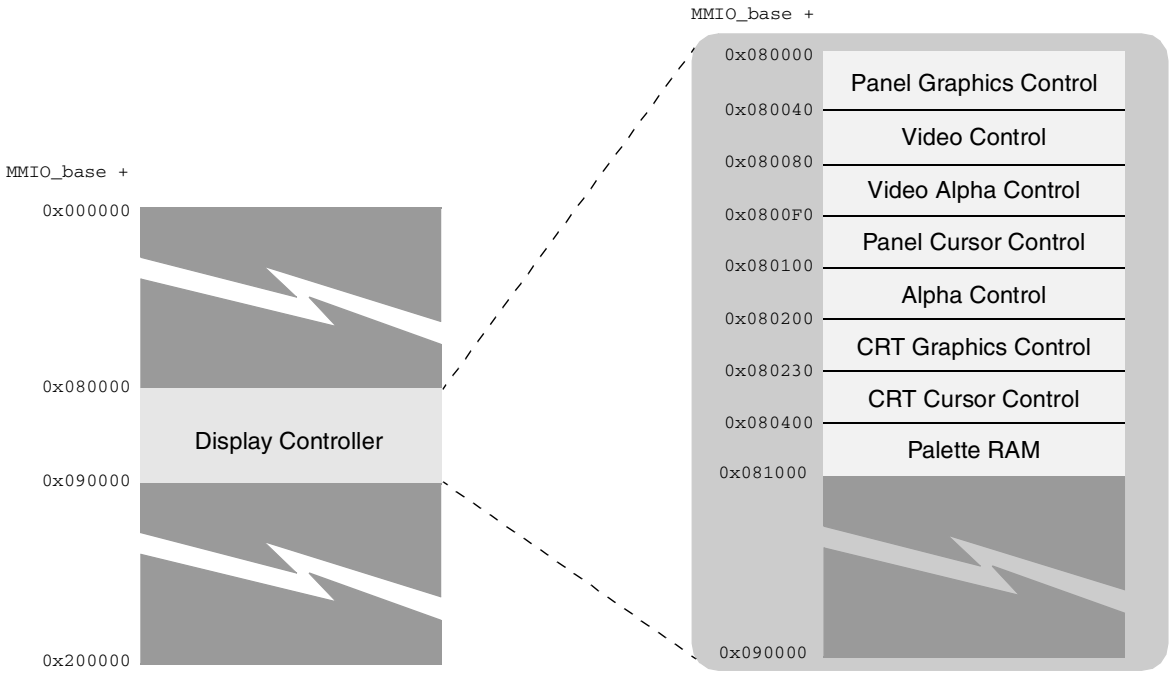
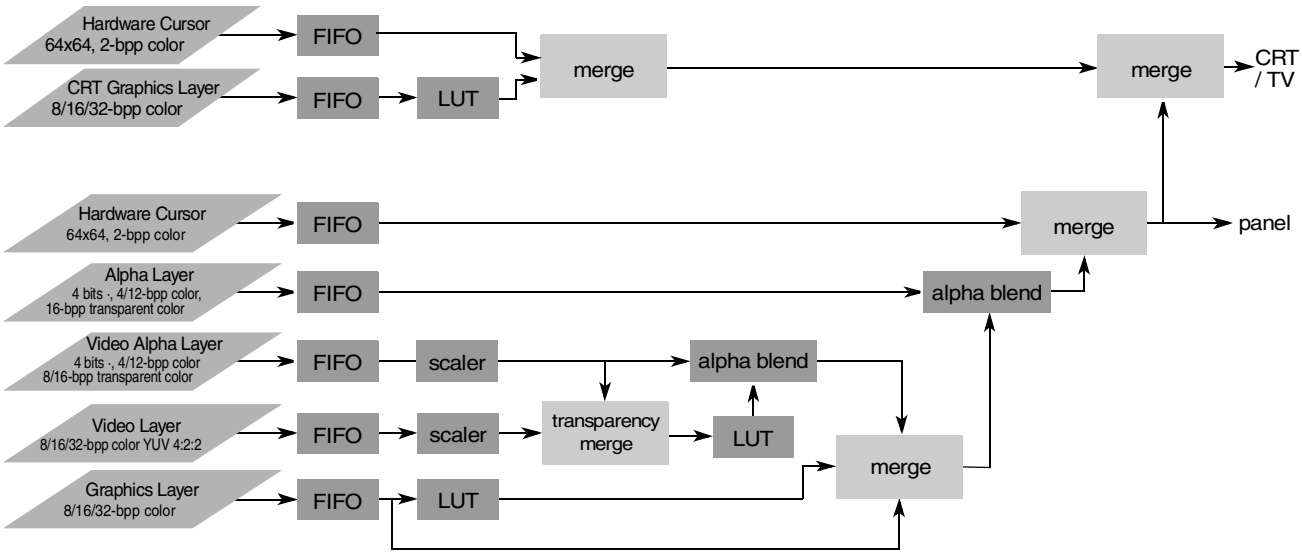


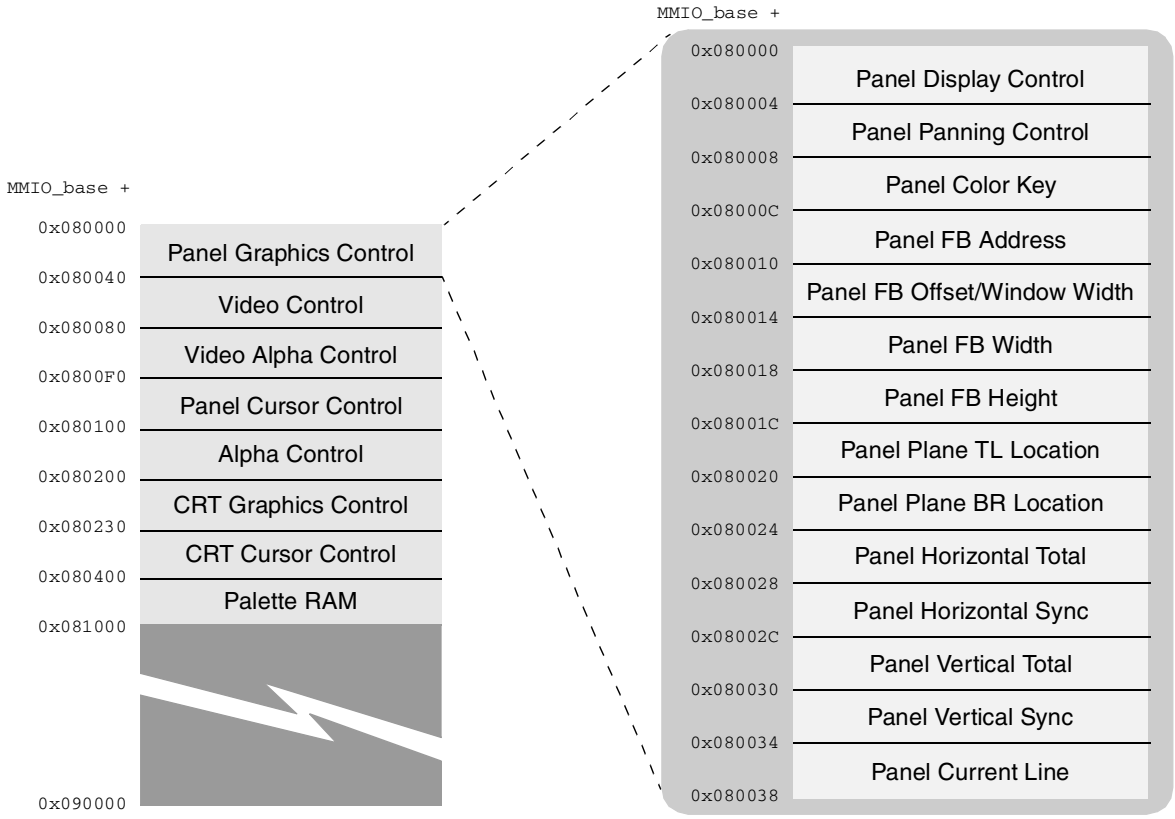
Figure 5-2: Video Layers



Panel Graphics Control Registers

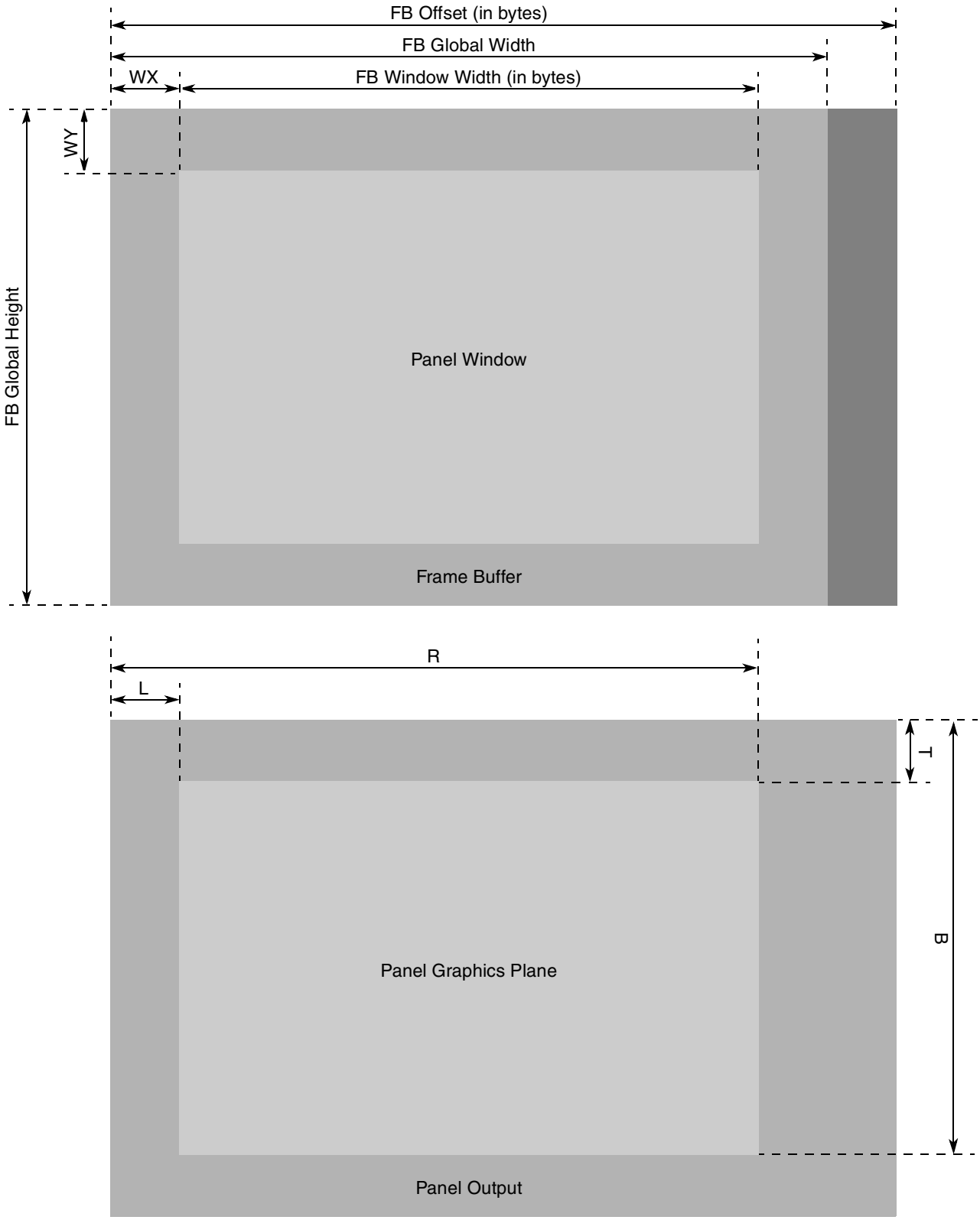
Figure 5-3 shows the layout of the Panel Graphics Control registers.

Figure 5-3: Panel Graphics Control Register Space



To understand video windowing, please refer to Figure 5-4. Here a window is created inside a much large frame buffer. That window is then being displayed on the panel as the Panel Graphics Plane.

Figure 5-4: Video Windowing



Panel Display Control

Read/Write MMIO_base + 0x080000

Power-on Default 0x00010000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res				En R/W	Bias R/W	Data R/W	VDD R/W	DP R/W	TFT R/W		DE R/W	LCD R/W		FIFO R/W	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res	CP R/W	VSP R/W	HSP R/W	Res		CK R/W	TE R/W	VPD R/W	VP R/W	HPD R/W	HP R/W	γ R/W	E R/W	Format R/W	

Bit(s)	Name	Description
31:28	Res	These bits are reserved.
27	En	Control FPEN Output Pin. 0: Driven low. 1: Driven high.
26	Bias	Control VBIASEN Output Pin. 0: Driven low. 1: Driven high.
25	Data	Panel Control Signals and Data Lines Enable. 0: Disable panel control signals and data lines. 1: Enable panel control signals and data lines.
24	VDD	Control FPVDDEN Output Pin. 0: Driven low. 1: Driven high.
23	DP	Select TFT Dithering Pattern. 0: 4-gray level dithering pattern. 1: 8-gray level dithering pattern.
22:21	TFT	Select TFT Panel Interface. 00: 24-bit RGB 8:8:8. 01: 9-bit RGB 3:3:3. 10: 12-bit RGB 4:4:4.
20	DE	Enable TFT Dithering. 0: Disable. 1: Enable.
19:18	LCD	Select LCD Type. 00: TFT panel. 01: 8-bit STN panel. 11: 12-bit STN panel.
17:16	FIFO	Panel Data FIFO Request Level. When the FIFO empty level reaches the level specified, a FIFO read request will be generated. 00: 1 or more entries empty. 01: 3 or more entries empty. 10: 7 or more entries empty. 11: 11 or more entries empty.
15	Res	This bit is reserved.
14	CP	Clock Phase Select. 0: Clock active high. 1: Clock active low.

Bit(s)	Name	Description
13	VSP	Vertical Sync Pulse Phase Select. 0: Vertical sync pulse active high. 1: Vertical sync pulse active low.
12	HSP	Horizontal Sync Pulse Phase Select. 0: Horizontal sync pulse active high. 1: Horizontal sync pulse active low.
11:10	Res	These bits are reserved.
9	CK	Enable Color Key. 0: Disable color key. 1: Enable color key.
8	TE	Enable Panel Timing. 0: Disable panel timing. 1: Enable panel timing.
7	VPD	Vertical Panning Direction. 0: Panning down. 1: Panning up.
6	VP	Enable Automatic Vertical Panning. 0: Disable. 1: Enable.
5	HPD	Horizontal Panning Direction. 0: Pan to the right. 1: Pan to the left.
4	HP	Enable Automatic Horizontal Panning. 0: Disable. 1: Enable.
3	γ	Enable Gamma Control. Gamma control can only be enabled in RGB 5:6:5 and RGB 8:8:8 modes. 0: Disable. 1: Enable.
2	E	Panel Graphics Plane Enable. 0: Disable panel graphics plane. 1: Enable panel graphics plane.
1:0	Format	Panel Graphics Plane Format. 00: 8-bit indexed mode. 01: 16-bit RGB 5:6:5 mode. 10: 32-bit RGB 8:8:8 mode.

Panel Panning Control

Read/Write MMIO_base + 0x080004

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VPan R/W								Reserved		VWait R/W					
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
HPan R/W								Reserved		HWait R/W					

Bit(s)	Name	Description
31:24	VPan	Number of lines to pan vertically.
23:22	Reserved	These bits are reserved.
21:16	VWait	Number of vertical sync pulses for each vertical pan.
15:8	HPan	Number of pixels to pan horizontally.
7:6	Reserved	These bits are reserved.
5:0	HWait	Number of horizontal sync pulses for each horizontal pan.

Panel Color Key

Read/Write MMIO_base + 0x080008

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mask R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Value R/W															

Bit(s)	Name	Description
31:16	Mask	Color key mask for video window plane.
15:0	Value	Color key value for video window plane.

Panel FB Address

Read/Write MMIO_base + 0x08000C

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: No flip pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of frame buffer for the panel graphics plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Panel FB Offset/Window Width

Read/Write MMIO_base + 0x080010

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		FB Window Width R/W										0 0 0 0			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved		FB Offset R/W										0 0 0 0			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29:20	FB Window Width	Number of bytes per line of the frame buffer window specified in 128-bit aligned bytes.

Bit(s)	Name	Description
19:16	0000	These bits are hardwired to zeros.
15:14	Reserved	These bits are reserved.
13:4	FB Offset	Number of 128-bit aligned bytes per line of the FB graphics plane (see Figure 5-4).
3:0	0000	These bits are hardwired to zeros.

Panel FB Width

Read/Write MMIO_base + 0x080014

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				FB Global Width R/W											
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				WX R/W											

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	FB Global Width	Width of FB graphics window specified in pixels (see Figure 5-4).
15:12	Reserved	These bits are reserved.
11:0	WX	Starting x-coordinate of panel graphics window specified in pixels (see Figure 5-4).

Panel FB Height

Read/Write MMIO_base + 0x080018
Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				FB Global Height R/W											
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				WY R/W											

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	FB GLobal Height	Height of FB graphics window specified in lines (see Figure 5-4).
15:12	Reserved	These bits are reserved.
11:0	WY	Starting y-coordinate of panel graphics window specified in lines (see Figure 5-4).

Panel Plane TL Location

Read/Write MMIO_base + 0x08001C
Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	T	Top location of the panel graphics plane specified in lines (see Figure 5-4).
15:11	Reserved	These bits are reserved.
10:0	L	Left location of the panel graphics plane specified in pixels (see Figure 5-4).

Panel Plane BR Location

Read/Write MMIO_base + 0x080020
Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	B	Bottom location of the panel graphics plane specified in lines (see Figure 5-4).
15:11	Reserved	These bits are reserved.
10:0	R	Right location of the panel graphics plane specified in pixels (see Figure 5-4).

Panel Horizontal Total

Read/Write MMIO_base + 0x080024
Power-on Default Undefined

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

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	HT	Panel horizontal total specified as number of pixels - 1.
15:12	Reserved	These bits are reserved.
11:0	HDE	Panel horizontal display end specified as number of pixels - 1.

Panel Horizontal Sync

Read/Write MMIO_base + 0x080028

Power-on Default Undefined

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

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:16	HSW	Panel horizontal sync width specified in pixels.
15:12	Reserved	These bits are reserved.
11:0	HS	Panel horizontal sync start specified as pixel number - 1.

Panel Vertical Total

Read/Write MMIO_base + 0x08002C

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	VT	Panel vertical total specified as number of lines - 1.
15:11	Reserved	These bits are reserved.
10:0	VDE	Panel vertical display end specified as number of lines - 1.

Panel Vertical Sync

Read/Write MMIO_base + 0x080030
Power-on Default Undefined

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

Bit(s)	Name	Description
31:22	Reserved	These bits are reserved.
21:16	VSH	Panel vertical sync height specified in lines.
15:11	Reserved	These bits are reserved.
10:0	VS	Panel vertical sync start specified as line number - 1.

Panel Current Line

Read MMIO_base + 0x080034
Power-on Default 0b0000.0000.0000.0000.0000.0XXX.XXXX.XXXX

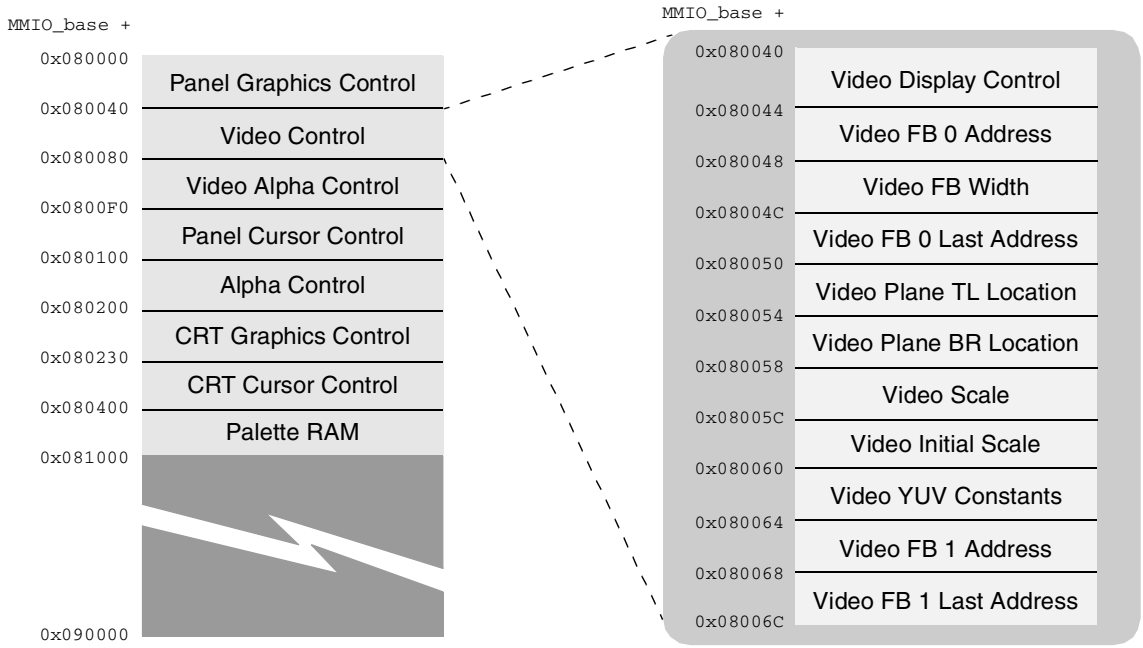
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved					Line R										

Bit(s)	Name	Description
31:11	Reserved	These bits are reserved.
10:0	Line	Panel current line being fetched.

Video Control Registers

Figure 5-5 shows the layout of the Video Control registers.

Figure 5-5: Video Control Register Space



Video Display Control

Read/Write MMIO_base + 0x080040

Power-on Default 0b0000.0000.0000.0001.X000.0000.0000.0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														FIFO R/W	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Buf R	CB R/W	DB R/W	BS R/W	VS R/W	HS R/W	VI R/W	HI R/W	Pixel R/W				γ R/W	E R/W	Format R/W	

Bit(s)	Name	Description
31:18	Reserved	These bits are reserved.
17:16	FIFO	Video Data FIFO Request Level. When the FIFO empty level reaches the level specified, a FIFO read request will be generated. 00: 1 or more entries empty. 01: 3 or more entries empty. 10: 7 or more entries empty. 11: 11 or more entries empty.

Bit(s)	Name	Description
15	Buf	Current Video Frame Buffer Used. This bit is read-only. 0: Buffer 0. 1: Buffer 1.
14	CB	Use Capture Frame Buffer as Video Frame Buffer. 0: Disable. 1: Enable.
13	DB	Enable Double Buffering. 0: Disable. 1: Enable.
12	BS	Enable Byte Swapping for YUV Data. 0: Disable (YUYV). 1: Enable (UYVY).
11	VS	Force Vertical Scale Factor to ½. 0: Disable. 1: Enable.
10	HS	Force Horizontal Scale Factor to ½. 0: Disable. 1: Enable.
9	VI	Enable Vertical Interpolation. 0: Disable. 1: Enable.
8	HI	Enable Horizontal Interpolation. 0: Disable. 1: Enable.
7:4	Pixel	Starting Pixel Number for Smooth Pixel Panning.
3	γ	Enable Gamma Control. Gamma control can be enabled only in RGB 5:6:5 and RGB 8:8:8 modes.[1] 0: Disable. 1: Enable.
2	E	Video Plane Enable. 0: Disable video plane. 1: Enable video plane.
1:0	Format	Video Plane Format. 00: 8-bit indexed mode. 01: 16-bit RGB 5:6:5 mode. 10: 32-bit RGB 8:8:8 mode. 11: 16-bit YUYV mode.

1. All display devices have an inherent non-linearity so that the intensity of the output is not linearly proportional to the input signal over the full range of input values. The gamma control helps to correct this nonlinearity.

Video FB 0 Address

Read/Write MMIO_base + 0x080044

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: No flip pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of frame buffer 0 for the video plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Video FB Width

Read/Write MMIO_base + 0x080048

Power-on Default Undefined

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

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29:20	Width	Number of bytes per line of the video plane specified in 128-bit aligned bytes.

Bit(s)	Name	Description
19:16	0000	These bits are hardwired to zeros.
15:14	Reserved	These bits are reserved.
13:4	Offset	Number of 128-bit aligned bytes per line of the video plane.
3:0	0000	These bits are hardwired to zeros.

Video FB 0 Last Address

Read/Write MMIO_base + 0x08004C

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of last byte of frame buffer 0 for the video plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Video Plane TL Location

Read/Write MMIO_base + 0x080050

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	T	Top location of the video plane specified in lines.
15:11	Reserved	These bits are reserved.
10:0	L	Left location of the video plane specified in pixels.

Video Plane BR Location

Read/Write MMIO_base + 0x080054

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	B	Bottom location of the video plane specified in lines.
15:11	Reserved	These bits are reserved.
10:0	R	Right location of the video plane specified in pixels.

Video Scale

Read/Write MMIO_base + 0x080058

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VS R/W	Reserved			VScale R/W											
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
HS R/W	Reserved			HScale R/W											

Bit(s)	Name	Description
31	VS	Select Vertical Video Scaling. 0: Video expands vertically. 1: Video shrinks vertically.
30:28	Reserved	These bits are reserved.
27:16	VScale	Vertical Video Scale Factor. For expansion: VScale = ($height_{src}$ / $height_{dest}$) * 2^{12}. For shrinking: VScale = ($height_{dest}$ / $height_{src}$) * 2^{12}.
15	HS	Select Horizontal Video Scaling. 0: Video expands horizontally. 1: Video shrinks horizontally.
14:12	Reserved	These bits are reserved.
11:0	HScale	Horizontal Video Scale Factor. For expansion: HScale = ($width_{src}$ / $width_{dest}$) * 2^{12} For shrinking: HScale = ($width_{dest}$ / $width_{src}$) *2^{12}

Scaling example: To expand (magnify) the horizontal scale by a factor of 3:

1. Set HS = 0.
2. Calculate the scaling factor:

   ($width_{src}$ / $width_{dest}$) * 2^{12} = (1/3) * 2^{12}
3. Set HScale. In this example, HScale = 0101 0101 0101 b or 555h.

To shrink the horizontal scale by a factor of 3:

1. Set HS = 1.
2. Calculate the scaling factor:

   ($width_{dest}$ / $width_{src}$) * 2^{12} = ((1/3)/1) * 2^{12} = 1/3 * 2^{12}
3. Set HScale. Note that the HScale setting is the same for shrinking by 1/3 as it is for magnifying by a factor of 3, only the setting of HS differs

Video Initial Scale

Read/Write MMIO_base + 0x08005C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				$VScale_1$ R/W											
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				$VScale_0$ R/W											

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	$VScale_1$	Initial vertical scale factor for video buffer 1.
15:12	Reserved	These bits are reserved.
11:0	$VScale_0$	Initial vertical scale factor for video buffer 0.

Video YUV Constants

Read/Write MMIO_base + 0x080060

Power-on Default 0x00EDEDED

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

Bit(s)	Name	Description
31:24	Y	Y Adjustment.
23:16	R	Red Conversion Constant.
15:8	G	Green Conversion Constant.
7:0	B	Blue Conversion Constant.

Video FB 1 Address

Read/Write MMIO_base + 0x080064

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W R/W	Address									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: No flip pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of frame buffer 1 for the video plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Video FB 1 Last Address

Read/Write MMIO_base + 0x080068

Power-on Default Undefined

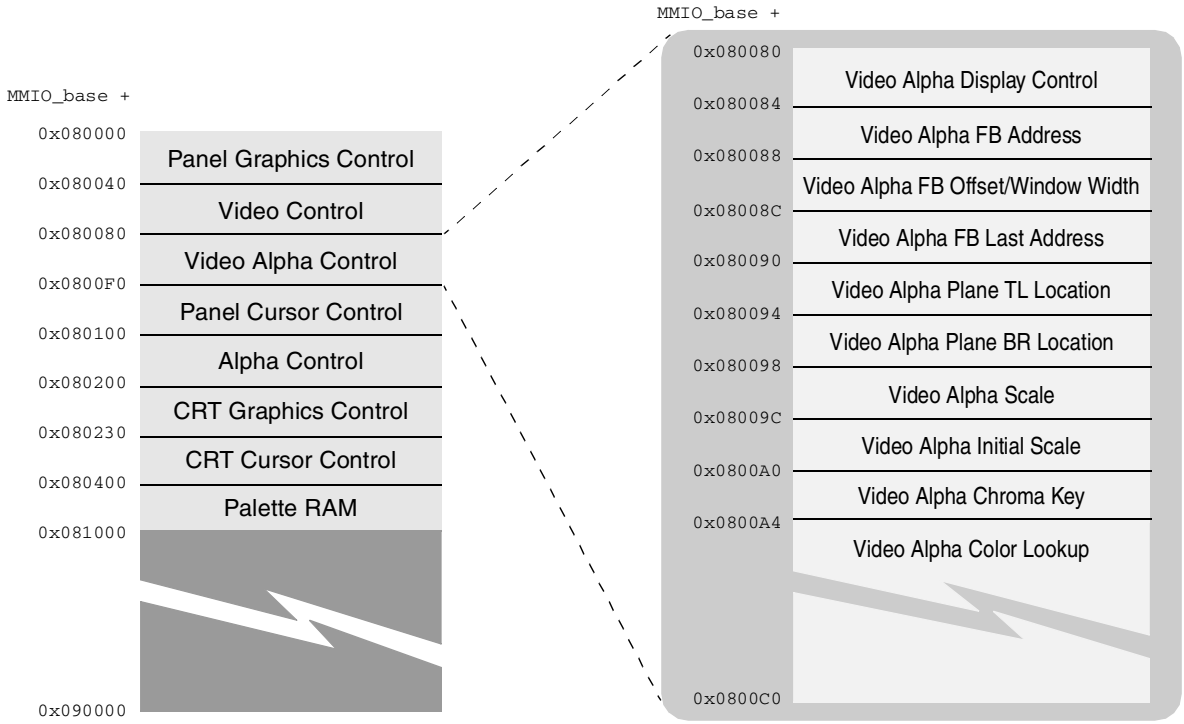
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of last byte of frame buffer 1 for the video plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Video Alpha Control Registers

Figure 5-6 shows the layout of the Video Alpha Control registers.

Figure 5-6: Video Alpha Control Register Space



Video Alpha Display Control

Read/Write MMIO_base + 0x080080

Power-on Default 0x00010000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			Sel R/W	Alpha R/W				Reserved						FIFO R/W	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				VS R/W	HS R/W	VI R/W	HI R/W	Pixel R/W				CK R/W	E R/W	Format R/W	

Bit(s)	Name	Description
31:29	Reserved	These bits are reserved.
28	Sel	Alpha Select. 0: Use per-pixel alpha values. 1: Use alpha value specified in *Alpha*.

Bit(s)	Name	Description
27:24	Alpha	Video Alpha Plane Alpha Value. This field is only valid when the *Sel* bit is 1.
23:18	Reserved	These bits are reserved.
17:16	FIFO	Video Alpha Data FIFO Request Level. When the FIFO empty level reaches the level specified, a FIFO read request will be generated. 00: 1 or more entries empty. 01: 3 or more entries empty. 10: 7 or more entries empty. 11: 11 or more entries empty.
15:12	Reserved	These bits are reserved.
11	VS	Force Vertical Scale Factor to ½. 0: Disable. 1: Enable.
10	HS	Force Horizontal Scale Factor to ½. 0: Disable. 1: Enable.
9	VI	Enable Vertical Interpolation. 0: Disable. 1: Enable.
8	HI	Enable Horizontal Interpolation. 0: Disable. 1: Enable.
7:4	Pixel	Starting Pixel Number for Smooth Pixel Panning.
3	CK	Enable Chroma Key. 0: Disable. 1: Enable.
2	E	Video Alpha Plane Enable. 0: Disable video plane. 1: Enable video plane.
1:0	Format	Video Alpha Plane Format. 00: 8-bit indexed mode. 01: 16-bit RGB 5:6:5 mode. 10: 8-bit indexed αI 4:4 mode. 11: 16-bit αRGB 4:4:4:4 mode.

Video Alpha FB Address

Read/Write MMIO_base + 0x080084

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: No flip pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of frame buffer for the video alpha plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Video Alpha FB Offset/Window Width

Read/Write MMIO_base + 0x080088

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		Window Width R/W										0 0 0 0			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved		FB Offset R/W										0 0 0 0			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29:20	Window Width	Number of bytes per line of the video alpha window specified in 128-bit aligned bytes.

Bit(s)	Name	Description
19:16	0000	These bits are hardwired to zeros.
15:14	Reserved	These bits are reserved.
13:4	FB Offset	Number of 128-bit aligned bytes per line of the video alpha FB.
3:0	0000	These bits are hardwired to zeros.

Video Alpha FB Last Address

Read/Write MMIO_base + 0x08008C

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0000			

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of last byte of frame buffer for the video alpha plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Video Alpha Plane TL Location

Read/Write MMIO_base + 0x080090

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	Top	Top location of the video alpha plane specified in lines.
15:11	Reserved	These bits are reserved.
10:0	Left	Left location of the video alpha plane specified in pixels.

Video Alpha Plane BR Location

Read/Write MMIO_base + 0x080094

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	Bottom	Bottom location of the video alpha plane specified in lines.
15:11	Reserved	These bits are reserved.
10:0	Right	Right location of the video alpha plane specified in pixels.

Video Alpha Scale

Read/Write MMIO_base + 0x080098

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VS R/W	Reserved			VScale R/W											
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
HS R/W	Reserved			HScale R/W											

Bit(s)	Name	Description
31	VS	Select Vertical Video Scaling. 0: Video expands vertically. 1: Video shrinks vertically.
30:28	Reserved	These bits are reserved.
27:16	VScale	Vertical Video Scale Factor. For expansion: VScale = $height_{src}$ / $height_{dest}$ * 2^{12}. For shrinking: VScale = $height_{dest}$ / $height_{src}$ * 2^{12}.
15	HS	Select Horizontal Video Scaling. 0: Video expands horizontally. 1: Video shrinks horizontally.
14:12	Reserved	These bits are reserved.
11:0	HScale	Horizontal Video Scale Factor. For expansion: HScale = $width_{src}$ / $width_{dest}$ * 2^{12}. For shrinking: HScale = $width_{dest}$ / $width_{src}$ * 2^{12}.

Video Alpha Initial Scale

Read/Write MMIO_base + 0x08009C

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:12	Reserved	These bits are reserved.
11:0	VScale	Initial Vertical Scale Factor.

Video Alpha Chroma Key

Read/Write MMIO_base + 0x0800A0

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mask R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Value R/W															

Bit(s)	Name	Description
31:16	Mask	Chroma Key Mask for Video Alpha Plane. 0: Compare respective bit. 1: Do not compare respective bit.
15:0	Value	Chroma Key Value for Video Alpha Plane.

Video Alpha Color Lookup

Read/Write MMIO_base + 0x0800A4 - 0x0800C0

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Lookup_1$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Lookup_0$ R/W															

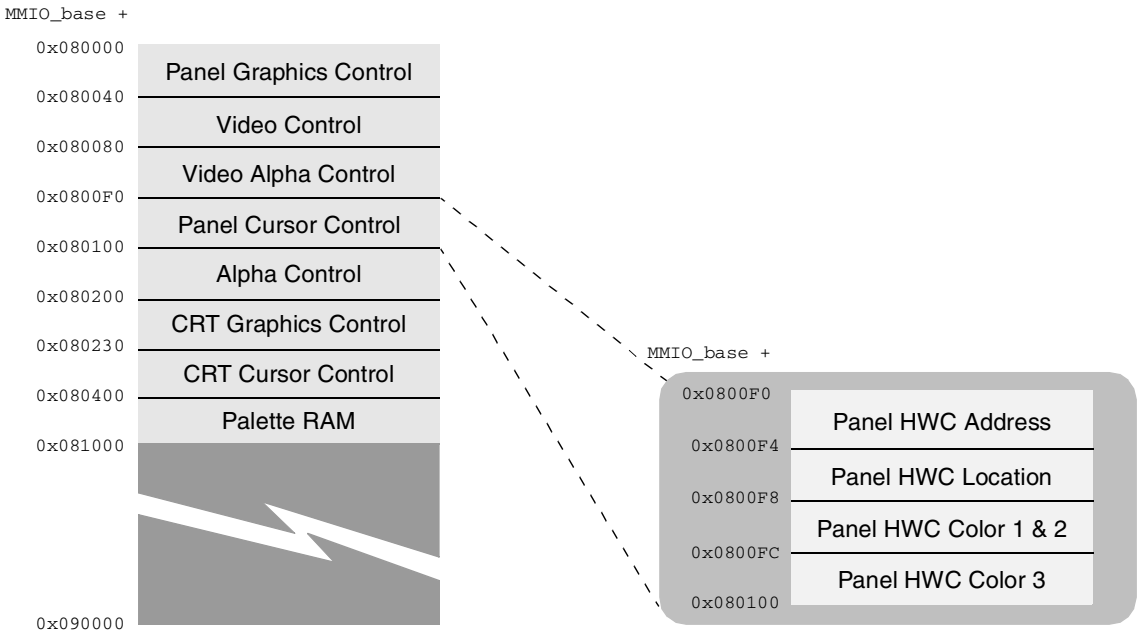
Bit(s)	Name	Description
31:16	$Lookup_1$	Alpha RGB 5:6:5 color lookup value for 4-bit indexed color 1.
15:0	$Lookup_0$	Alpha RGB 5:6:5 color lookup value for 4-bit indexed color 0.

There are 8 Video Alpha Color Lookup registers, each containing two RGB 5:6:5 color lookup values for each of the 16 4-bit indexed colors.

Panel Cursor Control Registers

Figure 5-7 shows the layout of the Panel Cursor Control registers.

Figure 5-7: Panel Cursor Control Register Space



Panel HWC Address

Read/Write MMIO_base + 0x0800F0

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
E R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	E	Enable Panel Hardware Cursor. 0: Disable. 1: Enable.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.

Bit(s)	Name	Description
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of panel hardware cursor with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Panel HWC Location

Read/Write MMIO_base + 0x0800F4

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				T R/W	Y R/W										
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				L R/W	X R/W										

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	T	Top Boundary Select. 0: Panel hardware cursor is within screen top boundary. 1: Panel hardware cursor is partially outside screen top boundary.
26:16	Y	Panel Hardware Cursor Y Position.
15:12	Reserved	These bits are reserved.
11	L	Left Boundary Select. 0: Panel hardware cursor is within screen left boundary. 1: Panel hardware cursor is partially outside screen left boundary.
10:0	X	Panel Hardware Cursor X Position.

Panel HWC Color 1 & 2

Read/Write MMIO_base + 0x0800F8

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Color_2$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Color_1$ R/W															

Bit(s)	Name	Description
31:16	$Color_2$	Panel hardware cursor color 2 in RGB 5:6:5.
15:0	$Color_1$	Panel hardware cursor color 1 in RGB 5:6:5.

Panel HWC Color 3

Read/Write MMIO_base + 0x0800FC

Power-on Default Undefined

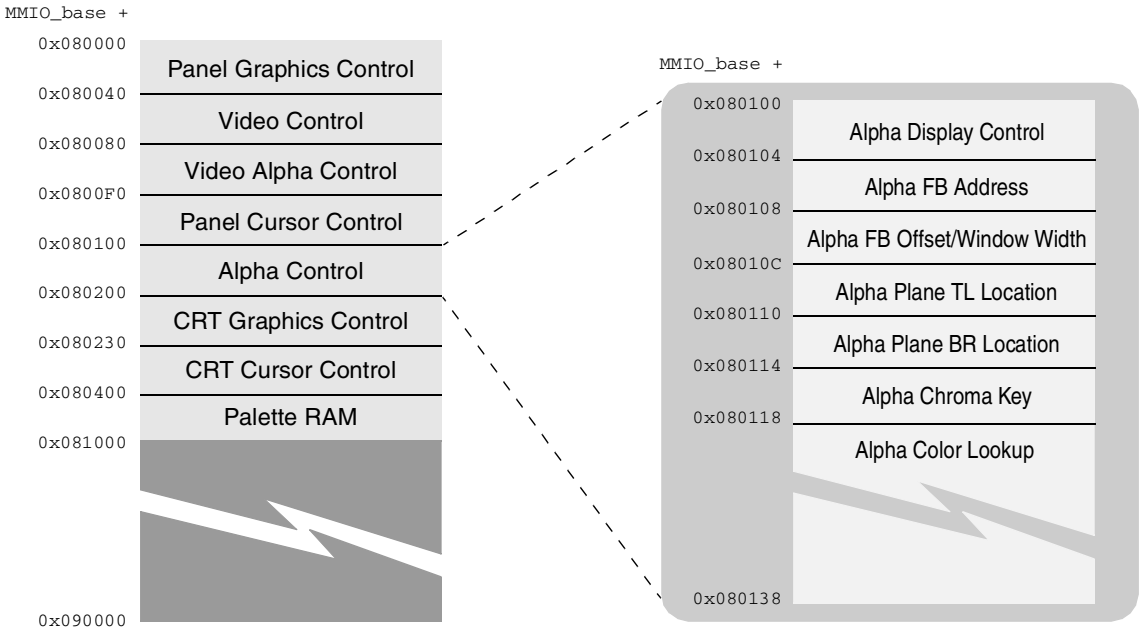
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Color_3$ R/W															

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	$Color_3$	Panel hardware cursor color 3 in RGB 5:6:5.

Alpha Control Registers

Figure 5-8 shows the layout of the Alpha Control registers.

Figure 5-8: Alpha Control Register Space



Alpha Display Control

Read/Write MMIO_base + 0x080100

Power-on Default 0x00010000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			Sel R/W	Alpha R/W				Reserved						FIFO R/W	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved								Pixel R/W				CK R/W	E R/W	Format R/W	

Bit(s)	Name	Description
31:29	Reserved	These bits are reserved.
28	Sel	Alpha Select. 0: Use per-pixel alpha values. 1: Use alpha value specified in *Alpha*.
27:24	Alpha	Alpha Plane Alpha Value. This field is only valid when the *Sel* bit is 1.
23:18	Reserved	These bits are reserved.

Bit(s)	Name	Description
17:16	FIFO	Video Data FIFO Request Level. When the FIFO empty level reaches the level specified, a FIFO read request will be generated. 00: 1 or more entries empty. 01: 3 or more entries empty. 10: 7 or more entries empty. 11: 11 or more entries empty.
15:8	Reserved	These bits are reserved.
7:4	Pixel	Starting Pixel Number for Smooth Pixel Panning.
3	CK	Enable Chroma Key. 0: Disable chroma key. 1: Enable chroma key.
2	E	Alpha Plane Enable. 0: Disable alpha plane. 1: Enable alpha plane.
1:0	Format	Alpha Plane Format. 01: 16-bit RGB 5:6:5 mode. 10: 8-bit indexed αI 4:4 mode. 11: 16-bit αRGB 4:4:4:4 mode.

Alpha FB Address

Read/Write MMIO_base + 0x080104

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: No flip pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of frame buffer for the alpha plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Alpha FB Offset/Window Width

Read/Write MMIO_base + 0x080108
Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		Window Width R/W										0 0 0 0			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved		FB Offset R/W										0 0 0 0			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29:20	Window Width	Number of bytes per line of the alpha window specified in 128-bit aligned bytes.
19:16	0000	These bits are hardwired to zeros.
15:14	Reserved	These bits are reserved.
13:4	FB Offset	Number of 128-bit aligned bytes per line of the alpha FB.
3:0	0000	These bits are hardwired to zeros.

Alpha Plane TL Location

Read/Write MMIO_base + 0x08010C
Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	Top	Top location of the alpha plane specified in lines.
15:11	Reserved	These bits are reserved.
10:0	Left	Left location of the alpha plane specified in pixels.

Alpha Plane BR Location

Read/Write MMIO_base + 0x080110

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	Bottom	Bottom location of the alpha plane specified in lines.
15:11	Reserved	These bits are reserved.
10:0	Right	Right location of the alpha plane specified in pixels.

Alpha Chroma Key

Read/Write MMIO_base + 0x080114

Power-on Default Undefined

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

Bit(s)	Name	Description
31:16	Mask	Chroma Key Mask for Alpha Plane. 0: Compare respective bit. 1: Do not compare respective bit.
15:0	Value	Chroma Key Value for Alpha Plane.

Alpha Color Lookup

Read/Write MMIO_base + 0x080118 - 0x080134

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Lookup_1$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Lookup_0$ R/W															

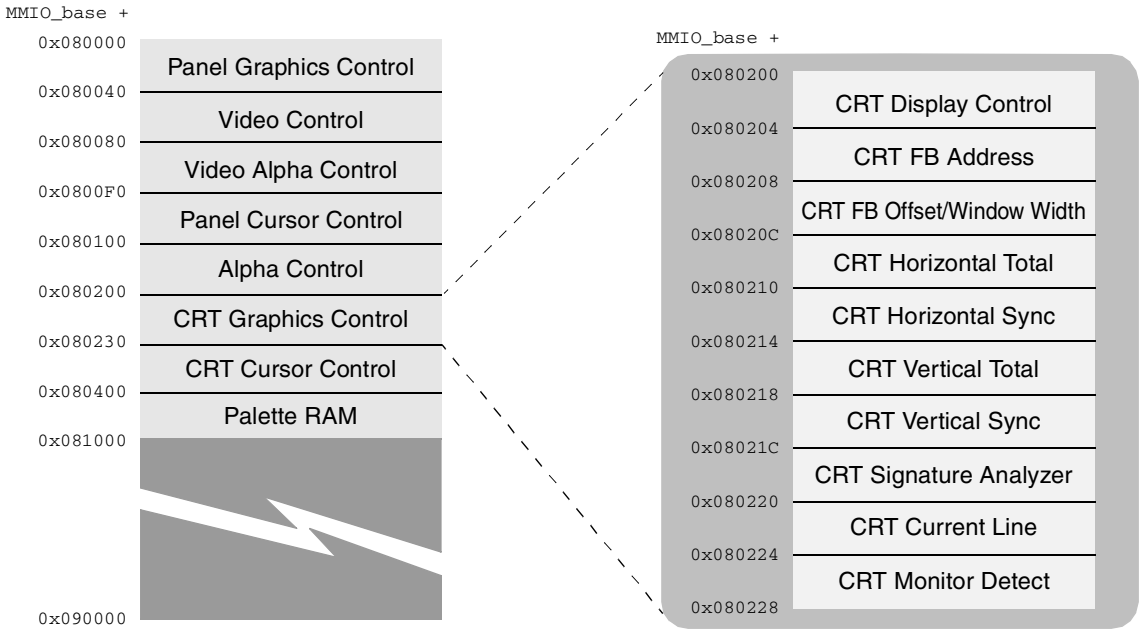
Bit(s)	Name	Description
31:16	$Lookup_1$	Alpha RGB 5:6:5 color lookup value for 4-bit indexed color 1.
15:0	$Lookup_0$	Alpha RGB 5:6:5 color lookup value for 4-bit indexed color 0.

There are 8 Alpha Color Lookup registers, each containing two RGB 5:6:5 color lookup values for each of the 16 4-bit indexed colors.

CRT Graphics Control Registers

Figure 5-9 shows the layout of the CRT Graphics Control registers.

Figure 5-9: CRT Graphics Control Register Space



CRT Display Control

Read/Write MMIO_base + 0x080200

Power-on Default 0x00010000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														FIFO R/W	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TVP R/W	CP R/W	VSP R/W	HSP R/W	VS R	B R/W	Sel R/W	TE R/W	Pixel R/W				γ R/W	E R/W	Format R/W	

Bit(s)	Name	Description
31:18	Reserved	These bits are reserved.
17:16	FIFO	CRT Data FIFO Request Level. When the FIFO empty level reaches the level specified, a FIFO read request will be generated. 00: 1 or more entries empty. 01: 3 or more entries empty. 10: 7 or more entries empty. 11: 11 or more entries empty.

Bit(s)	Name	Description
15	TVP	TV Clock Phase Select. 0: TV clock active high. 1: TV clock active low.
14	CP	CRT Clock Phase Select. 0: CRT clock active high. 1: CRT clock active low.
13	VSP	Vertical Sync Pulse Phase Select. 0: Vertical sync pulse active high. 1: Vertical sync pulse active low.
12	HSP	Horizontal Sync Pulse Phase Select. 0: Horizontal sync pulse active high. 1: Horizontal sync pulse active low.
11	VS	Vertical Sync. This bit is read only.
10	B	CRT Data Blanking. 0: CRT will show pixels. 1: CRT will be blank.
9	Sel	CRT Data Select. 0: CRT will display panel data. 1: CRT will display CRT data.
8	TE	Enable CRT Timing. 0: Disable CRT timing. 1: Enable CRT timing.
7:4	Pixel	Starting Pixel Number for Smooth Pixel Panning.
3	γ	Enable Gamma Control. Gamma control can be enabled only in RGB 5:6:5 and RGB 8:8:8 modes. 0: Disable gamma control. 1: Enable gamma control.
2	E	CRT Graphics Plane Enable. 0: Disable CRT Graphics plane. 1: Enable CRT Graphics plane.
1:0	Format	CRT Graphics Plane Format. 00: 8-bit indexed mode. 01: 16-bit RGB 5:6:5 mode. 10: 32-bit RGB 8:8:8 mode.

CRT FB Address

Read/Write MMIO_base + 0x080204

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: No flip pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of the frame buffer for the CRT graphics plane with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

CRT FB Offset/Window Width

Read/Write MMIO_base + 0x080208

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		Window Width R/W										0 0 0 0			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved		FB Offset R/W										0 0 0 0			

Bit(s)	Name	Description
31:30	Reserved	These bits are reserved.
29:20	Window Width	Number of bytes per line of the CRT graphics window specified in 128-bit aligned bytes.
19:16	0000	These bits are hardwired to zeros.
15:14	Reserved	These bits are reserved.
13:4	FB Offset	Number of 128-bit aligned bytes per line of the CRT graphics FB.
3:0	0000	These bits are hardwired to zeros.

CRT Horizontal Total

Read/Write MMIO_base + 0x08020C

Power-on Default Undefined

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

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27:16	HT	CRT horizontal total specified as number of pixels - 1.
15:12	Reserved	These bits are reserved.
11:0	HDE	CRT horizontal display end specified as number of pixels - 1.

CRT Horizontal Sync

Read/Write MMIO_base + 0x080210

Power-on Default Undefined

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

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:16	HSW	CRT horizontal sync width specified in pixels.
15:12	Reserved	These bits are reserved.
11:0	HS	CRT horizontal sync start specified as pixel number - 1.

CRT Vertical Total

Read/Write MMIO_base + 0x080214

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	VT	CRT vertical total specified as number of lines - 1.
15:11	Reserved	These bits are reserved.
10:0	VDE	CRT vertical display end specified as number of lines - 1.

CRT Vertical Sync

Read/Write MMIO_base + 0x080218

Power-on Default Undefined

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

Bit(s)	Name	Description
31:22	Reserved	These bits are reserved.
21:16	VSH	CRT vertical sync height specified in lines.
15:11	Reserved	These bits are reserved.
10:0	VS	CRT vertical sync start specified as line number - 1.

CRT Signature Analyzer

Read/Write MMIO_base + 0x08021C

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Status R															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved												E R/W	R R/W	Sel R/W	

Bit(s)	Name	Description
31:16	Status	Analyzer Signature. This field is read-only.
15:4	Reserved	These bits are reserved.
3	E	Enable Signature Analyzer. 0: Disable. 1: Enable.

Bit(s)	Name	Description
2	R	Reset Signature Analyzer. 0: Normal. 1: Reset.
1:0	Sel	Source Select for Signature Analyzer. 00: Red color. 01: Green color. 10: Blue color.

CRT Current Line

Read MMIO_base + 0x080220

Power-on Default 0b0000.0000.0000.0000.0000.0XXX.XXXX.XXXX

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

Bit(s)	Name	Description
31:11	Reserved	These bits are reserved.
10:0	Line	CRT Current Line Being Fetched.

CRT Monitor Detect

Read/Write MMIO_base + 0x080224

Power-on Default 0b0000.0000.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX

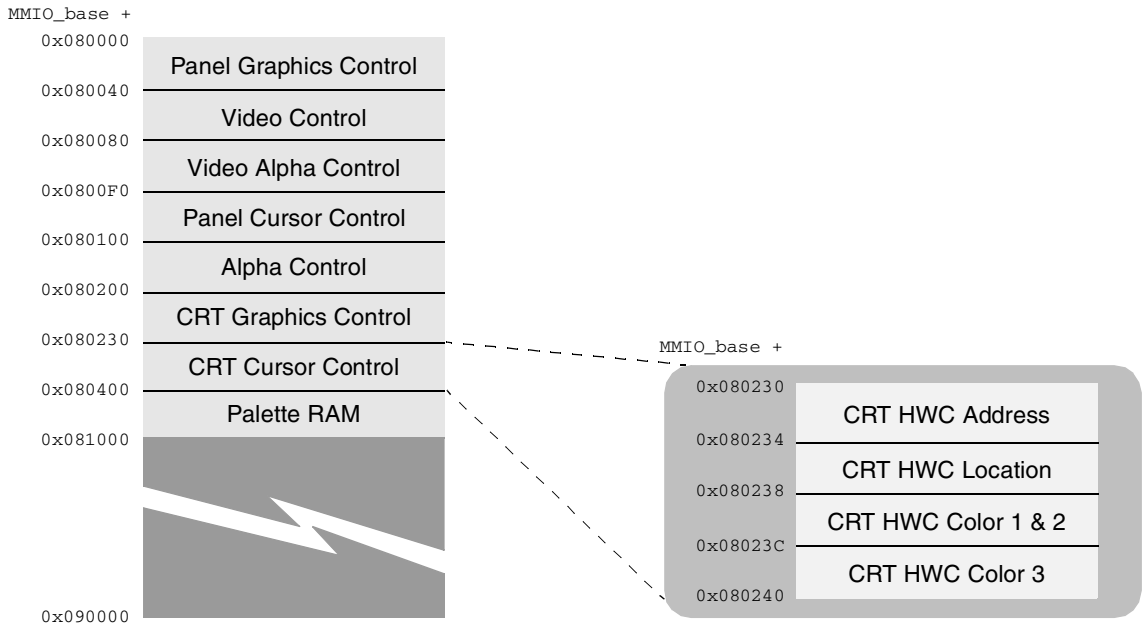
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						MDET R	E R/W	Data R							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Data R															

Bit(s)	Name	Description
31:26	Reserved	These bits are reserved.
25	MDET	Monitor Detect Read Back. 1: All R, G, and B voltages are greater than 0.325 V. 0: All R, G, and B voltages are less than or equal to 0.325 V.
24	E	Monitor Detect Enable. 0: Disable. 1: Enable.
23:0	Data	Monitor Detect Data in RGB 8:8:8. This field is read-only.

CRT Cursor Control Registers

Figure 5-10 shows the layout of the CRT Cursor Control registers.

Figure 5-10:CRT Cursor Control Register Space



CRT HWC Address

Read/Write MMIO_base + 0x080230

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
E R/W	Reserved			Ext R/W	CS R/W	Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	E	Enable CRT Hardware Cursor. 0: Disable. 1: Enable.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.

Bit(s)	Name	Description
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Address	Memory address of CRT hardware cursor with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

CRT HWC Location

Read/Write MMIO_base + 0x080234

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				T R/W	Y R/W										
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved				L R/W	X R/W										

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	T	Top Boundary Select. 0: CRT hardware cursor is within screen top boundary. 1: CRT hardware cursor is partially outside screen top boundary.
26:16	Y	CRT Hardware Cursor Y Position.
15:12	Reserved	These bits are reserved.
11	L	Left Boundary Select. 0: CRT hardware cursor is within screen left boundary. 1: CRT hardware cursor is partially outside screen left boundary.
10:0	X	CRT Hardware Cursor X Position.

CRT HWC Color 1 & 2

Read/Write MMIO_base + 0x080238

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Color_2$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Color_1$ R/W															

Bit(s)	Name	Description
31:16	$Color_2$	CRT Hardware Cursor Color 2 in RGB 5:6:5.
15:0	$Color_1$	CRT Hardware Cursor Color 1 in RGB 5:6:5.

CRT HWC Color 3

Read/Write MMIO_base + 0x08023C

Power-on Default Undefined

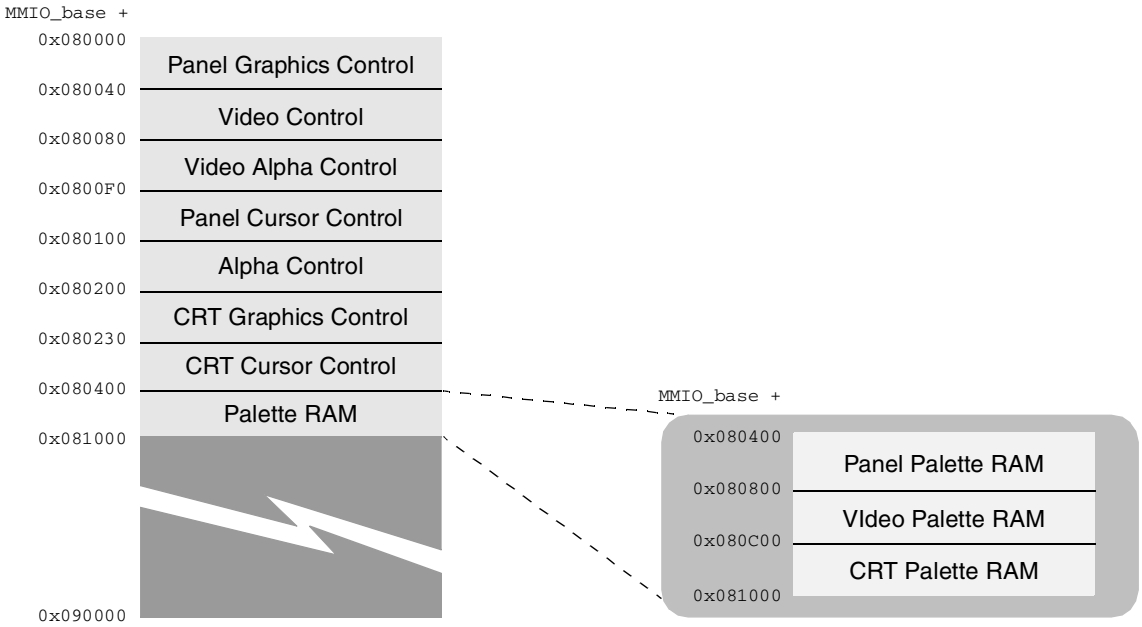
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Color_3$ R/W															

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	$Color_3$	CRT Hardware Cursor Color 3 in RGB 5:6:5.

Palette RAM Registers

Figure 5-11 shows the layout of the Palette RAM registers.

Figure 5-11:Palette RAM Register Space



Panel Palette RAM

Read/Write MMIO_base + 0x080400 - 0x0807FC

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								Red R/W							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Green R/W								Blue R/W							

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:16	Red	For indexed color modes: 8-bit red color value. For 16- and 32-bit color modes: 8-bit red alpha value.
15:8	Green	For indexed color modes: 8-bit green color value. For 16- and 32-bit color modes: 8-bit green alpha value.
7:0	Blue	For indexed color modes: 8-bit blue color value. For 16- and 32-bit color modes: 8-bit blue alpha value.

There are 256 Panel Palette RAM registers, each containing a 24-bit RGB 8:8:8 color value.

Video Palette RAM

Read/Write MMIO_base + 0x080800 - 0x080BFC

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								Red R/W							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Green R/W								Blue R/W							

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:16	Red	For indexed color modes: 8-bit red color value. For 16- and 32-bit color modes: 8-bit red alpha value.
15:8	Green	For indexed color modes: 8-bit green color value. For 16- and 32-bit color modes: 8-bit green alpha value.
7:0	Blue	For indexed color modes: 8-bit blue color value. For 16- and 32-bit color modes: 8-bit blue alpha value.

There are 256 Video Palette RAM registers, each containing a 24-bit RGB 8:8:8 color value.

CRT Palette RAM

Read/Write MMIO_base + 0x080C00 - 0x080FFC

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								Red R/W							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Green R/W								Blue R/W							

Bit(s)	Name	Description
31:24	Reserved	These bits are reserved.
23:16	Red	For indexed color modes: 8-bit red color value. For 16- and 32-bit color modes: 8-bit red alpha value.

Bit(s)	Name	Description
15:8	Green	For indexed color modes: 8-bit green color value. For 16- and 32-bit color modes: 8-bit green alpha value.
7:0	Blue	For indexed color modes: 8-bit blue color value. For 16- and 32-bit color modes: 8-bit blue alpha value.

There are 256 Video Palette RAM registers, each containing a 24-bit RGB 8:8:8 color value.

6 Command List Interpreter

Functional Overview

The Command List allows a number of tasks to be executed by a state machine to offload the CPU. Most of the commands are to write to registers and local memory in the SM501 memory map. There are also some flow commands that allow jumping to different locations in the Command List or calling common subroutines stored in the main memory.

The basic layout of an entry in the Command List looks like this:

Command **OPCODE**

63															**32**
Data															
31		**28**	**27**												**0**
				Address											

The command is a 4-bit field that is split into two regions. When bit 31 (bit 3 of the command) is 0, the command is to be executed. If bit 31 is 1, the specified command is a flow command and as a result the Command List FIFO will be flushed.

The Address field is specified in the SM501 Address Space. For internal memory, only bits 0 through 25 are used to address the 64MB address range. In this case, bits 26 and 27 are “0”. When bit 27 is set to “1”, the address space does not specify an internal memory address, but rather a memory address that lives on the host bus. Bits 0 through 26 specify a 128MB address range.

Programming

To execute the Command List, the CPU is first building a valid Command List structure and then programs the start address of the Command List in the Command List Start Address register. The Command List should be terminated by the FINISH command.

Flow Commands

Several commands can be used to change the flow of the Command List. There are GOTO and GOSUB commands, as well as a conditional JUMP command.

All destination addresses can be either relative or absolute. This makes it easy to jump over certain commands in the Command List or jump to common subroutines stored in main memory.

The conditional JUMP command can be used to test for any of the 32 software-programmable conditional states. If any of the requested conditional states is set, the jump is taken.

Appending to the Command List

The procedure for chaining command lists is:

1. Fill the command list buffer after the last FINISH command. The software should always keep track of the address of the last FINISH command.
2. Terminate the command list with a FINISH and remember the address of this FINISH.
3. Stop the command list by programming “0” in bit 31 of the Command List Address register.
4. Read and remember the current program counter.
5. Replace the previous FINISH command with a NOP command (00000000C0000000).
6. Restart the command list by programming the saved program counter and “1” in bit 31 of the Command List Address register.

Register Descriptions

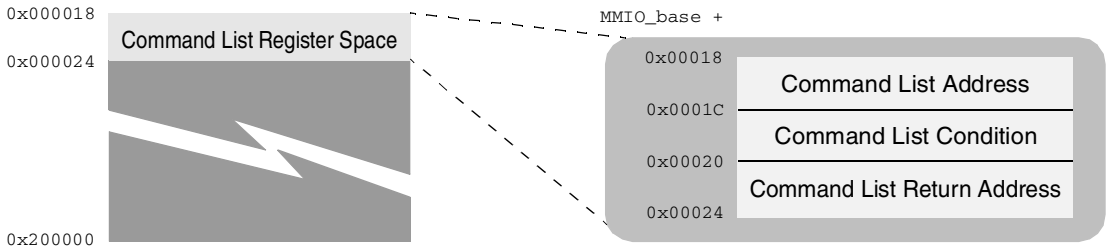
Table 6-1 summarizes the Command List registers.

Table 6-1: Command List Register Summary

Address	Type	Width	Reset Value	Register Name
0x000018	R/W	32	N/A	Command List Address
0x00001C	R/W	32	N/A	Command List Condition
0x000020	R	32	N/A	Command List Return Address

Figure 6-1 defines the register layout for the Command List registers.

Figure 6-1: Command List Register Space



Command List Address

Read/Write Address 0x000018

Power-on Default N/A

31	30	29	28	27								3	2		0
S R/W	Res	Res		Command List Address R/W									000		

Bit(s)	Name	Description
S	Start	When this bit is programmed to “1”, the Command List will fetch the first instruction from the Command List specified by the Command List Address field. It will remain “1” as long as the Command List is executing code in the Command List. As soon as you program this bit to “0”, the Command List will stop executing. Programming it back to “1” will continue the Command List at the address it has left off.
30	Idle	Idle status. 0: busy. 1: idle (default).
29:28	Res	These bits are reserved.
27:0	Command List Address	The current address of the Command List. The Command List updates this address continuously. Bits [2:0] are hardwired to “0” since every command must be aligned on a 64-bit boundary. It always points to the instruction being executed.

When the *Start* bit is programmed to “0”, the command interpreter will stop after the current command has been executed. This means the *Command List Address* (program counter) will contain the address of the next instruction that is going to be executed when the Start bit is programmed to “1” again.

When programming the *Start* bit to “0” when conditional jumps are being executed, the *Command List Address* contains the next logical address of the instruction to fetch, depending if the jump is taken or not.

Note: Note that a read of this register returns the Program Counter from the command list. This value is different than the value programmed into the *Command List Address* field. So if you want to restart the command list after a STOP (clearing the *S* bit), you need to program the correct Program Counter value into the *Command List Address* field. This normally is not a program since you need to do a read/modify/write instruction anyway to clear the *S* bit.

Command List Condition

Read/Write Address 0x00001C

Power-on Default N/A

31															0
Conditions R/W															

Bit(s)	Name	Description
31:0	Conditions	Every bit in the Conditions field holds one condition. The conditions are totally controlled by software. The Conditional Jump command will mask the requested condition with this Condition field. If any bit is set after this mask, the condition returns TRUE and the jump is taken.

Command List Return Address

Read Address 0x000020

Power-on Default N/A

31			28	27								3	2		0
Res				Return Address R									000		

Bit(s)	Name	Description
31:28	Res	These bits are reserved.
27:0	Return Address	The GOSUB command will store the address of the next command in the Command List in this register. The RETURN command will jump to the address specified in this register. Bits [2:0] are hardwired to “0” since every command must be aligned on a 64-bit boundary.

Commands

Table 6-2 lists the 16 commands recognized by the Command List Interpreter.

Table 6-2: SM501 Commands

0000	Load Memory	1000	Finish
0001	Load Register	1001	Goto
0010	Load Memory Immediate	1010	Gosub
0011	Load Register Immediate	1011	Return
0100	Load Memory Indirect	1100	Conditional Jump
0101	Load Register Indirect	1101	Reserved
0110	Status Test	1110	Reserved
0111	Reserved	1111	Reserved

Load Memory

Load Memory **0000b**

63	**62**	**61**													**32**
B2	B1	Data													
31		**28**	**27**											**1**	**0**
0000			Memory Address												W

Data — The data to be loaded in the memory address specified by Memory Address. The data format is either 32-bit DWords or 16-bit Words.

Memory Address — The Memory Address to write data to. Bits [3:0] are hardwired to “0” since all Memory Addresses should be 128-bit aligned.

W — When this bit is programmed to “0”, the 32-bit DWord data (bits [63:32]) is written to the Memory Address. When this bit is programmed to “1”, the 16-bit Word data (bits [47:32]) is written to the Memory Address.

B2, B1 — Bits [63:62] are the byte-enable signals for the Word data. They are active high.

Load Register

Load Register **0001b**

63															32
Data															
31		**28**	**27**										**2**	**1**	**0**
0001			Register Address											00	

Data The data to be loaded in the register specified by Register Address.

Register Address The register address (in the space `0x00000000` – `0x001FFFFF`) to write data to. Bits [0:1] are hardwired to "0" since all register addresses should be 32-bit aligned.

Load Memory Immediate

Load Memory Immediate **0010b**

63															32
DWORD Count															
31		**28**	**27**										**2**	**1**	**0**
0010			Memory Address											00	

DWORD Count The number of DWORDs to load into the memory.

Memory Address The starting memory address to write data to. Bits [1:0] are hardwired to "0".

The data that must be loaded into the memory directly follows this command. Make sure the correct number of DWORDs (*DWORD Count*) is provided, otherwise unpredicted results will happen. Also, if an odd number of DWORDs is specified, the last DWORD should be padded with a dummy DWORD to align the next command to 64-bit again.

Load Register Immediate

Load Register Immediate **0011b**

63															32
DWORD Count															
31		**28**	**27**										**2**	**1**	**0**
0011			Register Address											00	

DWORD Count The number of DWORDs to load into the registers.

Register Address The starting register address (in the space `0x00000000` – `0x001FFFFF`) to write data to. Bits [0:1] are hardwired to "0" since all register addresses should be 32-bit aligned.

The data that must be loaded into the registers directly follows this command. Make sure the correct number of DWORDs (*DWORD Count*) is provided, otherwise unpredicted results will happen. Also, if an odd number of DWORDs is specified, the last DWORD should be padded with a dummy DWORD to align the next command to 64-bit again.

Load Memory Indirect

Load Memory Indirect **0100b**

127															**96**
95		**92**	**91**										**66**	**65**	**64**
			Source Memory Address											00	
63															**32**
DWORD Count															
31		**28**	**27**										**2**	**1**	**0**
0100			Memory Address											00	

Source Memory Address The starting memory address to read data from. Bits [65:64] are hardwired to “0”.

DWORD Count The number of DWORDs to copy into the memory.

Memory Address The starting memory address to write data to. Bits [1:0] are hardwired to “0”.

This command copies data from the memory location specified by *Source Memory Address* into the memory location specified by *Memory Address*. The *DWORD Count* specifies the number of DWORDs to copy. This command is most useful to copy texture, bitmap, or vertex data to off-screen memory for caching purposes.

Load Register Indirect

Load Register Indirect **0101b**

127															**96**
95		**92**	**91**										**66**	**65**	**64**
			Source Memory Address											00	
63															**32**
DWORD Count															
31		**28**	**27**										**2**	**1**	**0**
0101			Register Address											00	

Source Memory Address The starting memory address to read data from. Bits [65:64] are hardwired to “0”.

DWORD Count The number of DWORDs to copy into the registers.

Register Address The starting register address (in the space `0x00000000` – `0x001FFFFF`) to write data to. Bits [1:0] are hardwired to “0” since all register addresses should be 32-bit aligned.

This command copies data from the memory location specified by *Source Memory Address* into the register bank location specified by *Register Address*. The *DWORD Count* specifies the number of DWORDs to copy. This command is most useful to copy texture, bitmap, or vertex data to the engine FIFOs for processing.

Status Test

Status Test **0110b**

63						53	52																	32
							Bit Values																	
31		28	27			21	20	19	18	17	16	15	14	13	12	11	10					3	2	1	0
0110							2_M	C_F	2_C	D_M	C_S	V_F	V_S	P_S	S_C	S_P							2_S	2_F	2_E

2D Memory FIFO (2_M)	2D and Color Space Conversion memory FIFO (0 = not empty, 1 = empty).
Command FIFO (C_F)	Command FIFO on HIF bus (0 = not empty, 1 = empty).
Color Space Conversion (2_C)	Color Space Conversion busy bit.
Memory DMA Busy (D_M)	Memory DMA busy bit.
CRT Status Bit (C_S)	CRT Graphics Layer status bit.
Current Field (V_F)	Current Video Layer field for BOB (0 = odd, 1 = even).
Video Status Bit (V_S)	Video Layer status bit.
Panel Status Bit (P_S)	Panel Graphics Layer status bit.
CRT Sync (S_C)	Vertical Sync for CRT pipe (0 = not active, 1 = active).
Panel Sync (S_P)	Vertical Sync for Panel pipe (0 = not active, 1 = active).
2D Setup (2_S)	2D Setup Engine (0 = idle, 1 = busy).
2D FIFO (2_F)	2D and Color Space Conversion command FIFO (0 = not empty, 1 = empty).
2D Engine (2_E)	2D Drawing Engine (0 = idle, 1 = busy).

The Status Test command will wait until the requested status is met. The value of the Status Test register is masked with the internal hardware state and compared to the state in the *Bit Values*. If the result does not equal the *Bit Values*, the command list interpreter will wait until the hardware status changes. The pseudo code looks like this:

WHILE (Hardware State *& Mask* [20:0] != *Bit Values* [52:32] & Mask [20:0]) NOP;

Finish

Finish **1000b**

63															32
31		**28**	**27**											**1**	**0**
1000															I

Interrupt (I) — If the *Interrupt* bit is set, the FINISH command will generate an interrupt that can still be masked by the *Command List* mask bit in the Interrupt Mask register. When an interrupt is generated, the *Command List* bit in Interrupt Status register will be set to "1".

The FINISH command stops executing commands in the Command List and clears the *Start* bit ([31]) of the Command List Address register.

Goto

Goto **1001b**

63														**33**	**32**
															R
31		**28**	**27**									**3**	**2**		**0**
1001			Address											000	

Relative (R) — If the *Relative* bit is set, the specified *Address* is relative to the address of the current command (signed addition).

Address — The address of the new code to execute. Bits [2:0] are hardwired to "0" since all addresses need to be 64-bit aligned.

The GOTO command will jump to the Command List code located at the specified *Address*.

Gosub

Gosub **1010b**

63														**33**	**32**
															R
31		**28**	**27**									**3**	**2**		**0**
1010			Address											000	

Relative (R) — If the *Relative* bit is set, the specified *Address* is relative to the address of the current command (signed addition).

Address — The address of the new code to execute. Bits [2:0] are hardwired to "0" since all addresses need to be 64-bit aligned.

The GOSUB command will store the address of the next instruction it would execute in the Command List Return Address register and starts executing the Command List code located at the specified *Address*.

Return

Return **1011b**

63															32
31		28	27												0
1011															

The RETURN command will jump to the address specified in the Command List Return Address register. The RETURN command should terminate a subroutine that is being called by GOSUB.

Conditional Jump

Conditional Jump **1100b**

63															32
Condition															
31		28	27									3	2		0
1100			Address										000		

Condition — The Condition field consists of a 32-bit mask that will be applied to the Command List Condition Register. If the result of this mask is TRUE (any bit set), the condition shall return TRUE and the jump is taken by adding the signed value of Address to the address of the next command in the Command List.
The formula of the condition is:
RESULT = Condition • Command List Condition register

Address — A signed relative value that will be added to the address of the next command in the Command List if the result of the condition is TRUE. Bits [2:0] are hardwired to “0” since all addresses need to be 64-bit aligned.

7 USB Host Controller

Functional Overview

The Host Controller (HC) is the device located between the USB bus and the Host Controller Driver (HCD) in the OpenHCI architecture. The SM501 USB Host Controller is compatible with USB Specification Revision 1.1 and OpenHCI Specification Revision 1.0. The Host Controller is charged with processing all of the Data Type lists built by the Host Controller Driver. It supports both low-speed and high-speed USB devices. Additionally, there is one USB Root Hub that is attached to the Host Controller.

Register Descriptions

This section describes the registers, their operations, and their options.

The Host Controller (HC) contains a set of on-chip operational registers which are mapped into a noncacheable portion of the system addressable space. These registers are used by the Host Controller Driver (HCD). According to the function of these registers, they are divided into four partitions, specifically for Control and Status, Memory Pointer, Frame Counter and Root Hub. All of the registers should be read and written as Dwords.

Reserved bits may be allocated in future releases of this specification. To ensure interoperability, the Host Controller Driver that does not use a reserved field should not assume that the reserved field contains 0. Furthermore, the Host Controller Driver should always preserve the value(s) of the reserved field. When a R/W register is modified, the Host Controller Driver should first read the register, modify the bits desired, then write the register with the reserved bits still containing the read value. Alternatively, the Host Controller Driver can maintain an in-memory copy of previously written values that can be modified and then written to the Host Controller register. When a write to set/clear register is written, bits written to reserved fields should be 0.

Table 7-1 summarizes the USB Host Controller Driver registers.

Table 7-1: USB Host Controller Driver Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Name	Description
Control and Status Registers					
0x040000	Read	8	10h	HcRevision	Contains BCD representation of version of HCI specification
0x040004	Read/Write	11	0b	HcControl	Defines operating modes for Host Controller
0x040008	Read/Write	6	0b	HcCommandStatus	Receives commands issued and reflects current status of Host Controller
0x04000C	Read/Write	9	0b	HcInterruptStatus	Provides Status on events that cause hardware interrupts
0x040010	Read/Write	9	0b	HcInterruptEnable	Controls which events generate a hardware interrupt

Table 7-1: USB Host Controller Driver Register Summary (Continued)

Offset from MMIO_base[1]	Type	Width	Reset Value	Name	Description
0x040014	Read/Write	9	0b	HcInterruptDisable	Coupled with HcInterruptEnable register
Memory Pointer Registers					
0x040018	Read/Write	32	0h	HcHCCA	Contains physical address of Host Controller Communication area
0x04001C	Read	32	0h	HcPeriodCurrentED	Contains physical address of current Isochronous or Interrupt Endpoint Descriptor
0x040020	Read/Write	32	0h	HcControlHeadED	Contains physical address of first Endpoint Descriptor of Control list
0x040024	Read/Write	32	0h	HcControlCurrentED	Contains physical address of current Endpoint Descriptor of Control list
0x040028	Read/Write	32	0h	HcBulkHeadED	Contains physical address of first Endpoint Descriptor of Bulk list
0x04002C	Read/Write	32	0h	HcBulkCurrentED	Contains physical address of current Endpoint Descriptor of Bulk list
0x040030	Read	32	0h	HcDoneHead	Contains physical address of last completed Transfer Descriptor added to done queue
Frame Counter Registers					
0x040034	Read/Write	30	Depends on key	HcFmInterval	Contains 14-bit value indicating a bit time interval in a Frame and a 15-bit value indicating the Full-Speed maximum packet size
0x040038	Read	14	Depends on key	HcFmRemaining	14-bit down counter showing bit time remaining in the current frame
0x04003C	Read	16	0h	HcFmNumber	16-bit counter providing a timing reference among events in Host Controller and Host Controller Driver
0x040040	Read/Write	14	0h	HcPeriodicStart	14-bit programmable value determining when earliest time HC starts processing periodic list

Table 7-1: USB Host Controller Driver Register Summary (Continued)

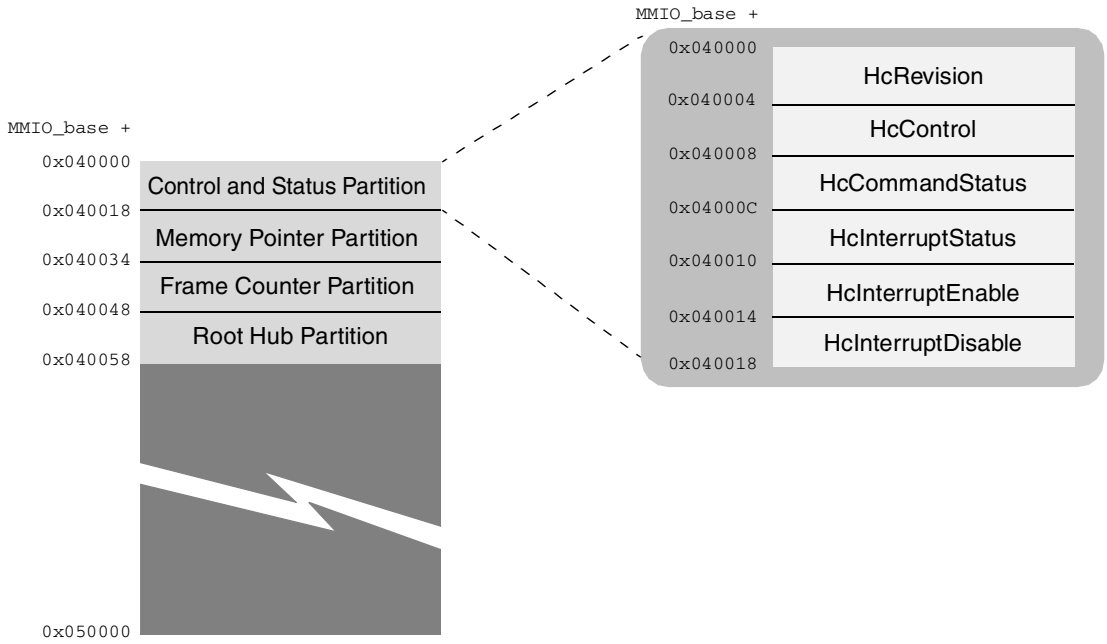
Offset from MMIO_base[1]	Type	Width	Reset Value	Name	Description
0x040044	Read/Write	12	0628h	HcLSThreshold	11-bit value determining whether to commit to transfer of a maximum of 8-byte LS packet
Root Hub Registers					
0x040048	Read/Write	21	Depends on key	HcRhDescriptorA	Describes characteristics of the root hub
0x04004C	Read/Write	32	IS[2]	HcRhDescriptorB	Describes characteristics of the root hub
0x040050	Read/Write	6	Depends on key	HcRhStatus	Lower word of a Dword represents Hub Status field and upper word represents Hub Status Change field
0x040054	Read/Write	12	Depends on key	HcRhPortStatus[1:0]	Control and report port events on a per-port basis

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. IS denotes an implementation-specific reset value for that field.

Control and Status Registers

Figure 7-1 shows the layout of the control and status partition registers.

Figure 7-1: USB Control and Status Partition Register Space



HcRevision

Read MMIO_base + 0x040000

Power-on Default: 10h--

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

Bit(s)	Name	Description
31:8	Reserved	These bits are reserved.
7:0	REV	Revision. This read-only field contains the BCD representation of the version of the HCI specification that is implemented by this HC. For example, a value of 0x11 corresponds to version 1.1. All of the HC implementations that are compliant with this specification will have a value of 0x10.

HcControl

Read/Write MMIO_base + 0x040004

Power-on Default: 00B for CBSR and HCFS, 0B otherwise

The *HcControl* register defines the operating modes for the Host Controller. Most of the fields in this register are modified only by the Host Controller Driver, except **HostControllerFunctionalState** and **RemoteWakeupConnected**.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved					RWE R/W R	RWC R/W R/W	IR R/W R	HCFS R/W R/W		BLE R/W R	CLE R/W R	IE R/W R	PLE R/W R	CBSR R/W R	

Bit(s)	Name	Description
31:11	Reserved	These bits are reserved.
10	RWE	RemoteWakeupEnable. Used by HCD to enable or disable the remote wakeup feature upon the detection of upstream resume signaling. When this bit is set and the **ResumeDetected** bit in *HcInterruptStatus* is set, a remote wakeup is signaled to the host system. Setting this bit has no impact on the generation of hardware interrupt.
9	RWC	RemoteWakeupConnected. Indicates whether HC supports remote wakeup signaling. If remote wakeup is supported and used by the system it is the responsibility of system firmware to set this bit during POST. HC clears the bit upon a hardware reset but does not alter it upon a software reset. Remote wakeup signaling of the host system is host-bus-specific and is not described in this specification.
8	IR	InterruptRouting. Determines the routing of interrupts generated by events registered in *HcInterruptStatus*. If clear, all interrupts are routed to the normal host bus interrupt mechanism. If set, interrupts are routed to the System Management Interrupt. HCD clears this bit upon a hardware reset, but it does not alter this bit upon a software reset. HCD uses this bit as a tag to indicate the ownership of HC.
7:6	HCFS	HostControllerFunctionalState. 00: USBReset 01: USBResume 10: USBOperational 11: USBSuspend A transition to UsbOperational from another state causes SOF generation to begin 1 ms later. HCD may determine whether HC has begun sending SOFs by reading the StartofFrame field of HcInterruptStatus. This field may be changed by HC only when in the UsbSuspend state. HC may move from the UsbSuspend state to the UsbResume state after detecting the resume signaling from a downstream port. HC enters UsbSuspend after a software reset, whereas it enters UsbReset after a hardware reset. The latter also resets the Root Hub and asserts subsequent reset signaling to downstream ports.
5	BLE	BulkListEnable. Set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, processing of the Bulk list does not occur after the next SOF. HC checks this bit whenever it determines to process the list. When disabled, HCD may modify the list. If *HcBulkCurrentED* is pointing to an ED to be removed, HCD must advance the pointer by updating *HcBulkCurrentED* before re-enabling processing of the list.

Bit(s)	Name	Description
4	CLE	ControlListEnable. Set to enable the processing of the Control list in the next Frame. If cleared by HCD, processing of the Control list does not occur after the next SOF. HC must check this bit whenever it determines to process the list. When disabled, HCD may modify the list. If *HcControlCurrentED* is pointing to an ED to be removed, HCD must advance the pointer by updating *HcControlCurrentED* before re-enabling processing of the list.
3	IE	IsochronousEnable. Used by HCD to enable/disable processing of isochronous EDs. While processing the periodic list in a Frame, HC checks the status of this bit when it finds an Isochronous ED (F=1). If set (enabled), HC continues processing the EDs. If cleared (disabled), HC halts processing of the periodic list (which now contains only isochronous EDs) and begins processing the Bulk/Control lists. Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).
2	PLE	PeriodicListEnable. Set to enable the processing of the periodic list in the next Frame. If cleared by HCD, processing of the periodic list does not occur after the next SOF. HC must check this bit before it starts processing the list.
1:0	CBSR	ControlBulkServiceRatio. Specifies the service ratio between Control and Bulk EDs. Before processing any of the nonperiodic lists, HC must compare the ratio specified with its internal count on how many nonempty Control EDs have been processed, in determining whether to continue serving another Control ED or switching to Bulk EDs. The internal count will be retained when crossing the frame boundary. In case of reset, HCD is responsible for restoring this value.

CBSR	No. of Control EDs Over Bulk EDs Served
0	1:1
1	2:1
2	3:1
3	4:1

HcCommandStatus

Read/Write MMIO_base + `0x040008`

Power-on Default: `00B` for SOC and HCFS, `0B` otherwise

The *HcCommandStatus* register is used by the Host Controller to receive commands issued by the Host Controller Driver, as well as reflecting the current status of the Host Controller. To the Host Controller Driver, it appears to be a "write to set" register. The Host Controller must ensure that bits written as '1' become set in the register while bits written as '0' remain unchanged in the register. The Host Controller Driver may issue multiple distinct commands to the Host Controller without concern for corrupting previously issued commands. The Host Controller Driver has normal read access to all bits.

The **SchedulingOverrunCount** field indicates the number of frames with which the Host Controller has detected the scheduling overrun error. This occurs when the Periodic list does not complete before EOF. When a scheduling overrun error is detected, the Host Controller increments the counter and sets the **SchedulingOverrun** field in the *HcInterruptStatus* register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														SOC R R/W	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved												OCR R/W R/W	BLF R/W R/W	CLF R/W R/W	HCR R/W R/W

Bit(s)	Name	Description
31:18	Reserved	These bits are reserved.
17:16	SOC	SchedulingOverrunCount. Incremented on each scheduling overrun error. It is initialized to 00b and wraps around at 11b. This will be incremented when a scheduling overrun is detected even if **SchedulingOverrun** in *HcInterruptStatus* has already been set. This is used by HCD to monitor any persistent scheduling problems.
15:4	Reserved	These bits are reserved.
3	OCR	OwnershipChangeRequest. Set by an OS HCD to request a change of control of the HC. When set HC will set the **OwnershipChange** field in *HcInterruptStatus*. After the changeover, this bit is cleared and remains so until the next request from OS HCD.
2	BLF	BulkListFilled. Used to indicate whether there are any TDs on the Bulk list. It is set by HCD whenever it adds a TD to an ED in the Bulk list. When HC begins to process the head of the Bulk list, it checks BF. As long as **BulkListFilled** is 0, HC will not start processing the Bulk list. If **BulkListFilled** is 1, HC will start processing the Bulk list and will set BF to 0. If HC finds a TD on the list, then HC will set **BulkListFilled** to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if HCD does not set **BulkListFilled**, then **BulkListFilled** will still be 0 when HC completes processing the Bulk list and Bulk list processing will stop.
1	BLF	ControlListFilled. Used to indicate whether there are any TDs on the Control list. It is set by HCD whenever it adds a TD to an ED in the Control list. When HC begins to process the head of the Control list, it checks CLF. As long as **ControlListFilled** is 0, HC will not start processing the Control list. If CF is 1, HC will start processing the Control list and will set **ControlListFilled** to 0. If HC finds a TD on the list, then HC will set **ControlListFilled** to 1 causing the Control list processing to continue. If no TD is found on the Control list, and if the HCD does not set **ControlListFilled**, then **ControlListFilled** will still be 0 when HC completes processing the Control list and Control list processing will stop.
0	HCR	HostControllerReset. Set by HCD to initiate a software reset of HC. Regardless of the functional state of HC, it moves to the UsbSuspend state in which most of the operational registers are reset except those stated otherwise; e.g., the InterruptRouting field of HcControl, and no Host bus accesses are allowed. This bit is cleared by HC upon the completion of the reset operation. The reset operation must be completed within 10 ms. This bit, when set, should not cause a reset to the Root Hub and no subsequent reset signaling should be asserted to its downstream ports.

HcInterruptStatus

Read/Write MMIO_base + 0x04000C

Power-on Default: 0B

This register provides status on various events that cause hardware interrupts. When an event occurs, Host Controller sets the corresponding bit in this register. When a bit becomes set, a hardware interrupt is generated if the interrupt is enabled in the *HcInterruptEnable* register (see page 7-9) and the **MasterInterruptEnable** bit is set. The Host Controller Driver may clear specific bits in this register by writing '1' to bit positions to be cleared. The Host Controller Driver may not set any of these bits. The Host Controller will never clear the bit.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0 R/W R/W	OC R/W R/W	Reserved													
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved									RHSC R/W R/W	FNO R/W R/W	UE R/W R/W	RD R/W R/W	SF R/W R/W	WDH R/W R/W	SO R/W R/W

Bit(s)	Name	Description
31	0	This bit is hardwired to zero.
30	OC	Ownership Change. Set by HC when HCD sets the OwnershipChangeRequest field in *HcCommandStatus*. This event, when unmasked, will always generate an System Management Interrupt (SMI) immediately. Tied to 0 when the SMI pin is not implemented.
29:7	Reserved	These bits are reserved.
6	RHSC	RootHubStatusChange. Set when the content of *HcRhStatus* or the content of any of *HcRhPortStatus*[**NumberofDownstreamPort**] has changed.
5	FNO	FrameNumberOverflow. Set when the MSb of *HcFmNumber* (bit 15) changes value, from 0 to 1 or from 1 to 0, and after *HccaFrameNumber* has been updated.
4	UE	UnrecoverableError. Set when HC detects a system error not related to USB. HC should not proceed with any processing nor signaling before the system error has been corrected. HCD clears this bit after HC has been reset.
3	RD	ResumeDetected. Set when HC detects that a device on the USB is asserting resume signaling. It is the transition from no resume signaling to resume signaling causing this bit to be set. This bit is not set when HCD sets the UsbResume state.
2	SF	StartofFrame. Set by HC at each start of a frame and after the update of *HccaFrameNumber*. HC also generates a SOF token at the same time.
1	WDH	WritebackDoneHead. Set immediately after HC has written *HcDoneHead* to *HccaDoneHead*. Further updates of the *HccaDoneHead* will not occur until this bit has been cleared. HCD should only clear this bit after it has saved the content of *HccaDoneHead*.
0	SO	SchedulingOverrun. Set when the USB schedule for the current Frame overruns and after the update of *HccaFrameNumber*. A scheduling overrun will also cause the **SchedulingOverrunCount** of *HcCommandStatus* to be incremented.

HcInterruptEnable

Read/Write MMIO_base + 0x040010

Power-on Default: 0B

Each enable bit in the *HcInterruptEnable* register corresponds to an associated interrupt bit in the *HcInterruptStatus* register. The *HcInterruptEnable* register is used to control which events generate a hardware interrupt. When a bit is set in the *HcInterruptStatus* register AND the corresponding bit in the *HcInterruptEnable* register is set AND the **MasterInterruptEnable** bit is set, then a hardware interrupt is requested on the host bus.

Writing a '1' to a bit in this register sets the corresponding bit, whereas writing a '0' to a bit in this register leaves the corresponding bit unchanged. On read, the current value of this register is returned.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MIE R/W R	OC R/W R	Reserved													
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved									RHSC R/W R	FNO R/W R	UE R/W R	RD R/W R	SF R/W R	WDH R/W R	SO R/W R

Bit(s)	Name	Description
31	MIE	MasterInterruptEnable. A ‘0’ written to this field is ignored by HC. A '1' written to this field enables interrupt generation due to events specified in the other bits of this register. This is used by HCD as a Master Interrupt Enable.
30	OC	Ownership Change. 0: Ignore. 1: Enable interrupt generation due to Ownership Change.
29:7	Reserved	These bits are reserved.
6	RHSC	RootHubStatusChange. 0: Ignore. 1: Enable interrupt generation due to Root Hub Status Change.
5	FNO	FrameNumberOverflow. 0: Ignore. 1: Enable interrupt generation due to Frame Number Overflow.
4	UE	UnrecoverableError. 0: Ignore. 1: Enable interrupt generation due to Unrecoverable Error.
3	RD	ResumeDetected. 0: Ignore. 1: Enable interrupt generation due to Resume Detected.
2	SF	StartofFrame. 0: Ignore. 1: Enable interrupt generation due to Ownership Change.

Bit(s)	Name	Description
1	WDH	WritebackDoneHead. 0: Ignore. 1: Enable interrupt generation due to HCDoneHead Writeback.
0	SO	SchedulingOverrun. 0: Ignore. 1: Enable interrupt generation due to Scheduling Overrun.

HcInterruptDisable

Read/Write MMIO_base + 0x040014

Power-on Default: 0B

Each disable bit in the *HcInterruptDisable* register corresponds to an associated interrupt bit in the *HcInterruptStatus* register. The *HcInterruptDisable* register is coupled with the *HcInterruptEnable* register. Thus, writing a '1' to a bit in this register clears the corresponding bit, whereas writing a '0' to a bit in this register leaves the corresponding bit in the *HcInterruptEnable* register unchanged. On read, the current value of the *HcInterruptEnable* register is returned.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MIE R/WR	OC R/WR	Reserved													
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved									RHSC R/WR	FNO R/WR	UE R/WR	RD R/WR	SF R/WR	WDH R/WR	SO R/WR

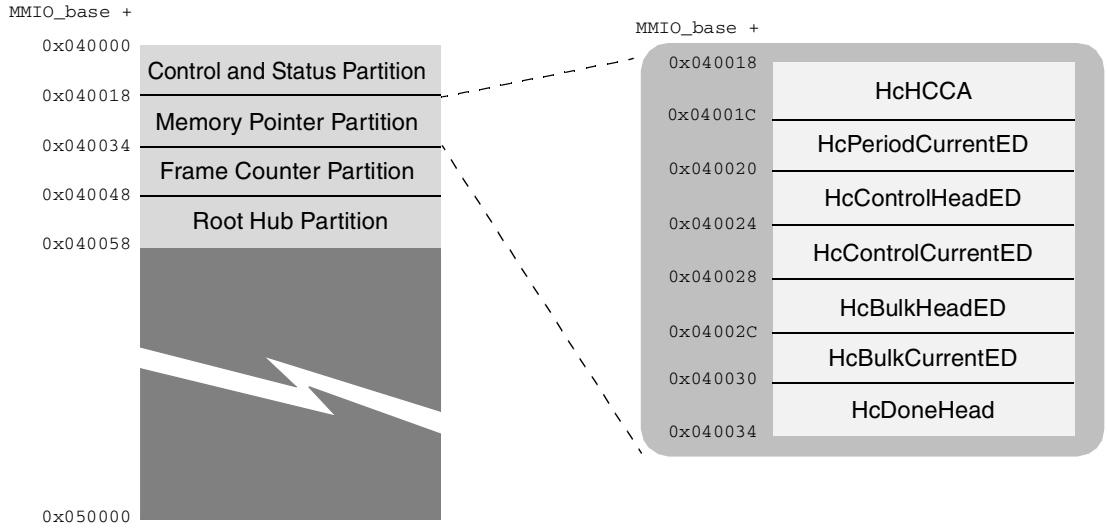
Bit(s)	Name	Description
31	MIE	MasterInterruptEnable. A ‘0’ written to this field is ignored by HC. A '1' written to this field disables interrupt generation due to events specified in the other bits of this register. This field is set after a hardware or software reset.
30	OC	OwnershipChange. 0: Ignore. 1: Disable interrupt generation due to Ownership Change.
29:7	Reserved	These bits are reserved.
6	RHSC	RootHubStatusChange. 0: Ignore. 1: Disable interrupt generation due to Root Hub Status Change.
5	FNO	FrameNumberOverflow. 0: Ignore. 1: Disable interrupt generation due to Frame Number Overflow.
4	UE	UnrecoverableError. 0: Ignore. 1: Disable interrupt generation due to Unrecoverable Error.
3	RD	ResumeDetected. 0: Ignore. 1: Disable interrupt generation due to Resume Detected.

Bit(s)	Name	Description
2	SF	StartofFrame. 0: Ignore. 1: Disable interrupt generation due to Ownership Change.
1	WDH	WritebackDoneHead. 0: Ignore. 1: Disable interrupt generation due to HCDoneHead Writeback.
0	SO	SchedulingOverrun. 0: Ignore. 1: Disable interrupt generation due to Scheduling Overrun.

Memory Pointer Registers

Figure 7-2 shows the layout of the memory pointer partition registers.

Figure 7-2: USB Memory Pointer Partition Register Space



HcHCCA

Read/Write MMIO_base + 0x040018

Power-on Default: 0h

The *HcHCCA* register contains the physical address of the Host Controller Communication Area. The Host Controller Driver determines the alignment restrictions by writing all 1s to *HcHCCA* and reading the content of *HcHCCA*. The alignment is evaluated by examining the number of zeroes in the lower order bits. The minimum alignment is 256 bytes; therefore, bits 0 through 7 must always return '0' when read. This area is used to hold the control structures and the Interrupt table that are accessed by both the Host Controller and the Host Controller Driver.

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

Bit(s)	Name	Description
31:8	HCCA	Host Controller Communication Area. The base address of the Host Controller Communication Area.
7:0	0	These bits are hardwired to zeros.

HcPeriodCurrentED

Read/Write MMIO_base + 0x04001C

Power-on Default: 0h

The *HcPeriodCurrentED* register contains the physical address of the current Isochronous or Interrupt Endpoint Descriptor.

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

Bit(s)	Name	Description
31:4	PCED	PeriodCurrentED. Used by the HC to point to the head of one of the Periodic lists that will be processed in the current Frame. The content of this register is updated by the HC after a periodic ED has been processed. HCD may read the content in determining which ED is currently being processed at the time of reading.
3:0	0	These bits are hardwired to zeros.

HcControlHeadED

Read/Write MMIO_base + 0x040020

Power-on Default: 0h

The *HcControlHeadED* register contains the physical address of the first Endpoint Descriptor of the Control list.

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

Bit(s)	Name	Description
31:4	PCED	ControlHeadED. The HC traverses the Control list starting with the HcControlHeadED pointer. The content is loaded from HCCA during the initialization of HC. HCD may read the content in determining which ED is currently being processed at the time of reading.
3:0	0	These bits are hardwired to zeros.

HcControlCurrentED

Read/Write MMIO_base + 0x040024

Power-on Default: 0h

This register contains the physical address of the first Endpoint Descriptor of the Control list.

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

Bit(s)	Name	Description
31:4	PCED	ControlCurrentHeadED. This pointer is advanced to the next ED after serving the present one.The HC continues processing the list from where it left off in the last Frame. When it reaches the end of the Control list, the HC checks the **ControlListFilled** of in *HcCommandStatus*. If set, it copies the contents of *HcControlHeadED* to *HcControlCurrentED* and clears the bit. If not set, it does nothing. The HCD is allowed to modify this register only when the **ControlListEnable** of *HcControl* is cleared. When set, HCD only reads the instantaneous value of this register. Initially, this is set to zero to indicate the end of the Control list.
3:0	0	These bits are hardwired to zeros.

HcBulkHeadED

Read/Write MMIO_base + 0x040028

Power-on Default: 0h

The *HcBulkHeadED* register contains the physical address of the first Endpoint Descriptor of the Bulk list.

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

Bit(s)	Name	Description
31:4	BHED	BulkHeadED. The HC traverses the Bulk list starting with the *HcBulkHeadED* pointer. The content is loaded from HCCA during the initialization of the HC.
3:0	0	These bits are hardwired to zeros.

HcBulkCurrentED

Read/Write MMIO_base + 0x04002C

Power-on Default: 0h

The *HcBulkCurrentED* register contains the physical address of the current endpoint of the Bulk list. As the Bulk list is served in a round-robin fashion, the endpoints are ordered according to their insertion to the list.

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

Bit(s)	Name	Description
31:4	BCED	BulkCurrentED. This is advanced to the next ED after the HC has served the present one. HC continues processing the list from where it left off in the last Frame. When it reaches the end of the Bulk list, the HC checks the **ControlListFilled** of *HcControl*. If set, it copies the content of *HcBulkHeadED* to *HcBulkCurrentED* and clears the bit. If it is not set, it does nothing. The HCD is only allowed to modify this register when the **BulkListEnable** of *HcControl* is cleared. When set, the HCD only reads the instantaneous value of this register. This is initially set to zero to indicate the end of the Bulk list.
3:0	0	These bits are hardwired to zeros.

HcDoneHead

Read/Write MMIO_base + 0x040030

Power-on Default: 0h

The *HcDoneHead* register contains the physical address of the last completed Transfer Descriptor that was added to the Done queue. In normal operation, the Host Controller Driver should not need to read this register as its content is periodically written to the HCCA.

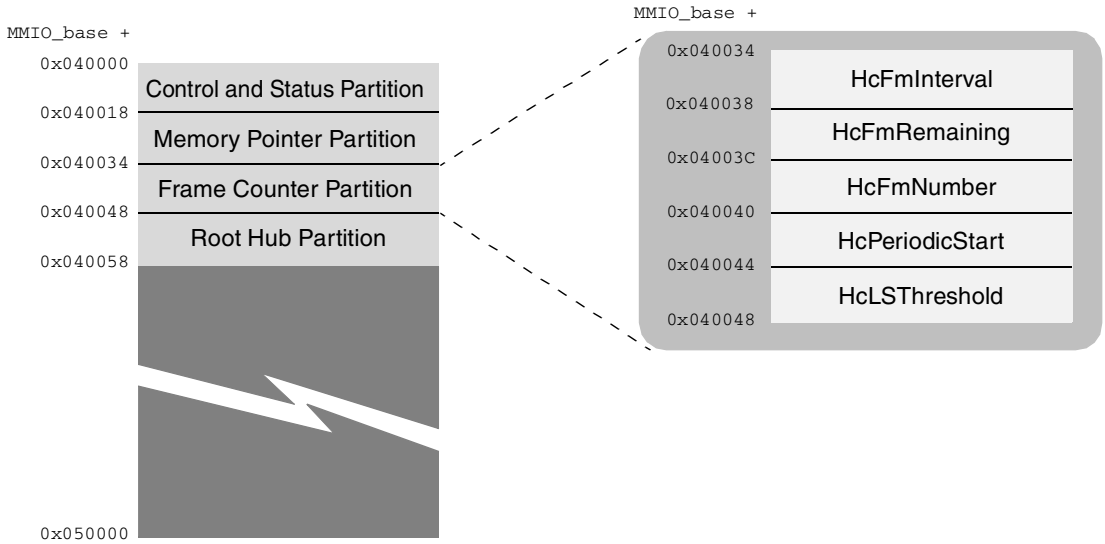
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DH R/W R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DH R/W R/W												0			

Bit(s)	Name	Description
31:4	DH	DoneHead. When a TD is completed, the HC writes the content of *HcDoneHead* to the NextTD field of the TD. The HC then overwrites the content of *HcDoneHead* with the address of this TD. This is set to zero whenever the HC writes the content of this register to HCCA. It also sets the **WritebackDoneHead** of *HcInterruptStatus*.
3:0	0	These bits are hardwired to zeros.

Frame Counter Registers

Figure 7-3 shows the layout of the frame counter partition registers.

Figure 7-3: USB Frame Counter Partition Register Space



HcFmInterval

Read/Write MMIO_base + 0x040034

Power-on Default: 2EDFh for FI, TBD for FSMPS, 0b for FIT

The *HcFmInterval* register contains a 14-bit value which indicates the bit time interval in a Frame, (i.e., between two consecutive SOFs), and a 15-bit value indicating the Full Speed maximum packet size that the Host Controller may transmit or receive without causing scheduling overrun. The Host Controller Driver may carry out minor adjustment on the **FrameInterval** by writing a new value over the present one at each SOF. This provides the programmability necessary for the Host Controller to synchronize with an external clocking resource and to adjust any unknown local clock offset.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIT R/W R	FSMPS R/W R														
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved		FI R/W R													

Bit(s)	Name	Description
31	FIT	FrameInterval. The HCD toggles this bit whenever it loads a new value to **FrameInterval**.
30:16	FSMPS	FSLargeDataPacket. Specifies a value which is loaded into the Largest Data Packet Counter at the beginning of each frame. The counter value represents the largest amount of data in bits which can be sent or received by the HC in a single transaction at any given time without causing scheduling overrun. The field value is calculated by the HCD.
15:14	Reserved	These bits are reserved.
13:0	FI	FrameInterval. Specifies the interval between two consecutive SOFs in bit times. The nominal value is set to be 11,999. The HCD should store the current value of this field before resetting HC. By setting the **HostControllerReset** field of *HcCommandStatus* as this will cause the HC to reset this field to its nominal value. The HCD may choose to restore the stored value upon the completion of the Reset sequence.

HcFmRemaining

Read/Write MMIO_base + `0x040038`

Power-on Default: `0h` for FR, `0b` for FRT

The *HcFmRemaining* register is a 14-bit down counter showing the bit time remaining in the current Frame.

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

Bit(s)	Name	Description
31	FRT	FrameRemainingToggle. This bit is loaded from the **FrameIntervalToggle** field of *HcFmInterval* whenever **FrameRemaining** reaches 0. This bit is used by HCD for the synchronization between **FrameInterval** and **FrameRemaining**.
30:14	Reserved	These bits are reserved.
13:0	FR	FrameRemaining. This counter is decremented at each bit time. When it reaches zero, it is reset by loading the **FrameInterval** value specified in *HcFmInterval* at the next bit time boundary. When entering the UsbOperational state, the HC reloads the content with the **FrameInterval** of *HcFmInterval* and uses the updated value from the next SOF.

HcFmNumber

Read/Write MMIO_base + 0x04003C

Power-on Default: 0h

The *HcFmNumber* register is a 16-bit counter. It provides a timing reference among events happening in the Host Controller and the Host Controller Driver. The Host Controller Driver may use the 16-bit value specified in this register and generate a 32-bit frame number without requiring frequent access to the register.

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	FN	FrameNumber. Incremented when *HcFmRemaining* is reloaded. It will be rolled over to 0x0 after 0xFFFF. When entering the UsbOperational state, this will be incremented automatically. The content will be written to HCCA after the HC has incremented the FrameNumber at each frame boundary and sent a SOF but before the HC reads the first ED in that Frame. After writing to HCCA, the HC will set the StartofFrame in *HcInterruptStatus*.

HcPeriodicStart

Read/Write MMIO_base + 0x040040

Power-on Default: 0h

The *HcPeriodicStart* register has a 14-bit programmable value which determines when is the earliest time HC should start processing the periodic list.

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

Bit(s)	Name	Description
31:14	Reserved	These bits are reserved.
13:0	PS	PeriodicStart. After a hardware reset, this field is cleared. This is then set by the HCD during the HC initialization. The value is calculated roughly as 10% off from *HcFmInterval*. A typical value will be 0x3E67. When *HcFmRemaining* reaches the value specified, processing of the periodic lists have priority over Control/Bulk processing. The HC therefore starts processing the Interrupt list after completing the current Control or Bulk transaction that is in progress.

HcLSThreshold

Read/Write MMIO_base + 0x040044

Power-on Default: 0628h

The *HcLSThreshold* register contains an 11-bit value used by the Host Controller to determine whether to commit to the transfer of a maximum of 8-byte LS packet before EOF. Neither the Host Controller nor the Host Controller Driver are allowed to change this value.

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

Bit(s)	Name	Description
31:12	Reserved	These bits are reserved.
11:0	LST	LSThreshold. Contains a value that is compared to the FrameRemaining field prior to initiating a Low Speed transaction. The transaction is started only if FrameRemaining ≥ this field. The value is calculated by the HCD with the consideration of transmission and setup overhead.

Root Hub Registers

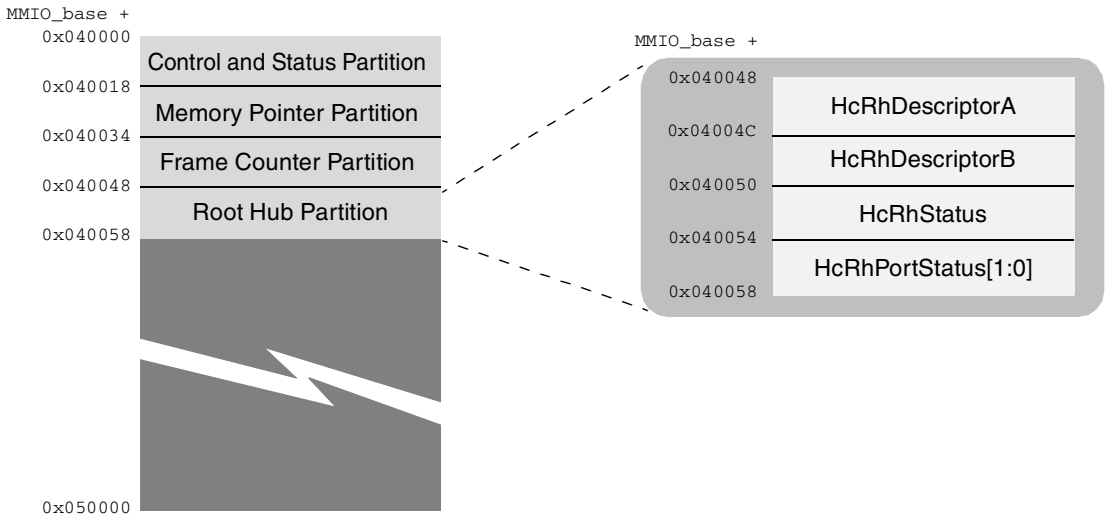
All registers included in this partition are dedicated to the USB Root Hub which is an integral part of the Host Controller though still a functionally separate entity. The HCD emulates USBD accesses to the Root Hub via a register interface. The HCD maintains many USB-defined hub features which are not required to be supported in hardware. For example, the Hub's Device, Configuration, Interface, and Endpoint Descriptors are maintained only in the HCD as well as some static fields of the Class Descriptor. The HCD also maintains and decodes the Root Hub's device address as well as other trivial operations which are better suited to software than hardware.

The Root Hub register interface is otherwise developed to maintain similarity of bit organization and operation to typical hubs which are found in the system. Below are four register definitions: HcRhDescriptorA, HcRhDescriptorB, HcRhStatus, and HcRhPortStatus[1:0]. Each register is read and written as a Dword. These registers are only written during initialization to correspond with the system implementation. The HcRhDescriptorA and HcRhDescriptorB registers should be implemented such that they are writable regardless of the HC USB state. HcRhStatus and HcRhPortStatus must be writable during the UsbOperational state.

Note: IS denotes an implementation-specific reset value for that field.

Figure 7-4 shows the layout of the root hub partition registers.

Figure 7-4: USB Root Hub Partition Register Space



HcRhDescriptorA

Read/Write MMIO_base + 0x040048

Power-on Default: 0b for DT, IS otherwise

The *HcRhDescriptorA* register is the first register of two describing the characteristics of the Root Hub. Reset values are implementation-specific. The descriptor length (11), descriptor type (TBD), and hub controller current (0) fields of the hub Class Descriptor are emulated by the HCD. All other fields are located in the *HcRhDescriptorA* and *HcRhDescriptorB* registers.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
POTPGT IS R/W								Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			NOCP IS R/W	OCPM IS R/W	DT 0b R	NPS IS R/W	PSM IS R/W	NDP R R							

Bit(s)	Name	Description
31:24	POTPGT	PowerOnToPowerGoodTime. Specifies the duration HCD has to wait before accessing a powered-on port of the Root Hub. It is implementation-specific. The unit of time is 2 ms. The duration is calculated as **POTPGT** * 2 ms.
23:13	Reserved	These bits are reserved.
12	NOCP	NoOverCurrentProtection. Describes how the overcurrent status for the Root Hub ports are reported. When this bit is cleared, the **OverCurrentProtectionMode** field specifies global or per-port reporting. 0: Overcurrent status is reported collectively for all downstream ports. 1: No overcurrent protection supported.
11	OCPM	OverCurrentProtectionMode. Describes how the overcurrent status for the Root Hub ports are reported. At reset, this fields should reflect the same mode as **PowerSwitchingMode**. This field is valid only if the **NoOverCurrentProtection** field is cleared. 0: Overcurrent status is reported collectively for all downstream ports. 1: Overcurrent status is reported on a per-port basis.
10	DT	DeviceType. Specifies that the Root Hub is not a compound device. The Root Hub is not permitted to be a compound device. This field should always read/write 0.
9	NPS	NoPowerSwitching. Used to specify whether power switching is supported or port are always powered. It is implementation-specific. When this bit is cleared, the **PowerSwitchingMode** specifies global or per-port switching. 0: Ports are power switched. 1: Ports are always powered on when the HC is powered on.
8	PSM	PowerSwitchingMode. This bit is used to specify how the power switching of the Root Hub ports is controlled. It is implementation-specific. This field is only valid if the **NoPowerSwitching** field is cleared. 0: All ports are powered at the same time. 1: Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the **PortPowerControlMask** bit is set, the port responds only to port power commands (**Set/ClearPortPower**). If the port mask is cleared, then the port is controlled only by the global power switch (**Set/ClearGlobalPower**).
7:0	NDP	NumberDownstreamPorts. These bits specify the number of downstream ports supported by the Root Hub. It is implementation-specific. The minimum number of ports is 1. The maximum number of ports supported by OpenHCI is 15.

HcRhDescriptorB

Read/Write MMIO_base + 0x04004C

Power-on Default: IS

The *HcRhDescriptorB* register is the second register of two describing the characteristics of the Root Hub. These fields are written during initialization to correspond with the system implementation. Reset values are implementation-specific.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PPCM R/W R															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DR R/W R															

Bit(s)	Name	Description
31:16	PPCM	PortPowerControlMask. Each bit indicates if a port is affected by a global power control command when **PowerSwitchingMode** is set. When set, the port's power state is only affected by per-port power control (**Set/ClearPortPower**). When cleared, the port is controlled by the global power switch (**Set/ClearGlobalPower**). If the device is configured to global switching mode (**PowerSwitchingMode**=0), this field is not valid. 0: Reserved. 1: Ganged-power mask on Port #1. 2: Ganged-power mask on Port #2. ... 15: Ganged-power mask on Port #15.
15:0	DR	DeviceRemovable. Each bit is dedicated to a port of the Root Hub. When cleared, the attached device is removable. When set, the attached device is not removable. 0: Reserved. 1: Device attached to Port #1. 2: Device attached to Port #2. ... 15: Device attached to Port #15.

HcRhStatus

Read/Write MMIO_base + 0x040050

Power-on Default: 0B

The *HcRhStatus* register is divided into two parts. The lower word of a Dword represents the **Hub Status** field and the upper word represents the **Hub Status Change** field. Reserved bits should always be written '0'.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRWE W R	Reserved													OCIC R/W R/W	LPSC R/W R
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DRWE R/W R	Reserved													OCI R R/W	LPS R/W R

Bit(s)	Name	Description
31	CRWE	ClearRemoteWakeupEnable. Writing a '1' clears **DeviceRemoveWakeupEnable**. Writing a '0' has no effect.
30:18	Reserved	These bits are reserved.
17	OCIC	OverCurrentIndicatorChange. Set by hardware when a change has occurred to the OCI field of this register. The HCD clears this bit by writing a ‘1’. Writing a ‘0’ has no effect.
16	LPSC	LocalPowerStatusChange (read). The Root Hub does not support the local power status feature; thus, this bit is always read as ‘0’. SetGlobalPower (write). In global power mode (**PowerSwitchingMode**=0), This bit is written to ‘1’ to turn on power to all ports (clear **PortPowerStatus**). In per-port power mode, it sets **PortPowerStatus** only on ports whose **PortPowerControlMask** bit is not set. Writing a ‘0’ has no effect.
15	DRWE	DeviceRemoteWakeupEnable (read). This bit enables a ConnectStatusChange bit as a resume event, causing a UsbSuspend to UsbResume state transition and setting the ResumeDetected interrupt. 0: **ConnectStatusChange** is not a remote wakeup event. 1: **ConnectStatusChange** is a remote wakeup event. SetRemoteWakeupEnable (write). Writing a '1' sets **DeviceRemoveWakeupEnable**. Writing a '0' has no effect.
14:2	Reserved	These bits are reserved.
1	OCI	OverCurrentIndicator. Reports overcurrent conditions when the global reporting is implemented. When set, an overcurrent condition exists. When cleared, all power operations are normal. If per-port overcurrent protection is implemented this bit is always ‘0’.
0	LPS	LocalPowerStatus (read). The Root Hub does not support the local power status feature; thus, this bit is always read as ‘0’. ClearGlobalPower (write). In global power mode (**PowerSwitchingMode**=0), This bit is written to ‘1’ to turn off power to all ports (clear **PortPowerStatus**). In per-port power mode, it clears **PortPowerStatus** only on ports whose **PortPowerControlMask** bit is not set. Writing a ‘0’ has no effect.

HcRhPortStatus[1:NDP]

Read/Write MMIO_base + 0x040054

Power-on Default XB for LSDA, 0B otherwise

The *HcRhPortStatus*[1:0] register is used to control and report port events on a per-port basis. **NumberDownstreamPorts** represents the number of *HcRhPortStatus* registers that are implemented in hardware. The lower word is used to reflect the port status, whereas the upper word reflects the status change bits. Some status bits are implemented with special write behavior (see below). If a transaction (token through handshake) is in progress when a write to change port status occurs, the resulting port status change must be postponed until the transaction completes. Reserved bits should always be written '0'.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved											PRSC R/W R/W	OCIC R/W R/W	PSSC R/W R/W	PESC R/W R/W	CSC R/W R/W
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved						LSDA R/W R/W	PPS R/W R/W	Reserved			PRS R/W R/W	POCI R/W R/W	PSS R/W R/W	PES R/W R/W	CCS R/W R/W

Bit(s)	Name	Description
31:21	Reserved	These bits are reserved.
20	PRSC	PortResetStatusChange. Set at the end of the 10-ms port reset signal. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. 0: Port reset is not complete. 1: Port reset is complete.
19	OCIC	PortOverCurrentIndicatorChange. Valid only if overcurrent conditions are reported on a per-port basis. This bit is set when Root Hub changes the **PortOverCurrentIndicator** bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. 0: No change in **PortOverCurrentIndicator**. 1: **PortOverCurrentIndicator** has changed.
18	PSSC	PortSuspendStatusChange. Set when the full resume sequence has been completed. This sequence includes the 20-s resume pulse, LS EOP, and 3-ms resychronization delay. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. This bit is also cleared when **ResetStatusChange** is set. 0: Resume is not completed. 1: Resume completed.
17	PESC	PortEnableStatusChange. Set when hardware events cause the PortEnableStatus bit to be cleared. Changes from HCD writes do not set this bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. 0: No change in **PortEnableStatus**. 1: Change in **PortEnableStatus**.
16	CSC	ConnectStatusChange. Set whenever a connect or disconnect event occurs. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. If **CurrentConnectStatus** is cleared when a **SetPortReset**, **SetPortEnable**, or **SetPortSuspend** write occurs, this bit is set to force the driver to re-evaluate the connection status since these writes should not occur if the port is disconnected. 0: No change in **CurrentConnectStatus**. 1: Change in **CurrentConnectStatus**. **Note:** If the **DeviceRemovable[NDP]** bit is set, this bit is set only after a Root Hub reset to inform the system that the device is attached.

Bit(s)	Name	Description
15:10	Reserved	Reserved.
9	LSDA	LowSpeedDeviceAttached (read). Indicates the speed of the device attached to this port. When set, a Low Speed device is attached to this port. When clear, a Full Speed device is attached to this port. This field is valid only when the **CurrentConnectStatus** is set. 0: Full-speed device attached. 1: Low-speed device attached. ClearPortPower (write). The HCD clears the PortPowerStatus bit by writing a '1' to this bit. Writing a '0' has no effect.
8	PPS	PortPowerStatus (read). This bit reflects the port's power status, regardless of the type of power switching implemented. This bit is cleared if an overcurrent condition is detected. HCD sets this bit by writing **SetPortPower** or **SetGlobalPower**. HCD clears this bit by writing **ClearPortPower** or **ClearGlobalPower**. Which power control switches are enabled is determined by **PowerSwitchingMode** and **PortPortControlMask[NDP]**. In global switching mode (**PowerSwitchingMode**=0), only **Set/ClearGlobalPower** controls this bit. In per-port power switching (**PowerSwitchingMode**=1), if the **PortPowerControlMask[NDP]** bit for the port is set, only **Set/ClearPortPower** commands are enabled. If the mask is not set, only **Set/ClearGlobalPower** commands are enabled. When port power is disabled, **CurrentConnectStatus**, **PortEnableStatus**, **PortSuspendStatus**, and **PortResetStatus** should be reset. 0: Port power is off. 1: Port power is on. SetPortPower (write). The HCD writes a '1' to set the PortPowerStatus bit. Writing a '0' has no effect. **Note:** This bit always reads as '1' if power switching is not supported.
7:5	Reserved	Reserved.
4	PRS	PortResetStatus (read). Set by a write to **SetPortReset**, port reset signaling is asserted. When reset is completed, this bit is cleared when **PortResetStatusChange** is set. This bit cannot be set if **CurrentConnectStatus** is cleared. 0: Port reset signal is not active. 1: Port reset signal is active. SetPortReset (write). The HCD sets the port reset signaling by writing a '1' to this bit. Writing a '0' has no effect. If **CurrentConnectStatus** is cleared, this write does not set **PortResetStatus**, but instead sets **ConnectStatusChange**. This informs the driver that it attempted to reset a disconnected port.
3	POCI	PortOverCurrentIndicator. Only valid when the Root Hub is configured in such a way that overcurrent conditions are reported on a per-port basis. If per-port overcurrent reporting is not supported, this bit is set to 0. If cleared, all power operations are normal for this port. If set, an overcurrent condition exists on this port. This bit always reflects the overcurrent input signal. 0: No overcurrent condition. 1: Overcurrent condition detected. ClearSuspendStatus (write). The HCD writes a '1' to initiate a resume. Writing a '0' has no effect. A resume is initiated only if **PortSuspendStatus** is set.

Bit(s)	Name	Description
2	PSS	PortSuspendStatus (read). Indicates whether the port is suspended or in the resume sequence. It is set by a SetSuspendState write and cleared when PortSuspendStatusChange is set at the end of the resume interval. This bit cannot be set if CurrentConnectStatus is cleared. This bit is also cleared when PortResetStatusChange is set at the end of the port reset or when the HC is placed in the UsbResume state. If an upstream resume is in progress, it should propagate to the HC. 0: Port is not suspended. 1: Port is suspended. SetPortSuspend (write). The HCD sets the **PortSuspendStatus** bit by writing a '1' to this bit. Writing a '0' has no effect. If **CurrentConnectStatus** is cleared, this write does not set **PortSuspendStatus**; instead it sets **ConnectStatusChange**. This informs the driver that it attempted to suspend a disconnected port.
1	PES	PortEnableStatus (read). Indicates whether the port is enabled or disabled. The Root Hub may clear this bit when an overcurrent condition, disconnect event, switched-off power, or operational bus error such as babble is detected. This change also causes **PortEnabledStatusChange** to be set. HCD sets this bit by writing **SetPortEnable** and clears it by writing **ClearPortEnable**. This bit cannot be set when **CurrentConnectStatus** is cleared. This bit is also set, if not already, at the completion of a port reset when **ResetStatusChange** is set or port suspend when **SuspendStatusChange** is set. 0: Port is disabled. 1: Port is enabled. SetPortEnable (write). The HCD sets **PortEnableStatus** by writing a '1'. Writing a '0' has no effect. If **CurrentConnectStatus** is cleared, this write does not set **PortEnableStatus**, but instead sets **ConnectStatusChange**. This informs the driver that it attempted to enable a disconnected port.
0	CCS	CurrentConnectStatus (read). Reflects the current state of the downstream port. 0: No device is connected. 1: Device is connected. ClearPortEnable (write). The HCD writes a '1' to this bit to clear the **PortEnableStatus** bit. Writing a '0' has no effect. The **CurrentConnectStatus** is not affected by any write. **Note:** This bit always reads as '1' when the attached device is nonremovable (DeviceRemoveable[NDP]).

8 USB Slave

Register Descriptions

This section defines the different registers and their functionality.

Note that an indexing scheme is used for the Endpoint registers:

1. IN CSR1 and IN CSR2 (IN Control Status Registers)
2. OUT CSR1 and OUT CSR2 (OUT Control Status Registers)
3. IN MAXP (IN Maximum Packet Size Register)
4. OUT MAXP (OUT Maximum Packet Size Register)
5. Two OUT WRITE COUNT Registers

The Interrupt (Status) and Interrupt Enable registers are grouped in three banks:

- IN Endpoint Interrupts
- OUT Endpoint Interrupts
- USB Interrupts

The IN MAXP, IN INTERRUPT, and IN INTERRUPT ENABLE registers are used regardless of the direction of the endpoint. The associated CSR registers correspond to the direction of endpoint[1].

Below is the address map for the registers within the core, along with the abbreviations used in the code. Detailed descriptions of the registers are given in the following sections, along with details of Reset values.

Note: The given Reset values are selected both in response to a Power ON reset *and* to Reset Signaling on the USB, *except where stated otherwise*.

If the core does contain any OUT endpoints, the OUT CSR, OUT MAXP, OUT INTERRUPT and OUT INTERRUPT ENABLE registers do not exist. However, an OUT WRITE COUNT register exists to read the Write Count for Endpoint 0.

Address Map

The USBFuncFS register map is divided into two main sections:

1. Common USB registers. These registers provide control and status for the entire function controller.
2. Indexed registers. These registers provide control and status for one endpoint. There is an IN MAXP and two IN CSR registers for each endpoint that supports IN transactions and an OUT MAXP, two OUT CSR, and two OUT Count registers for each endpoint that supports OUT transactions (except for Endpoint 0 which has a reduced registered set: see below).

Only the registers for one IN endpoint and one OUT endpoint appear in the register map at any one time. The endpoints are selected by writing the endpoint number to the Index register. Therefore to access the registers

1. EPO is considered as an IN_OR_OUT endpoint and thus its MAXP is mapped to the IN_MAXP register.

for Endpoint 1, 1 must first be written to the Index register and then the control and status registers appear in the memory map. Table 8-1 provides a description of each register.

Table 8-1: USB Device Internal Register Summary

8051/USB Space	Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Name	Description
00	0x060000	R/W	8	0b00000000	Function Address	Maintains USB device address assigned by host.
01	0x060004	R/W	5	0b00000000	Power Management	Suspends, resumes, and resets signaling
02	0x060008	R	8	0b00000000	In Interrupt (EP0–EP7)	Mapping corresponds to endpoints whose direction is programmable (TYPE=IN_OR_OUT)
04	0x060010	R	7	0b00000000	Out Interrupt (EP0–EP7)	No mapping corresponds to endpoints whose direction is programmable (TYPE=IN_OR_OUT)
06	0x060018	R	3	0b000	USB Interrupt	Maintains interrupt status flags for bus signaling conditions
07	0x06001C	R	8	00000111	In Interrupt Enable (EP0–EP7)	Enables for endpoints 0-7
09	0x060024	R/W	8	00000111	Out Interrupt Enable (EP0–EP7)	Enables for endpoints 0-7
0B	0x06002C	R/W	3	0b100	USB Interrupt Enable	Enables for USB interrupts
0C	0x060030	R/W	8	0b00000000	Frame Number1	Maintains bits 7:0 of the current Frame number
0D	0x060034	R/W	3	0b000	Frame Number2	Maintains bits 10:8 of the current Frame number
0E	0x060038	R/W	8	0b00000000	Index	Maintains the number of the currently selected endpoint
10	0x060040	R/W	8	0b00000000	IN MAXP	Defines maximum packet size for IN endpoints
11	0x060044	R/W	12	0b00000000	IN CSR1	Maintains status bits for IN endpoints. **Note:** the EP0 CSR register is mapped to the IN CSR1 register.

Table 8-1: USB Device Internal Register Summary (Continued)

8051/USB Space	Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Name	Description
12	0x060048	R/W	4	0b00000000	IN CSR2	Configures IN endpoints
13	0x06004C	R/W	8	0b00000000	OUT MAXP	Defines maximum packet size for OUT endpoints
14	0x060050	R/W	8	0b00000000	OUT CSR1	Maintains status information at the endpoint
15	0x060054	R/W	2	0b0X000000	OUT CSR2	Configures the endpoint
16	0x060058	R/W	6	0b00000000	OUT Write Count1	Maintains bits 7:0 of the write count
17	0x06005C	R/W	12	0b00000000	OUT Write Count2	Maintains bits 15:8 of the write count

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. In the reset values, “X” indicates don’t care.

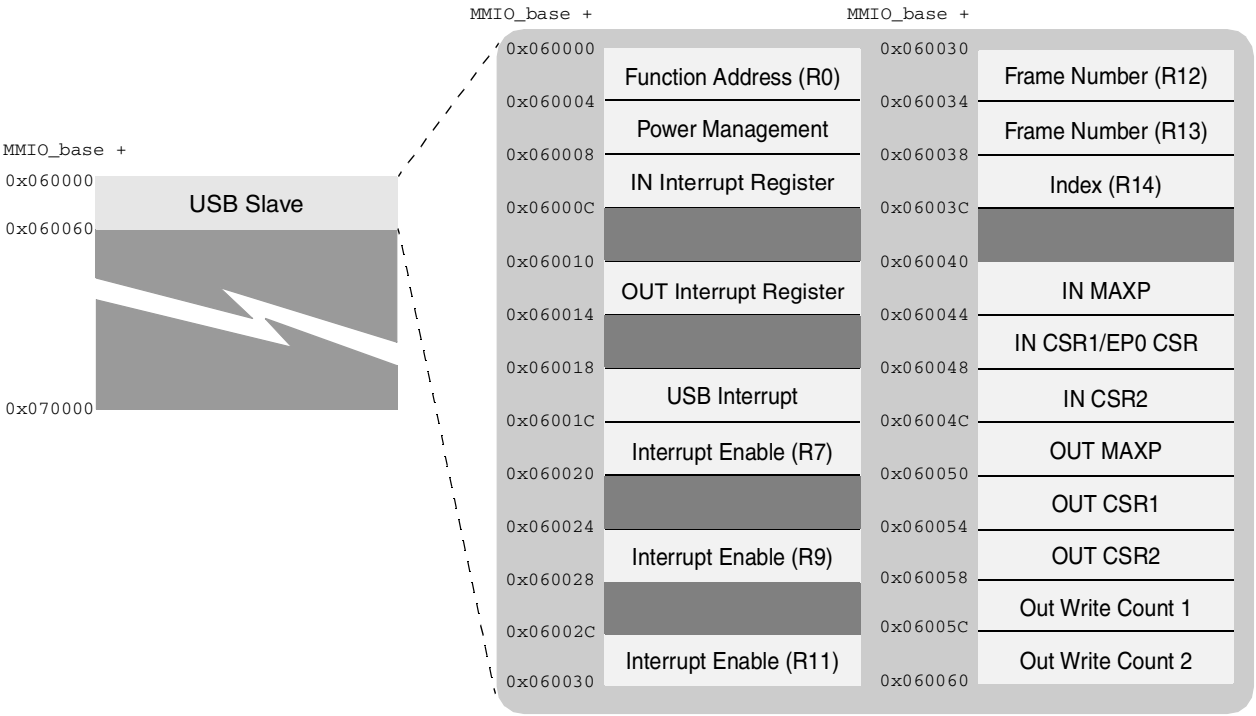
Table 8-2 shows the FIFO register address map.

Table 8-2: FIFO Register Address Map

8051/USB Space	Register Name	Byte Access Offset	Dword Access Offset
20	EP0 FIFO	0x060080	0x070000
21	EP1 FIFO	0x060084	0x071000
22	EP2 FIFO	0x060088	0x072000

Figure 8-1 lists the registers available in the USB Slave register space.

Figure 8-1: USB Slave Register Space



Address and Power Management Registers

Function Address (R0)

Read/Write MMIO_base + 0x060000

Power On Default: 0b00000000

This register maintains the USB Device Address assigned by the host.

The MCU writes to this register the value received through a SET_ADDRESS device request. This address will then be used for the next token following the successful completion of the SET_ADDRESS transfer.

7	6	5	4	3	2	1	0
ADDR Set R	FUNCTION RW R						

Bit(s)	Name	Description
7	ADDR	ADDR_UPDATE. The MCU sets this bit whenever it updates the FUNCTION_ADDR field in this register. The USB will clear this bit when the updated FUNCTION_ADDR field is used.
6:0	FUNCTION	FUNCTION_ADDR. The MCU writes the address to these bits. This address is used after the conclusion of the Status phase of the Control transfer, signaled by the clearing of the DATA_END bit in the Endpoint 0 CSR.

Power Management

Read/Write MMIO_base + 0x060004

Power On Default: 0b00000000

This register controls suspend, resume, and reset signaling.

7	6	5	4	3	2	1	0
ISO RW R	Reserved RW R			RESET R RW	UC RW R	SUSPEND R RW	ENABLE RW R

Bit(s)	Name	Description
7	ISO	ISO_UPDATE. Used only in ISO Mode. If set, the USB Function Controller will wait for a SOF token from the time IN_PKT_RDY was set before sending the packet. If an IN token is received before a SOF token, then a zero length data packet will be sent.
6:4	Reserved	These bits are reserved.

Bit(s)	Name	Description
3	RESET	RESET_MODE. The USB Function Controller sets this bit 2.6µs after reset signaling is received from the host (both USB inputs low). The USB_RSTN signal then goes low for 250ns. The RESET_MODE bit remains set as long as reset signaling persists on the bus.
2	UC	UC_RESUME. The MCU sets this bit for a duration of 10ms (maximum of 15ms) to initiate resume signaling. The USB Function Controller, while still in Suspend mode, generates resume signaling while this bit is set.
1	SUSPEND	SUSPEND_MODE. This bit is set by the USB Function Controller when it enters Suspend mode. It is cleared under the following conditions: • The MCU clears the UC_RESUME bit, to end resume signaling. • The MCU reads the USB Interrupt register for the USB_RESUME interrupt.
0	ENABLE	ENABLE_SUSPEND. If this bit is a zero, the USB Function Controller will not enter Suspend mode. 0: Disable Suspend mode (default). 1: Enable Suspend mode.

Note: The MCU should use the USB Interrupt Register to poll for suspend and reset conditions.

Interrupt Registers

The USB core has three groups of interrupt registers:

- IN Interrupt Registers
- OUT Interrupt Registers
- USB Interrupt Register

These registers act as status registers for the MCU when it is interrupted. The bits in these registers are cleared by the MCU by writing a 1 to each bit that was set.

Each register and its bits are explained below.

Note: Once the MCU is interrupted, it needs to read all the interrupt registers and write back the read data to all registers to clear the interrupt. The write back operation must be performed irrespective of the read value. The USB Interrupt register must be the last register.

IN Interrupt Register

Read/Write MMIO_base + 0x060008

Power On Default: 0b00000000

Each bit in this register corresponds to the respective Endpoint number.

All interrupts corresponding to endpoints whose direction is programmable (TYPE = IN_OR_OUT) are mapped to this register. The corresponding bits in the OUT INTERRUPT register will be reserved.

Note: Left unchanged by Reset Signaling.

7	6	5	4	3	2	1	0
EP1 - EP7 IN or IN_OR_OUT INTERRUPTS[1] R Set							EP0 R Set

1. If an endpoint is OUT only (not IN, IN_OR_OUT, or IN_AND_OUT type), the corresponding bits are reserved.

Bit(s)	Name	Description
7:1	IN or IN_OR_OUT INTERRUPTS	*For BULK/Interrupt endpoints:* The USB sets the corresponding bit under the following conditions: 1. IN_PKT_RDY bit (IN CSR1.D0) is cleared 2. SENT_STALL bit (IN CSR1.D5) is set 3. FIFO is flushed *For ISO endpoints:* The USB sets the corresponding bit under the following conditions: 1. IN_PKT_RDY bit (IN CSR1.D0) is cleared. 2. UNDER_RUN bit (IN CSR1.D2) is set. 3. SENT_STALL bit (IN CSR1.D5) is set. 4. FIFO is flushed. **Note:** Conditions 1 and 2 are mutually exclusive.
0	EP0	Endpoint 0 Interrupt. The USB sets this bit under the following conditions, as recorded in Endpoint 0 CSR: 1. OUT_PKT_RDY bit (D0) is set. 2. IN_PKT_RDY bit (D1) is cleared. 3. SENT_STALL bit (D2) is set. 4. DATA_END bit (D3) is cleared (Indicates End of control transfer). 5. SETUP_END bit (D4) is set.

OUT Interrupt Register

Read/Write MMIO_base + 0x060010

Power On Default: 0b00000000

Each bit in this register corresponds to the respective Endpoint number.

All interrupts corresponding to endpoints whose direction is programmable (TYPE = IN_OR_OUT) are not mapped to this register. These bits will be RESERVED.

Note: Left unchanged by Reset Signaling.

7	6	5	4	3	2	1	0
EP1 - EP7 OUT INTERRUPTS[1] R Set							Reserved

1. If an endpoint is direction IN or programmable (TYPE=IN or TYPE=IN_OR_OUT), the corresponding bits are reserved.

Bit(s)	Name	Description
7:1	EP1 - EP7 OUT INTERRUPTS	*For BULK/Interrupt endpoints:* The USB sets the corresponding bit under the following conditions: 1. OUT_PKT_RDY bit (OUT CSR1.D0) is cleared 2. SENT_STALL bit (OUT CSR1.D6) is set *For ISO endpoints:* The USB sets the corresponding bit under the following conditions: 1. OUT_PKT_RDY bit (OUT CSR1.D0) is cleared. 2. OVER_RUN bit (OUT CSR1.D2) is set. 3. SENT_STALL bit (OUT CSR1.D6) is set. **Note:** Conditions 1 and 2 are mutually exclusive.
0	Reserved	This bit is reserved.

USB Interrupt

Read/Write MMIO_base + 0x060018

Power On Default: 0b000

This register maintains interrupt status flags for bus signaling conditions.

Note: Left unchanged by Reset Signaling.

7	6	5	4	3	2	1	0
Reserved					USB RESET R Set	RESUME R Set	SUSPEND R Set

Bit(s)	Name	Description
7:3	Reserved	These bits are reserved.
2	USB RESET	USB Reset Interrupt. The USB sets this bit when it receives reset signaling.
1	RESUME	Resume Interrupt. The USB sets this bit, when it receives resume signaling, *while in Suspend mode*. If the resume is due to a USB reset, the MCU is first interrupted with a RESUME interrupt. Once the clocks resume and the SE0 condition persists for 3ms, USB RESET interrupt will be asserted.
0	SUSPEND	Suspend Interrupt. The USB sets this bit when it receives Suspend signaling. This bit is set whenever there is no activity for 3ms on the bus. Thus if the MCU does not stop the clock after the first suspend interrupt, it will continue to be interrupted every 3ms as long as there is no activity on the USB bus. By default this interrupt is disabled.

Interrupt Enable Registers (R7, R9, R11)

For each interrupt register, there is a corresponding Interrupt Enable register. Register R7 masks the interrupts in the IN Interrupt Register R2; register R9 masks the interrupts in the OUT Interrupt Register R4; and register R11 masks the interrupts in the USB Interrupt Register R6 (except that the Resume Interrupt cannot be masked, i.e. USB Interrupt Enable Register D1 is ignored).

In each of these registers, if bit = 0, the interrupt is disabled. If bit = 1, the interrupt is enabled.

Please note that in each case it is the connection between the interrupt bit becoming set and MC_INTR going low that is enabled or disabled: the interrupt registers themselves record when each interrupt is set even when the interrupt is disabled.

After reset, the IN and OUT Interrupt Enable Registers (registers R7 and R9) are set to enable the interrupts corresponding to the configured range of Endpoints, while the USB Interrupt Enable Register (R11) is set to just enable USB Reset Interrupts.

Interrupt Enable (R7)

Read/Write MMIO_base + 0x06001C

Power On Default: N/A

7	6	5	4	3	2	1	0
IN INTERRUPT ENABLE R/W R							

Bit(s)	Name	Description
7:0	IN INTERRUPT ENABLE	Enables for Endpoints 0-7.

Interrupt Enable (R9)

Read/Write MMIO_base + 0x060024

Power On Default: N/A

7	6	5	4	3	2	1	0
OUT INTERRUPT ENABLE R/W R							

Bit(s)	Name	Description
7:0	OUT INTERRUPT ENABLE	Enables for Endpoints 0-7.

Interrupt Enable (R11)

Read/Write MMIO_base + 0x06002C

Power On Default: 0b100

Note: Reset signaling leaves D2 unchanged.

7	6	5	4	3	2	1	0
Reserved					USB INTERRUPT ENABLE R/W R		

Bit(s)	Name	Description
7:3	Reserved	These bits are reserved.
2:0	USB INTERRUPT ENABLE	The D1 setting is ignored because the Resume interrupt cannot be masked.

Frame Number Registers

There are two Frame Number registers, R12 and R13. These two registers record the frame number received through the SOF taken. This information can be read by the MCU when it detects a SOF_PULSE from the core.

If the SOF token received is corrupted, a synthetic SOF_PULSE is generated (pulse width = one 12MHz clock) but the Frame number register remains unchanged.

Firmware can use these registers to determine the start and end points of an Isochronous transfer.

Frame Number (R12)

Read/Write MMIO_base + 0x060030

Power On Default: 0b00000000

7	6	5	4	3	2	1	0
FRAME NUMBER 1 R W							

Bit(s)	Name	Description
7:0	FRAME NUMBER 1	Maintains bits 7:0 of the current Frame number.

Frame Number (R13)

Read/Write MMIO_base + 0x060034

Power On Default: 0b000

7	6	5	4	3	2	1	0
Reserved					FRAME NUMBER 2 R W		

Bit(s)	Name	Description
7:3	Reserved	These bits are reserved.
2:0	FRAME NUMBER 2	Maintains bits 10:8 of the current Frame number.

Index Register

Index (R14)

Read/Write MMIO_base + 0x060038

Power On Default: 0b00000000

The Index register maintains the number of the Endpoint that is currently selected, i.e. the Endpoint that the Indexed Registers IN-R1 through IN-R3, OUT-R1 through OUT-R5 currently display information about.

7	6	5	4	3	2	1	0
INDEX REGISTER R/W R							

Bit(s)	Name	Description
7:0	INDEX REGISTER	Maintains the number of the currently selected endpoint.

IN MAXP Register

IN MAXP

Read/Write MMIO_base + 0x060040

Power On Default: 0b00000001 (MAXP=8 bytes)

This register defines the maximum packet size for IN endpoints[1].

The packet size may be set in multiples of 8 bytes up to 1023 bytes. If the MCU writes a value greater than the FIFO size, the value recorded will be automatically changed to select the FIFO size.

7	6	5	4	3	2	1	0
MAXP R/W R							

Bit(s)	Name	Description
7:0	MAXP	0x00 MAXP = 0 0x01 MAXP = 8 0x02 MAXP = 16 0x03 MAXP = 24 ... ... 0x80 MAXP = 1023

1. For IN_OR_OUT type endpoints, MAXP is mapped to this register irrespective of the direction of the endpoint. EP0 is considered to be IN_OR_OUT type, so its MAXP is mapped to IN MAXP register

IN CSR Registers

These registers maintain the control and status bits for IN endpoints. They are split into IN CSR1 and IN CSR2. IN CSR1 maintains the status bits, while IN CSR2 maintains the configuration bits.

IN CSR1

Read/Write MMIO_base + 0x060044

Power On Default: 0b00000000

This register maintains the status bits for IN endpoints. The MCU only needs to access this register for an IN endpoint, once the endpoint has been configured.

7	6	5	4	3	2	1	0
Reserved R R	CLR W R	SENT R Set	SEND R/W R	FIFO RW Clear	UNDER R Set	FIFO_NOT R Set	IN R Clear

Bit(s)	Name	Description
7	Reserved	This bit is reserved.
6	CLR	CLR_DATA_TOGGLE. When the MCU writes a 1 to this bit, the data toggle sequence bit is reset to DATA0.
5	SENT	SENT_STALL. The USB sets this bit when a STALL handshake is issued to an IN token, due to the MCU setting the SEND_STALL bit (D4). When the USB issues a STALL handshake, IN PKT RDY is cleared.
4	SEND	SEND_STALL. The MCU writes a 1 to this register to issue a STALL handshake to the USB. The MCU clears this bit to end the STALL condition.
3	FIFO	FIFO_FLUSH. The MCU sets this bit when it intends to flush the IN FIFO. This bit is cleared by the USB when the FIFO is flushed. The MCU is interrupted when this happens. If a token is in progress, the USB waits until the transmission is complete before flushing the FIFO. If two packets are loaded into the FIFO, only the top-most packet (one that was intended to be sent to the host) is flushed, and the corresponding IN_PKT_RDY bit for that packet is cleared.
2	UNDER	UNDER_RUN. *Valid For Iso Mode Only* The USB sets this bit in ISO mode when an IN token is received and the IN_PKT_RDY bit is not set. Under such conditions, the USB sends a zero length data packet and the next packet that is loaded into the IN FIFO is flushed.
1	FIFO_NOT	FIFO_NOT_EMPTY. When set, this bit indicates that there is at-least one packet of data in the FIFO.
0	IN	IN_PKT_RDY. The MCU sets this bit after writing a packet of data into the FIFO. The USB clears this bit once the packet has been successfully sent to the host. The USB also clears this bit if the FIFO is double-buffered and there is room for a second packet. An interrupt is generated when the USB clears this bit to prompt the MCU to load the next packet. While this bit is set, the MCU will not be able to write to the FIFO. This bit cannot be set if the SEND_STALL bit (D5) has been set by the MCU.

Note: D0 and D1 together can be used to indicate the FIFO state—in particular, whether it is Write Ready or Write Busy—as follows:

D0=0; D1=0; ⇒	FIFO empty	Write Ready
D0=0; D1=1; ⇒	1 packet ≤ ½ FIFO size	Write Ready
D0=1; D1=0;	*Impossible combination*	
D0=1; D1=1; ⇒	2 Packets ≤ ½ FIFO size or 1 Packet > ½ FIFO size	Write Busy

IN CSR2

Read/Write MMIO_base + 0x060048

Power On Default: 0b0XX00000

This register is used to configure IN endpoints.

7	6	5	4	3	2	1	0
AUTO_SET R/W R	ISO R/W R	MODE_IN R/W R	RATE R/W R	Reserved			

Bit(s)	Name	Description
7	AUTO_SET	If set, IN_PKT_RDY automatically are set by the core whenever the MCU writes MAXP data, without any intervention from MCU. If the MCU writes less than MAXP data, then IN_PKT_RDY bit has to be set by the MCU. The default for this bit is 0.
6	ISO	This bit is only R/W for endpoints whose transfer type is programmable, i.e., TYPE = ISO_OR_BULK. For other endpoints, this bit is read-only and returns the appropriate value. 0: Configures endpoint for Bulk transfers (default). 1: Configures endpoint for ISO transfers.
5	MODE_IN	This bit is only used for endpoints whose direction is programmable, i.e., TYPE = IN_OR_OUT. For other endpoints, this bit is Read only and returns zero when read. 1: Configures endpoint direction as IN. 0: Configures endpoint direction as OUT. The default for this bit is IN (D5 = 1).
4	RATE_INTERRUPT	This bit is used only for BULK/Interrupt type endpoints. Isochronous endpoints automatically toggle the Data PID regardless of the state of this bit. When set, this bit configures the endpoint as a rate feedback interrupt endpoint i.e. the PID for each packet will toggle between DATA0 and DATA1 regardless of whether the previous transfer was acknowledged. 0: Configures endpoint direction as IN. 1: Configures endpoint direction as OUT.
3:0	Reserved	These bits are reserved.

OUT MAXP Register

OUT MAXP

Read/Write MMIO_base + 0x06004C

Power On Default: 0b00000001 (MAXP=8 bytes)

This register defines the maximum packet size for OUT endpoints.

The packet size may be set in multiples of 8 bytes up to 1023 bytes. If the MCU writes a value greater than the FIFO size, the value recorded will be automatically changed to select the FIFO size.

The packet size may be set in multiples of 8 bytes up to 1023 bytes. If the MCU writes a value greater than the FIFO size, the value recorded will be automatically changed to select the FIFO size.

7	6	5	4	3	2	1	0
MAXP R/W R							

Bit(s)	Name	Description
7:0	MAXP	0x00 MAXP = 0 0x01 MAXP = 8 0x02 MAXP = 16 0x03 MAXP = 24 ... ... 0x80 MAXP = 1023

OUT CSR Registers

There are two CSR registers, OUT CSR1 and OUT CSR2, which are used to control OUT endpoints by the MCU. OUT CSR1 maintains status information, while OUT CSR2 is used to configure the endpoint.

OUT CSR1

Read/Write MMIO_base + 0x060050

Power On Default: 0b00000000

The OUT CSR1 register maintains status information.

7	6	5	4	3	2	1	0
CLR W R	SENT R Set	SEND R/W R	FIFO W Clear	DATA R R/W	OVER R Set	FIFO_FULL R Set	OUT R Set

Bit(s)	Name	Description
7	CLR	CLR_DATA_TOGGLE. When the MCU writes a 1 to this bit, the data toggle sequence bit is reset to DATA0.
6	SENT	SENT_STALL. The USB sets this bit when an OUT token is ended with a STALL handshake. The USB issues a STALL handshake to the host if it sends more than MAXP data for the OUT token.
5	SEND	SEND_STALL. The MCU writes a 1 to this bit to issue a STALL handshake to the USB. The MCU clears this bit to end the STALL condition.
4	FIFO	FIFO_FLUSH. The MCU writes a 1 to this bit to flush the FIFO. This bit can be set only when OUT_PKT_RDY (D0) is set. The packet due to be unloaded by the MCU will be flushed.
3	DATA	DATA_ERROR. This bit should be sampled alongside OUT_PKT_RDY (D0). When set, it indicates the data packet due to be unloaded by the MCU has an error (either bit stuffing or CRC). If two packets are loaded into the FIFO, and the second packet has an error, then this bit gets set only after the first packet is unloaded. This bit is automatically cleared when OUT_PKT_RDY gets cleared.
2	OVER	OVER_RUN. *Valid For Iso Mode Only* This bit is set if the core is not able to load an OUT ISO token into the FIFO.
1	FIFO_FULL	Indicates no more packets can be accepted. D0=0; D1=0; ⇒ : FIFO empty D0=1; D1=0; : 1 packet ≤ ½ FIFO size in FIFO D0=1; D1=1; ⇒ : 2 Packets ≤ ½ FIFO size or 1 Packet > ½ FIFO size
0	OUT	OUT_PKT_RDY. The USB sets this bit once it has loaded a packet of data into the FIFO. Once the MCU reads the FIFO for the entire packet, this bit should be cleared by the MCU[1].

1. See AUTO_CLR.

OUT CSR2

Read/Write MMIO_base + 0x060054

Power On Default: 0b0XX00000

This register is used to configure OUT endpoints.

7	6	5	4	3	2	1	0
AUTO_CLR R/W R	ISO R/W R	Reserved					

Bit(s)	Name	Description
7	AUTO_CLR	If set, OUT_PKT_RDY automatically is cleared by the core whenever the MCU reads data from the OUT FIFO, without any intervention from MCU. If the MCU writes less than MAXP data, then IN_PKT_RDY bit has to be set by the MCU. The default for this bit is 0.
6	ISO	This bit is only R/W for endpoints whose transfer type is programmable, i.e., TYPE = ISO_OR_BULK. For other endpoints, this bit is Read only and returns the appropriate value. 0: Configures endpoint for Bulk transfers (default). 1: Configures endpoint for ISO transfers.
5:0	Reserved	These bits are reserved.

Endpoint 0 CSR Register

EP0 CSR

Read/Write MMIO_base + 0x060044

Power On Default: 0b00000000

This register contains the control and status bits for Endpoint 0. Because a control transaction involves both IN and OUT tokens, there is only one CSR register, mapped to the IN CSR1 register.

7	6	5	4	3	2	1	0
SERV_SET W Clear	SERV_OUT W Clear	SEND R/W Clear	SETUP_END R Set	DATA_END R Clear	SENT R Set	IN R Clear	OUT R Set

Bit(s)	Name	Description
7	SERV_SET	SERVICED_SETUP_END. The MCU writes a 1 to this bit to clear SETUP_END (D4).
6	SERC_OUT	SERVICED_OUT_PKT_RDY. The MCU writes a 1 to this bit to clear OUT_PKT_RDY (D0).
5	SEND	SEND_STALL. When the MCU decodes a invalid token, it writes a 1 to this bit at the same time as it clears OUT_PKT_RDY (D0). The USB then issues a STALL handshake to the current control transfer. The MCU writes a 0 to end the STALL condition.
4	SETUP_END	The USB sets this read-only bit when a control transfer ends before DATA_END (D3) is set. The MCU clears this bit by writing a 1 to the SERVICED_SETUP_END (D7) bit. When the USB sets this bit, an interrupt is generated to the MCU. The USB then flushes the FIFO and invalidates MCU access to the FIFO. When MCU access to the FIFO is invalidated, this bit is cleared.[1]
3	DATA_END	The MCU sets this bit: 1. After the last packet of data is loaded into the FIFO, IN_PKT_RDY is set. 2. While it clears OUT_PKT_RDY after unloading the last packet of data. 3. For a zero length data phase, when it clears OUT_PKT_RDY and sets IN_PKT_RDY[2]
2	SENT	SENT_STALL. The USB sets this bit if a control transaction is ended following a protocol violation. An interrupt is generated when this bit is set.
1	IN	IN_PKT_RDY. The MCU sets this bit after writing a packet of data into Endpoint 0 FIFO. The USB clears this bit once the packet has been successfully sent to the host. An interrupt is generated when the USB clears this bit, so the MCU can load the next packet. For a zero length data phase, the MCU sets IN_PKT_RDY and DATA_END (D3) at the same time.
0	OUT	OUT_PKT_RDY. The USB sets this read-only bit once a valid token is written to the FIFO. An interrupt is generated when the USB sets this bit. The MCU clears this bit by writing a 1 to the SERVICED_OUT_PKT_RDY bit (D6).

1. The OUT_PKT_RDY bit may be set at the same time that the SETUP_END bit is set. This happens when the current transfer has ended, and a new control transfer is received before the MCU can service the interrupt. In such a case, the MCU should first clear the SETUP_END bit, and then start servicing the new control transfer.
2. In the case of a control transfer with no data phase, then after unloading the setup token the MCU sets IN_PKT_RDY and DATA_END at the same time that it clears OUT_PKT_RDY for the Setup token.

OUT Write Counter Registers

The OUT Write Count registers are used to record the write count (that is, the number of bytes in the packet that is due to be unloaded by the MCU).

Out Write Count 1

Read/Write MMIO_base + 0x060058

Power On Default: 0b00000000

The OUT_WRT_CNT1 register maintains the lower byte of this write count (bits 7:0).

7	6	5	4	3	2	1	0
OUT_WRT_CNT1 R W							

Bit(s)	Name	Description
7:0	OUT_WRT_CNT1	Maintains bits 7:0 of the Write Count.

Out Write Count 2

Read/Write MMIO_base + 0x06005C

Power On Default: 0b00000000

The OUT_WRT_CNT2 register maintains the higher byte of the write count (bits 15:8).

7	6	5	4	3	2	1	0
OUT_WRT_CNT2 R W							

Bit(s)	Name	Description
7:0	OUT_WRT_CNT2	Maintains bits 15:8 of the Write Count.

9 GPIO/I²C

Functional Overview

GPIO Interface

The GPIO peripheral includes the following registers:

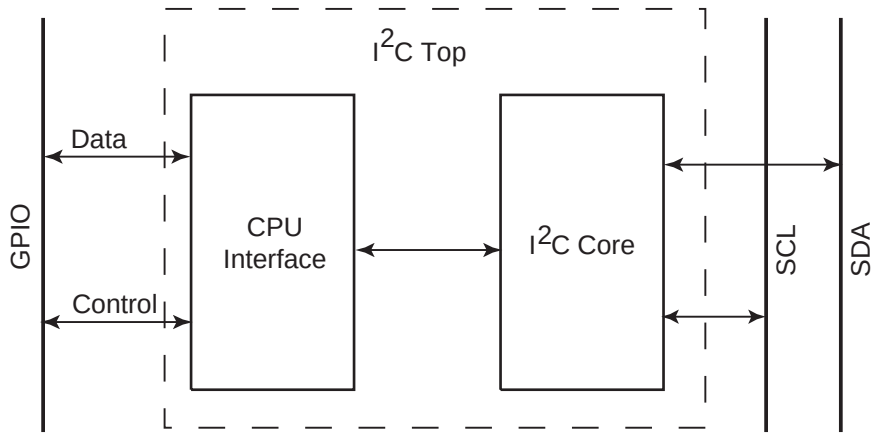
- Data register
- Data Direction register
- Interrupt Setup register
- Interrupt Status register
- Interrupt Reset register

I²C Interface

The SM501 supports one I2C interface (GPIO bits 46:47). The interface is in Master mode with 7-bit addressing. It supports speeds up to 400 kb/s (Fast mode).

Figure 9-1 shows a simplified block diagram of the I²C interface.

Figure 9-1: I²C Interface Block Diagram



Programmer's Model

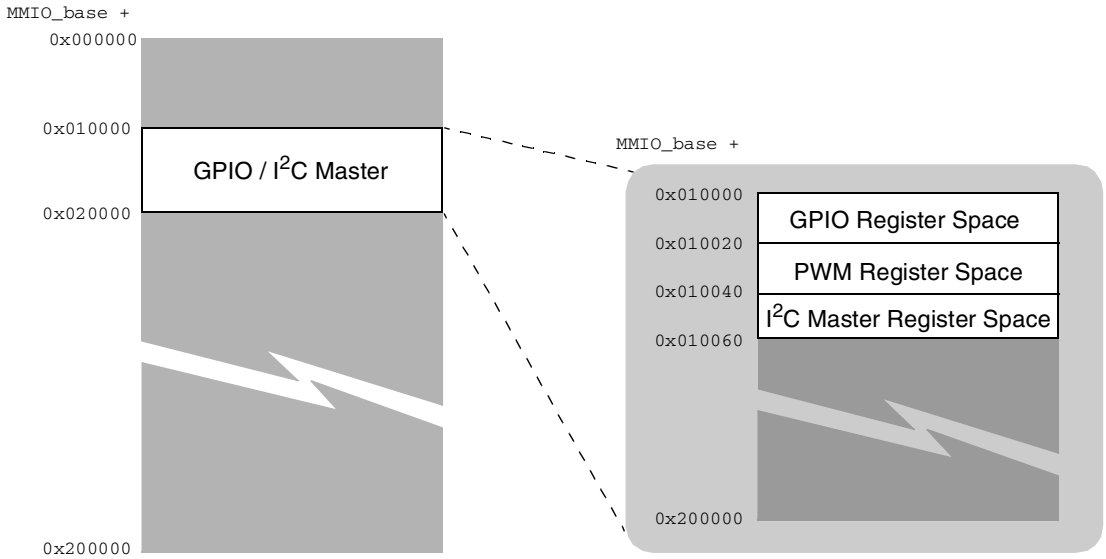
The base address of the GPIO is not fixed, and can be different for any particular system implementation. However, the offset of any particular register from the base address is fixed.

The following locations are reserved and must not be used during normal operation:

- Locations at offsets 0x424 to 0xFCC are reserved for possible future extensions and test purposes
- Locations at offsets +0xFDO to +0xFDC are reserved for future ID expansion.

Figure 9-2 shows how this 64kB region in the MMIO space is laid out. It controls the GPIO registers and the I^2C Master registers.

Figure 9-2: GPIO / I^2C Master Register Space



The following sections define each region in more detail.

Register Descriptions

The GPIO and I^2C registers are shown in Table 9-1.

Table 9-1: GPIO and I^2C Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Register Name
0x010000	R/W	32	0x00000000	GPIO Data Low
0x010004	R/W	32	0x00000000	GPIO Data High
0x010008	R/W	32	0x00000000	GPIO Data Direction Low
0x01000C	R/W	32	0x00000000	GPIO Data Direction High
0x010010	R/W	32	0x00000000	GPIO Interrupt Setup
0x010014	R	32	0x00000000	GPIO Interrupt Status
0x010014	W	32	0x00000000	GPIO Interrupt Reset
I^2C Master				
0x010040	R/W	8	0x00	I^2C Byte Count
0x010041	R/W	8	0x00	I^2C Control
0x010042	R	8	0x00	I^2C Status
0x010042	W	8	0x00	I^2C Reset
0x010043	R/W	8	0x00	I^2C Slave Address
0x010044 - 0x010053	R/W	8	0x00	I^2C Data

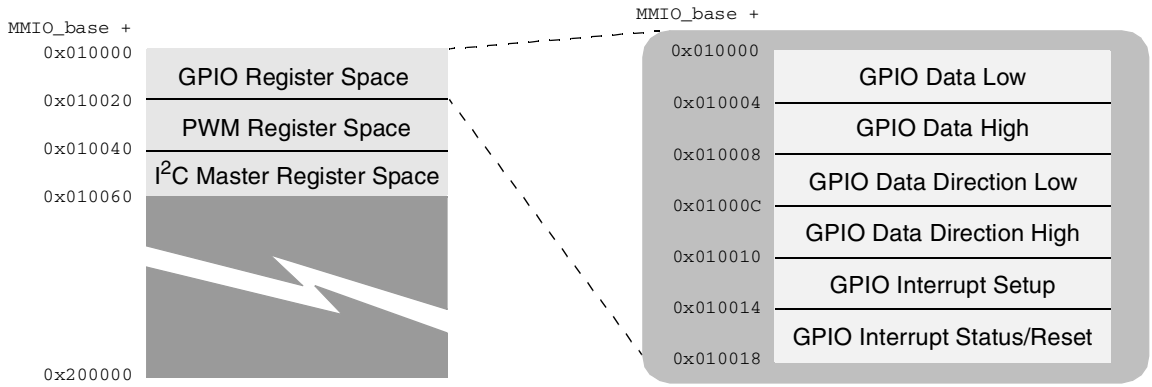
1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.

GPIO Register Descriptions

The GPIO registers are described in this section.

The GPIO registers control the GPIO pins. There are seven GPIO registers, two of which share the same address for interrupt status/reset. Figure 9-3 defines the register layout for the GPIO registers.

Figure 9-3: GPIO Register Space



GPIO Data Low

Read/Write Address 0x010000

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Data_{31:16}$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Data_{15:0}$ R/W															

Bit(s)	Name	Description
31:0	$Data_{31:0}$	The values in the $Data_{31:0}$ bits reflect the value on the $GPIO_{31:0}$ pins.

This register reflects the value on a GPIO pin. If it is programmed as an input, the value of the GPIO pin is transferred to the corresponding bit in this register. If it is programmed as an output, the value of the bit is transferred to the corresponding GPIO pin.

GPIO Data High

Read/Write Address 0x010004

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Data_{63:48}$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Data_{47:32}$ R/W															

Bit(s)	Name	Description
31:0	$Data_{63:32}$	The values in the $Data_{63:32}$ bits reflect the value on the $GPIO_{63:32}$ pins.

This register reflects the value on a GPIO pin. If it is programmed as an input, the value of the GPIO pin is transferred to the corresponding bit in this register. If it is programmed as an output, the value of the bit is transferred to the corresponding GPIO pin.

GPIO Data Direction Low

Read/Write Address 0x010008

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Direction_{31:16}$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Direction_{15:0}$ R/W															

Bit(s)	Name	Description
31:0	$Direction_{31:0}$	This register defines whether a GPIO pin is programmed as in input or as an output. 0: Input. 1: Output.

GPIO Data Direction High

Read/Write Address 0x01000C

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
$Direction_{63:48}$ R/W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
$Direction_{47:32}$ R/W															

Bit(s)	Name	Description
31:0	$Direction_{63:32}$	This register defines whether a GPIO pin is programmed as in input or as an output. 0: Input. 1: Output.

GPIO Interrupt Setup

Read/Write Address 0x010010

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									$Trigger_{54:48}$ R/W						
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res	$Active_{54:48}$ R/W							Res	$Enable_{54:48}$ R/W						

Bit(s)	Name	Description
31:23	Reserved	These bits are reserved.
22:16	$Trigger_{54:48}$	Triggering Type. 0: Edge triggered. 1: Level triggered.
15	Res	This bit is reserved.
14:8	$Active_{54:48}$	Active State. 0: Active low or falling edge. 1: Active high or rising edge.
7	Res	This bit is reserved.
6:0	$Enable_{54:48}$	This register defines whether GPIO54:48 pins are programmed as regular input/output pins or as interrupt input pins. It also defines the interrupt type. 0: Regular GPIO Input/Output. 1: GPIO Interrupt.

GPIO Interrupt Status

Read Address 0x010014

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									$Status_{54:48}$ R/W						
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved															

Bit(s)	Name	Description
31:23	Reserved	These bits are reserved.
22:16	$Status_{54:48}$	This read/only register reflects the status of the interrupt pins. When an external interrupt happens on a GPIO interrupt pin, the status bit will be set to “1” until the software resets the interrupt by writing to the GPIO Interrupt Status. 0: Interrupt inactive. 1: Interrupt active.
15:0	Reserved	These bits are reserved.

GPIO Interrupt Reset

Write Address 0x010014

Power-on Default 0x00000000

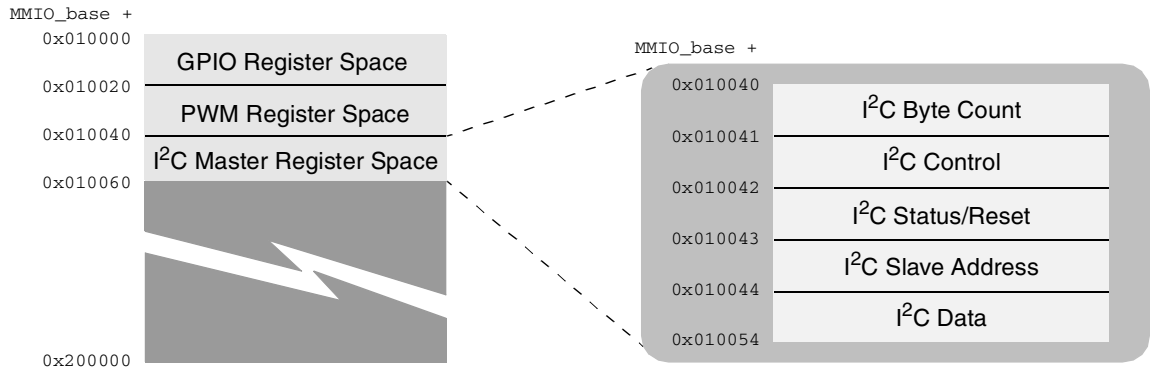
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									$Reset_{54:48}$ W						
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved															

Bit(s)	Name	Description
31:23	Reserved	These bits are reserved.
22:16	$Reset_{54:48}$	This field resets the GPIO interrupt. 0: No action. 1: Reset interrupt.
15:0	Reserved	These bits are reserved.

I²C Master Register Descriptions

The I²C Master registers control the I²C GPIO pins. Figure 9-4 defines the register layout for the I²C Master registers.

Figure 9-4: I²C Master Register Space



I²C Byte Count

Read/Write Address 0x010040

Power-on Default 0x00

7	6	5	4	3	2	1	0
Reserved				Count R/W			

Bit(s)	Name	Description
7:4	Reserved	These bits are reserved.
3:0	Count	Byte count – 1.

I²C Control

Read/Write Address 0x010041

Power-on Default 0x00

7	6	5	4	3	2	1	0
Reserved R/W	Repeat R/W	IntAck R/W	Int R/W	Reserved R/W	Status R/W	Mode R/W	E R/W

Bit(s)	Name	Description
7	Reserved	This bit is reserved.
6	Repeat	Repeated Start Enable. 0: Disable. 1: Enable.
5	IntAck	Interrupt Acknowledge. 0: Not acknowledge. 1: Acknowledge.
4	Int	Interrupt Enable. 0: Disable. 1: Enable.
3	Reserved	This bit is reserved.
2	Status	Start/Stop Status. 0: Stop. 1: Start.
1	Mode	Bus Speed Mode Select. 0: Standard mode (100kbps). 1: Fast mode (400kbps).
0	E	Controller Enable. 0: Disable. 1: Enable.

I²C Status

Read Address 0x010042

Power-on Default 0x00

7	6	5	4	3	2	1	0
Reserved				Complete R	Error R	Ack R	Busy R

Bit(s)	Name	Description
7:4	Reserved	These bits are reserved.
3	Complete	Whole Transfer Completed. 0: Transfer in progress. 1: Transfer completed.
2	Error	Bus Error. 0: Normal. 1: Error.
1	Ack	Slave Acknowledge Received. 0: Not received. 1: Received.
0	Busy	Bus Busy. 0: Idle. 1: Busy.

I²C Reset

Write Address 0x010042

Power-on Default 0x00

7	6	5	4	3	2	1	0
Reserved					Error W	Reserved	

Bit(s)	Name	Description
7:3	Reserved	These bits are reserved.
2	Error	Bus Error. 0: Clear.
1:0	Reserved	These bits are reserved.

I²C Slave Address

Read/Write Address 0x010043

Power-on Default 0x00

7	6	5	4	3	2	1	0
Address R/W							R/W R/W

Bit(s)	Name	Description
7:1	Address	7-bit Slave Address.
0	R/W	Read/Write Select. 0: Write. 1: Read.

I²C Data

Read/Write Address 0x010044 through 0x010053

Power-on Default 0x00

7	6	5	4	3	2	1	0
Data R/W							

Bit(s)	Name	Description
7:0	Data	There are 16 I²C Data registers that hold the data to be written to or read from the I²C slave. These registers can be accessed in 8-bit, 16-bit, or 32-bit mode for very fast FIFO transfers.

10 ZV Port

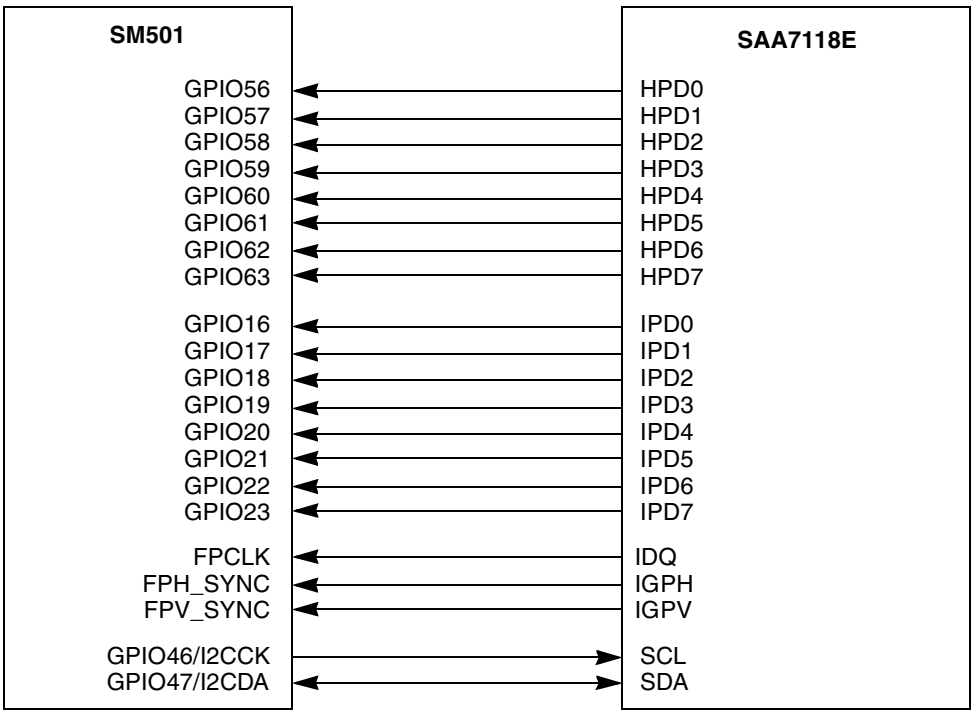
Functional Overview

This section covers the ZV Port and the Video Capture Unit.

ZV Port Overview

The SM501 Zoom Video Port (ZV Port) can interface with video decoders, such as NTSC/PAL decoders, MPEG-2 decoders, and JPEG codecs. The ZV Port also can interface directly to an NTSC/PAL decoder, such as the Philips SA7111 or BT819. Figure 10-1 illustrates an example of the interface between the Philips SA7118 TV decoder and the ZV Port.

Figure 10-1: TV Decoder Interface through the ZV Port



Incoming video data from the ZV Port can be interlaced or non-interlaced and YUV or RGB format. By disabling the video capture function, the ZV Port can be configured in output mode. In output mode, the ZV Port can send video data and 18-bit graphics in RGB format.

Video Capture Unit Overview

The Video Capture Unit captures incoming video data from the ZV Port and then stores the data into the frame buffer. The Video Capture Unit maintains display quality and balances the capture rate. Its key features are:

- 2-to-1 reduction for horizontal and vertical frame size
- YUV 4:2:2, YUV 4:2:2 with byte swapping, and RGB 5:6:5

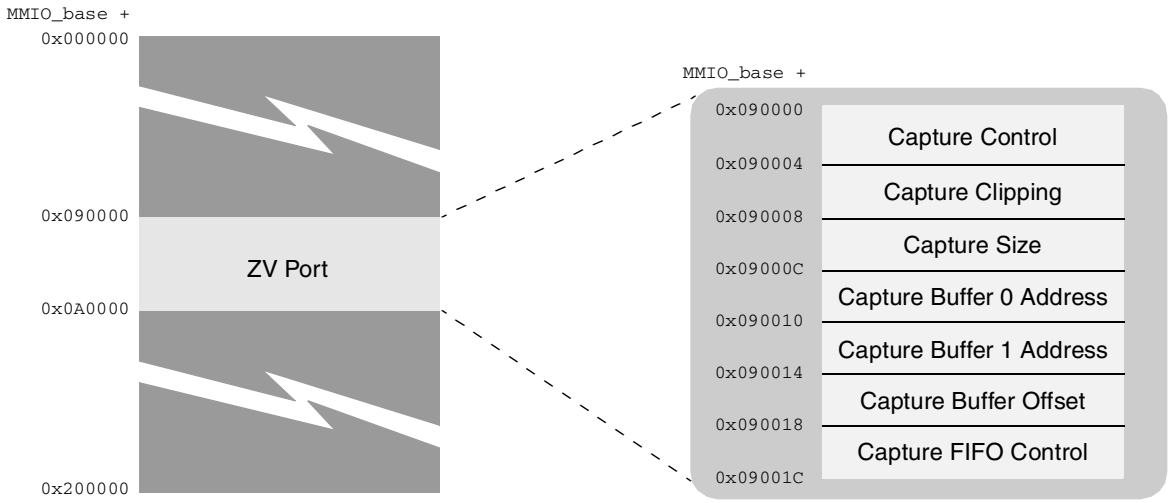
- Interlaced data and non-interlaced data capture
- Single buffer and double buffer capture
- Cropping

The SM501 uses the Video Processor block to display captured data on the display (LCD, TV, or CRT). The Video Window displays the captured data. The Video Processor does the stretching, color interpolation, YUV-to-RGB conversion, and color key functions.

Programmer's Model

Figure 10-2 shows how this 64kB region in the MMIO space is laid out. It controls the ZV Port capture registers.

Figure 10-2: ZV Port Register Space



Register Descriptions

The ZV Port registers are shown in Table 10-1.

Table 10-1: ZV Port Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
0x090000	R/W	32	0b0000.XXXX.0000.0000. 0000.0000.0000.0000	Capture Control
0x090004	R/W	32	Undefined	Capture Clipping
0x090008	R/W	32	Undefined	Capture Size
0x09000C	R/W	32	Undefined	Capture Buffer 0 Address
0x090010	R/W	32	Undefined	Capture Buffer 1 Address
0x090014	R/W	32	Undefined	Capture Buffer Offset
0x090018	R/W	32	0x00000004	Capture FIFO Control

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. In the reset values, “X” indicates don’t care.

Capture Control

Read/Write MMIO_base + 0x090000

Power-on Default 0b0000.XXXX.0000.0000.0000.0000.0000.0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				F R	I R	CB R	VSS R	Reserved					HA R/W	VS R/W	HS R/W
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FD R/W	VP R/W	HP R/W	CP R/W	UVS R/W	BS R/W	CS R/W	CF R/W	FS R/W	W R/W	B R/W	DB R/W	CC R/W	Reserved		E R/W

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	F	Field Input Status. This bit is read-only. o: Even field. 1: Odd field.
26	I	Interlace Status. This bit is read-only. 0: Non-interlaced. 1: Interlaced.

Bit(s)	Name	Description
25	CB	Current Buffer Status. This bit is read-only. 0: Capturing data into buffer 0. 1: Capturing data into buffer 1.
24	VSS	Vertical Sync Status. This bit is read-only. 0: VSync pulse is inactive. 1: VSync pulse is active.
23:19	Reserved	These bits are reserved.
18	HA	Enable Horizontal Averaging. 0: Disable. 1: Enable.
17	VS	Enable 2÷1 Vertical Shrink. 0: Disable. 1: Enable.
16	HS	Enable 2÷1 Horizontal Shrink. 0: Disable. 1: Enable.
15	FD	Field Detect Method. 0: Rising edge of VSync. 1: Falling edge of VSync.
14	VP	Select VSync Phase. 0: Active high. 1: Active low.
13	HP	Select HRef Phase. 0: Active high. 1: Active low.
12	CP	Select Input Clock Polarity. 0: Active high. 1: Active low.
11	UVS	Enable UV Swap. 0: Disable. 1: Enable.
10	BS	Enable Byte Swap. 0: Disable. 1: Enable.
9	CS	Capture Size. 0: 16-bit. 1: 8-bit.
8	CF	Capture Format. 0: YUV. 1: RGB.
7	FS	Enable Field Swap. 0: Disable. 1: Enable.
6	W	Enable Interlaced Data Capturing in Weave. 0: Disable. 1: Enable.
5	B	Enable Interlaced Data Capturing in Bob. 0: Disable. 1: Enable.
4	DB	Enable Double Buffering. 0: Disable. 1: Enable.

Bit(s)	Name	Description
3	CC	Select Capture Control. 0: Continuous capture. 1: Conditional capture by using the S bit.
2:1	Reserved	These bits are reserved.
0	E	Enable Capture. 0: Disable. 1: Enable.

Capture Clipping

Read/Write MMIO_base + 0x090004

Power-on Default Undefined

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

Bit(s)	Name	Description
31:26	Reserved	These bits are reserved.
25:16	YClip	Number of lines to skip after VSync.
15:10	Reserved	These bits are reserved.
9:0	XClip.	Number of pixels to skip after HRef.

Capture Size

Read/Write MMIO_base + 0x090008

Power-on Default Undefined

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

Bit(s)	Name	Description
31:27	Reserved	These bits are reserved.
26:16	Height	Number of lines to capture.
15:11	Reserved	These bits are reserved.
10:0	Width	Number of pixels to capture.

Capture Buffer 0 Address

Read/Write MMIO_base + 0x09000C

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Memory Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Memory Address R/W												0000			

Bit(s)	Name	Description
31	S	Status Bit. 0: Flip not pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.

Bit(s)	Name	Description
25:4	Memory Address	Memory address of capture buffer 0 with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Capture Buffer 1 Address

Read/Write MMIO_base + 0x090010

Power-on Default Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
S R/W	Reserved			Ext R/W	CS R/W	Memory Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Memory Address R/W												0 0 0 0			

Bit(s)	Name	Description
31	S	Status Bit. 0: Flip not pending. 1: Flip pending.
30:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:4	Memory Address	Memory address of capture buffer 1 with 128-bit alignment.
3:0	0000	These bits are hardwired to zeros.

Capture Buffer Offset

Read/Write MMIO_base + 0x090014
Power-on Default Undefined

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:4	Offset	Number of 128-bit aligned bytes per line of the capture buffer.
3:0	0000	These bits are hardwired to zeros.

Capture FIFO Control

Read/Write MMIO_base + 0x090018
Power-on Default 0x00000004

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

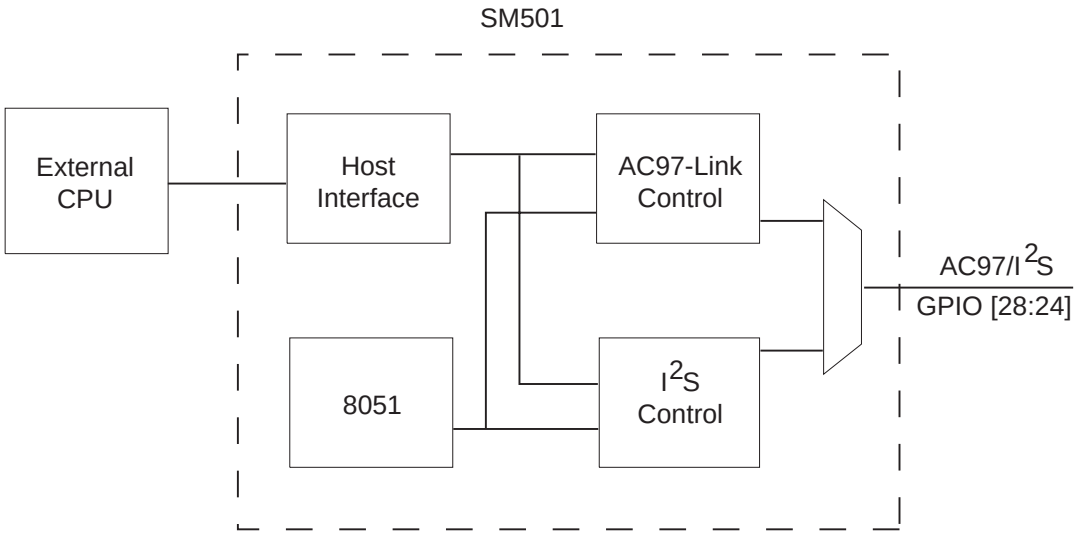
Bit(s)	Name	Description
31:3	Reserved	These bits are reserved.
2:0	FIFO	FIFO Request Level. When the FIFO empty level reaches the level specified, a FIFO read request will be generated. 000: 2 or more empty. 001: 3 or more empty. 010: 4 or more empty. 011: 5 or more empty. 100: 6 or more empty. 101: 8 or more empty. 110: 10 or more empty. 111: 12 or more empty.

11 AC97-Link & I²S

Functional Overview

Figure 11-1 shows a simplified block diagram of the AC97-link and I²S interfaces.

Figure 11-1: AC97-Link and I²S Interface Block Diagram



The SM501 provides a raw audio data transmission interface that can be configured as either an I²S or an AC97-link interface.

The AC97-link and I²S functional blocks share the same set of external pins, so only one can be active at a time. Both AC97-link and I²S can be programmed by the on-chip 8051 or an external CPU. The on-chip 8051 uses byte accesses.

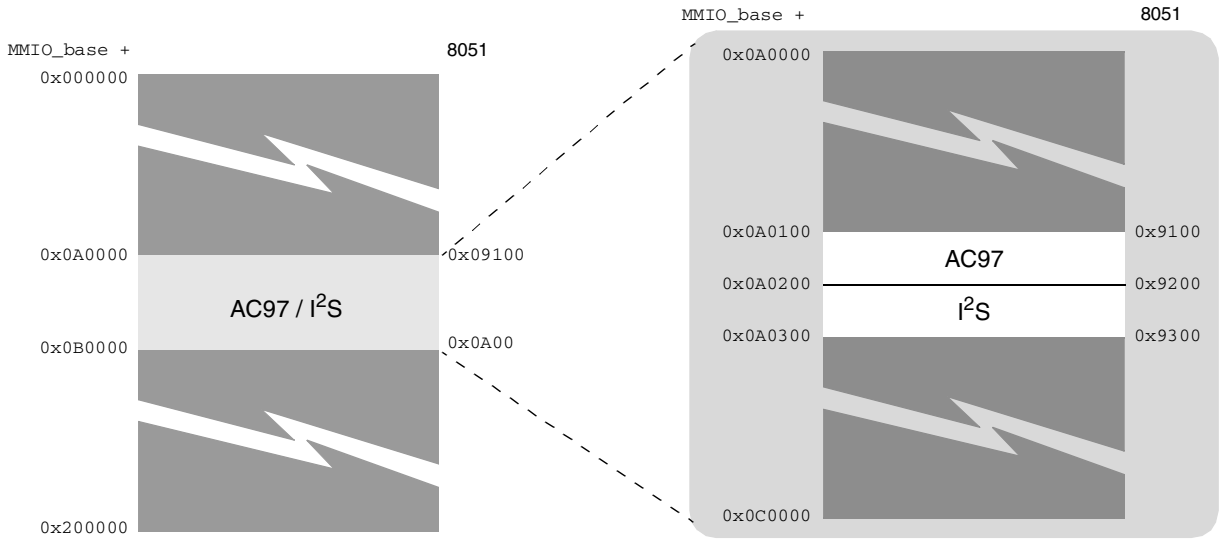
For I²S, the SM501 can be programmed to be either master or slave. For AC97-link, the SM501 acts as a controller.

For the AC97-link interface, only slot 0 to slot 4 transmissions are supported.

Programmer's Model

Figure 11-2 shows how this 64kB region in the MMIO space is laid out. It contains the registers to control the AC97-link and I^2S interfaces.

Figure 11-2:AC97-Link and I^2S Register Space



The following subsections define each region in more detail.

Register Descriptions

The AC97-link and I²S registers are shown in Table 11-1.

Table 11-1: AC97-Link and I²S Port Register Summary

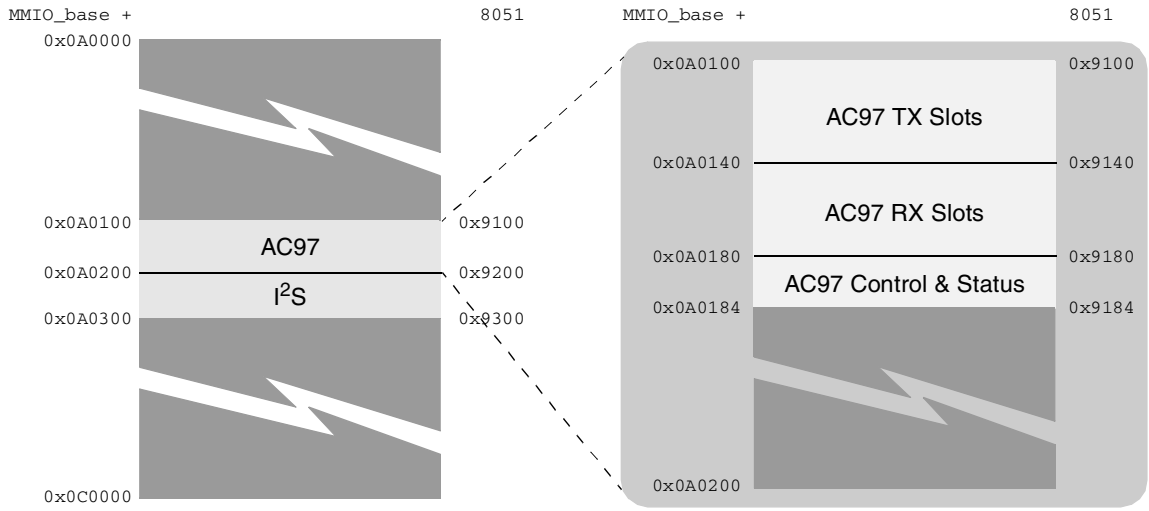
External CPU Offset from MMIO_base[1]	Internal 8051 Address[2]	Type	Width	Reset Value	Register Name
AC97-Link					
0x0A0100	0x9100	R/W	32	0x00000000	AC97 TX Slot 0: TAG
0x0A0104	0x9104	R/W	32	0x00000000	AC97 TX Slot 1: Command Address Port
0x0A0108	0x9108	R/W	32	0x00000000	AC97 TX Slot 2: Command Data Port
0x0A010C	0x910C	R/W	32	0x00000000	AC97 TX Slot 3: PCM Playback Left Channel
0x0A0110	0x9110	R/W	32	0x00000000	AC97 TX Slot 4: PCM Playback Right Channel
0x0A0140	0x9140	R	32	0x00000000	AC97 RX Slot 0: TAG
0x0A0144	0x9144	R	32	0x00000000	AC97 RX Slot 1: Status Address Port
0x0A0148	0x9148	R	32	0x00000000	AC97 RX Slot 2: Status Data Port
0x0A014C	0x914C	R	32	0x00000000	AC97 RX Slot 3: PCM Record Left Channel
0x0A0150	0x9150	R	32	0x00000000	AC97 RX Slot 4: PCM Record Right Channel
0x0A0180	0x9180	R/W	32	0x00000000	AC97 Control & Status
I²S					
0x0A0200	0x9200	R/W	32	0x00000000	I²S TX Data Left
0x0A0204	0x9204	R/W	32	0x00000000	I²S TX Data Right
0x0A0208	0x9208	R/W	32	0x00000000	I²S RX Data Left
0x0A020C	0x920C	R/W	32	0x00000000	I²S RX Data Right
0x0A0210	0x9210	R/W	32	0x00000000	I²S Control & Status
0x0A0214	0x9214	R/W	32	0x00000023	I²S Clock Control

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. The 8051 uses byte accesses.

AC97-Link Register Descriptions

Figure 11-3 shows how the 256-byte region in the MMIO space is laid out. It contains the registers to control the AC97-Link Controller.

Figure 11-3:AC97-Link Register Space



AC97 TX Slot 0: TAG

Read/Write MMIO_base + 0x0A0100 / 8051 Address 0x9100

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
V R/W	S1 R/W	S2 R/W	S3 R/W	S4 R/W	0 0 0 0 0 0 0 0 0 0 0										

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15	V	Valid Frame Tag. 0: Ignore entire frame. 1: Interpret S1 through S4 bits for valid frame slots.
14	S1	Slot 1 Valid Tag. 0: No data to transmit. 1: Data to transmit.
13	S2	Slot 2 Valid Tag. 0: No data to transmit. 1: Data to transmit.

Bit(s)	Name	Description
12	S3	Slot 3 Valid Tag. 0: No data to transmit. 1: Data to transmit.
11	S4	Slot 4 Valid Tag. 0: No data to transmit. 1: Data to transmit.
10:0	00000000000	These bits are programmed to zeros.

AC97 TX Slot 1: Command Address Port

Read/Write MMIO_base + 0x0A0104 / 8051 Address 0x9104

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19	R	Read/Write Select. 0: Write. 1: Read.
18:12	Index	Control Register Index.
11:0	000000000000	These bits are programmed to zeros.

AC97 TX Slot 2: Command Data Port

Read/Write MMIO_base + 0x0A0108 / 8051 Address 0x9108

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												Data R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Data R/W												0000			

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19:4	Data	Control Register Write Data.
3:0	0000	These bits are programmed to zeros.

AC97 TX Slot 3: PCM Playback Left Channel

Read/Write MMIO_base + 0x0A010C / 8051 Address 0x910C

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19:0	Data	Composite Digital Audio Left Playback Data.

AC97 TX Slot 4: PCM Playback Right Channel

Read/Write MMIO_base + 0x0A0110 / 8051 Address 0x9110

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19:0	Data	Composite Digital Audio Right Playback Data.

AC97 RX Slot 0: TAG

Read MMIO_base + 0x0A0140 / 8051 Address 0x9140

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
R R	S1 R	S2 R	S3 R	S4 R	Reserved										

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15	R	Codec Ready Flag. 0: Codec is not ready, ignore entire frame. 1: Codec is ready, interpret S1 through S4 bits for valid slots.
14	S1	Slot 1 Valid Tag. 0: No data to receive. 1: Data to receive.
13	S2	Slot 2 Valid Tag. 0: No data to receive. 1: Data to receive.
12	S3	Slot 3 Valid Tag. 0: No data to receive. 1: Data to receive.

Bit(s)	Name	Description
11	S4	Slot 4 Valid Tag. 0: No data to receive. 1: Data to receive.
10:0	Reserved	These bits are reserved.

AC97 RX Slot 1: Status Address Port

Read MMIO_base + 0x0A0144 / 8051 Address 0x9144

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												0	Index R		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Index R				R3 R	R4 R	Reserved									

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19	0	This bit is hardwired to 0.
18:12	Index	Control Register Index.
11	R3	On Demand Data Request Flag for Slot 3. 0: Send data next transmission. 1: Do not send data next transmission.
10	R4	On Demand Data Request Flag for Slot 4. 0: Send data next transmission. 1: Do not send data next transmission.
9:0	Reserved	These bits are reserved.

AC97 RX Slot 2: Status Data Port

Read MMIO_base + 0x0A0148 / 8051 Address 0x9148

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19:4	Data	Control Register Read Data.
3:0	Reserved	These bits are reserved.

AC97 RX Slot 3: PCM Record Left Channel

Read MMIO_base + 0x0A014C / 8051 Address 0x914C

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19:0	Data	Composite Digital Audio Left Record Data.

AC97 RX Slot 4: PCM Record Right Channel

Read MMIO_base + 0x0A0150 / 8051 Address 0x9150

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:20	Reserved	These bits are reserved.
19:0	Data	Composite Digital Audio Right Record Data.

AC97 Control & Status

Read/Write MMIO_base + 0x0A0180 / 8051 Address 0x9180

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Drop R						S R/W	B R	W R	Res	Status R		WI R/W	WR R/W	CR R/W	E R/W

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:10	Drop	Data Dropped Count. These bits are read-only.
9	S	Sync Control. 0: Enable SYNC. 1: Force SYNC to be stopped.
8	B	BClk Status. This bit is read-only. 0: Not running. 1: Running.
7	W	Wakeup Request. This bit is read-only. 0: No wakeup requested. 1: Wakeup requested.
6	Res	This bit is reserved.

Bit(s)	Name	Description
5:4	Status	AC97 Status. These bits are read-only. 00: Off. 01: Reset. 10: Waiting for wakeup. 11: Active.
3	WI	Enable Wakeup Interrupt. 0: Disabled. 1: Enabled.
2	WR	Warm Reset Control. 0: No reset. 1: Reset – need to stay active for at least 1 µs.
1	CR	Cold Reset Control. 0: No reset. 1: Reset – need to stay active for at least 1 µs.
0	E	AC97 Control. 0: Disabled. 1: Enabled.

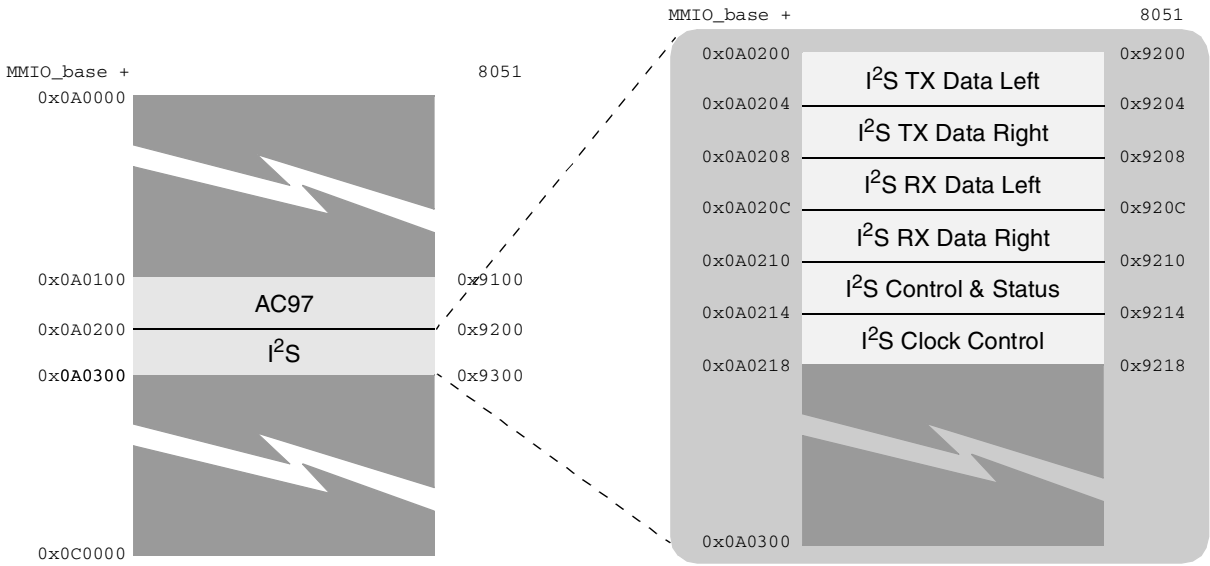
There are two ways to clear the AC97 interrupt:

1. The CPU reads this register to clear the AC97 interrupt.
2. The 8051 reads bits [15:8] of this register to clear this interrupt.

I²S Register Descriptions

Figure 11-4 shows how the 256-byte region in the MMIO space is laid out. It contains the registers to control the I²S Controller.

Figure 11-4:I²S Register Space



I²S TX Data Left

Read/Write MMIO_base + 0x0A0200 / 8051 Address 0x9200

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	Data	Data to Transmit to Left Channel.

I²S TX Data Right

Read/Write MMIO_base + 0x0A0204 / 8051 Address 0x9204

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	Data	Data to Transmit to Right Channel.

I²S RX Data Left

Read/Write MMIO_base + 0x0A0208 / 8051 Address 0x9208

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	Data	Data Received from Left Channel.

I²S RX Data Right

Read/Write MMIO_base + 0x0A020C / 8051 Address 0x920C
Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	Data	Data Received from Right Channel.

I²S Control & Status

Read/Write MMIO_base + 0x0A0210 / 8051 Address 0x9210
Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:12	Reserved	These bits are reserved.
11	R	Receive Status. This bit is read-only. 0: No error. 1: Buffer overflow.
10	T	Transmit Status. This bit is read-only. 0: No error. 1: Buffer underflow.
9:3	Reserved	These bits are reserved.
2	E	I²S Control. 0: Disabled. 1: Enabled.
1:0	Reserved	These bits are reserved.

There are two ways to clear the I^2S interrupt:

1. The CPU reads this register to clear the I^2S interrupt.
2. The 8051 reads bits [15:8] of this register to clear this interrupt.

I^2S Clock Control

Read/Write MMIO_base + 0x0A0214 / 8051 Address 0x9214

Power-on Default 0x00000023

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

Bit(s)	Name	Description
31:8	Reserved	These bits are reserved.
7	M	Mode Select. 0: Slave mode. 1: Master mode.
6:5	W	Word Select. 00: 16-bit serial clock. 01: 24-bit serial clock. 10: 32-bit serial clock.
4:0	C	Clock Divider. BitClock = 16.9344MHz ÷ (2 * (1 + *C*)). SampleRate = BitClock ÷ (2 * WordLength).

12 8051 µ-Controller

Functional Overview

The 8051 µ-controller is an industrial-standard microcontroller. It is embedded in the SM501 to offload the host CPU from some I/O controlling tasks, such as I^2S, AC97-link, and USB slave control. The 8051 also can be used to control a 16-bit external bus for interfacing with external devices.

The 8051 has the following memory allocated to it:

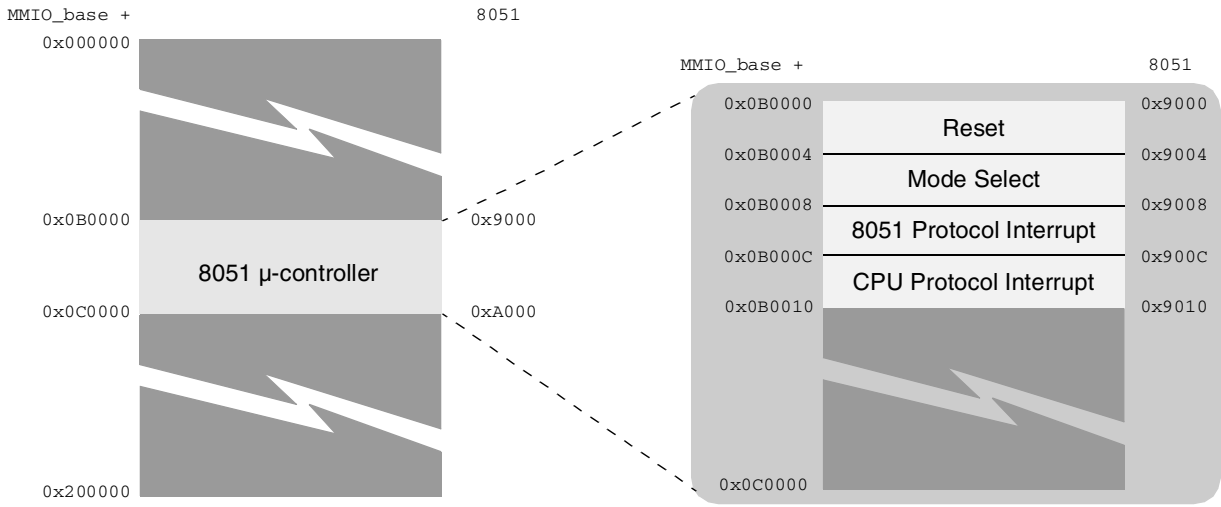
- 12kB of memory for use as program and data memory
- 4kB of SRAM to be shared between the 8051 and the host CPU

Two interrupts (A host CPU to 8051 interrupt and an 8051 to host CPU interrupt) help communications between the 8051 µ-controller and the host CPU.

Programmer's Model

Figure 12-1 shows how this 64kB region in the MMIO space is laid out. It contains the registers that control the 8051 µ-controller.

Figure 12-1: 8051 µ-Controller Register Space



The following sections define each region in more detail.

Register Descriptions

The 8051 µ-controller registers are shown in Table 12-1.

Table 12-1: 8051 µ-Controller Register Summary

External CPU Offset from MMIO_base[1]	Internal 8051 Address	Type	Width	Reset Value	Register Name
0x0B0000	—[2]	R/W	32	0x00000000	Reset
0x0B0004	0x9004	R/W	32	0x00000000	Mode Select
0x0B0008	0x9008	R	32	0x00000000	8051 Protocol Interrupt
0x0B000C	0x900C	W	32	0x00000000	CPU Protocol Interrupt

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. Not accessible.

Reset

Read/Write MMIO_base + 0x0B0000

Power-on Default 0x00000000

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

Note: The 8051 cannot access this register.

Bit(s)	Name	Description
31:1	Reserved	These bits are reserved.
0	E	8051 µ-Controller Control. 0: Keep 8051 µ-controller in reset mode. 1: Enable 8051 µ-controller.

Make sure *E* is 0 when downloading code into the 12kB Program/Data SRAM.

Mode Select

Read/Write MMIO_base + 0x0B0004
Read 8051 Address 0x9004
Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved								USB R/W R		M R/W R	I R/W R	T R/W R	C R/W R	S R/W R	

Bit(s)	Name	Description
31:8	Reserved	These bits are reserved.
7:6	USB	USB Slave Wait State Select. 00: No wait state (1 clock). For the 8051, clk < 24 MHz. 01: 1 wait state (3 clocks). For the 8051, clk is between 24 MHz and 40 MHz. 10: 2 wait states (5 clocks). For the 8051, clk > 40 MHz.
5	M	SRAM Memory Control. 0: Turn on SRAM. 1: Turn off SRAM.
4	I	External Interface Select (see Table 12-2 on page 7) 0: 12-bit address. 1: 8-bit address plus 4 control output (P2[3:0]).
3	T	AC-Link/I^2S Test Mode. 0: Normal AC-Link/I2S operation. 1: Test mode.
2	C	Codec Select. 0: AC-Link. 1: I^2S.
1:0	S	8051 µ-Controller Speed Select. 00: HIF clock ÷ 2. 01: HIF clock ÷ 3. 10: HIF clock ÷ 4. 11: HIF clock ÷ 5.

8051 Protocol Interrupt

Read MMIO_base + 0x0B0008
Read 8051 Address 0x9008
Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:0	Token	Protocol token send out by 8051 µ-controller. When the 8051 µ-controller writes to bits [7:0], an interrupt is generated to the CPU. The interrupt is cleared when the CPU reads the *8051 Protocol Interrupt* register.

CPU Protocol Interrupt

Write MMIO_base + 0x0B000C
Read 8051 Address 0x900C
Power-on Default 0x00000000

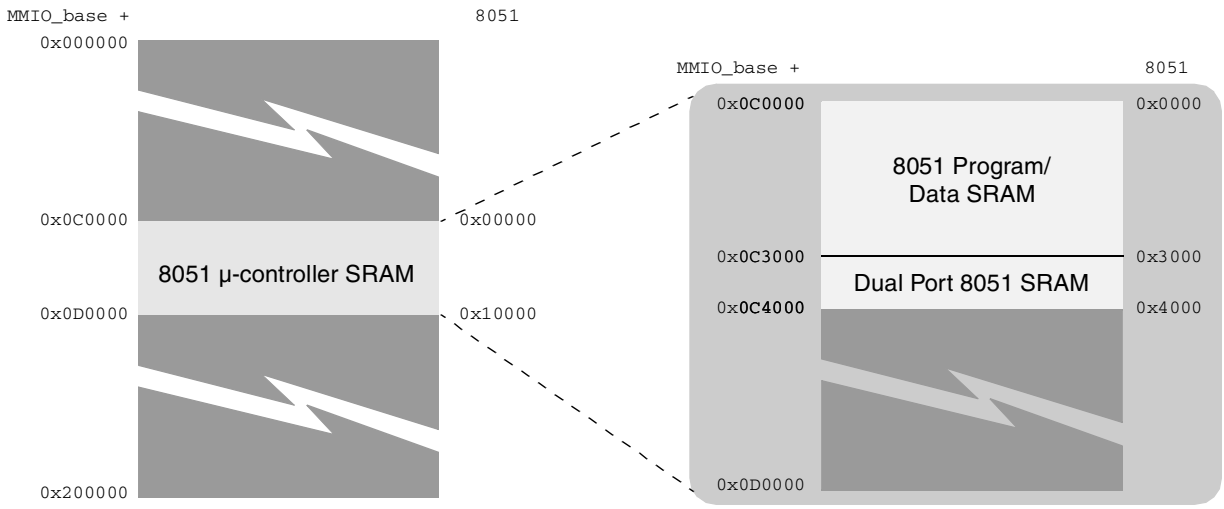
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Token W															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Token W															

Bit(s)	Name	Description
31:0	Token	Protocol token send out by CPU. When the CPU writes to bits [31:0], an interrupt is generated to the 8051 µ-controller. The interrupt is cleared when the 8051 reads this register.

8051 µ-Controller SRAM Address Space

Figure 12-2 shows how this 64kB region in the MMIO space is laid out. It contains 16kB of SRAM for use by the 8051 µ-controller.

Figure 12-2: 8051 µ-Controller SRAM Address Space



The first 12kB of SRAM contain the program code and data for the 8051 µ-controller. The host CPU can initialize this memory with the program and data code and can let the 8051 µ-controller execute the code by waking up the 8051 from Reset mode.

The host CPU can access the first 12kB of SRAM only when the 8051 is in Reset mode.

The 4kB of Dual Port 8051 SRAM acts as a data buffer for communication between the 8051 µ-controller and the host CPU. Care should be taken to never let the CPU and 8051 µ-controller access the same memory location at the same time. This can be done by using software protocols and interrupt handshaking:

1. The CPU writes some data into any address inside the 4kB region.
2. The CPU generates an interrupt to the 8051 and sends address and size as a token to the 8051.
3. The 8051 starts working on the data and generates an interrupt back to the CPU when it is finished.
4. In the meantime, the CPU can start filling another chunk of data inside the 4kB region and can send the 8051 that address and size when the 8051 interrupts the CPU from step 3.

8051-Controlled External Bus Interface

GPIO[15:0] can be programmed as an 8051 external bus interface through GPIO [31:10] and Control Register (0008). All 8051 access to address 0X8000 to 0X8FFF will generate bus cycles on GPIO [15:0], if GPIO [15:0] is programmed as an 8051 external bus.

In 12-bit address mode, the address lines can either be send directly out from the 8051 µ-controller, or they can be latched by the ALE# signal. This way the 8051 µ-controller can be very flexible in protocol programming. For example, it could be a very simple protocol or a more complicated MOST protocol.

In 8-bit address mode, the GPIO[15:12] pins are connected to the 8051 µ-controller P2[3:0] outputs.

Table 12-2 shows the pin assignment for both 8-bit and 12-bit address mode interfaces.

Table 12-2: GPIO[15:0] in 8-bit and 12-bit Modes

Mode	GPIO[15:12][1]	GPIO[11]	GPIO[10]	GPIO9	GPIO8	GPIO[7:0]
8-bit Address Mode	P2[3:0] output	Ext_wait	ALE	WRn	RDn	AD[7:0]
12-bit Address Mode	Addr[11:8]	Ext_wait	ALE	WRn	RDn	AD[7:0]

1. GPIO[15:12] is either a latched or unlatched version of Addr[11:8]/P2[3:0], controlled by Miscellaneous Control Register, bit 26 (see “Miscellaneous Control” on page 2-6).

Bit 4 in the Mode Select register (0x0B0004) selects between 8-bit and 12-bit address modes.

GPIO bits 48:53 are tied to the 8051 µ-controller port P1 input bits 0:5, and can be used as either standard GPIO inputs (the 8051 µ-controller ignores them), interrupts for external devices, or 8051 µ-controller input pins. Bit 54 of GPIO is tied to the 8051 µ-controller INT0# line.

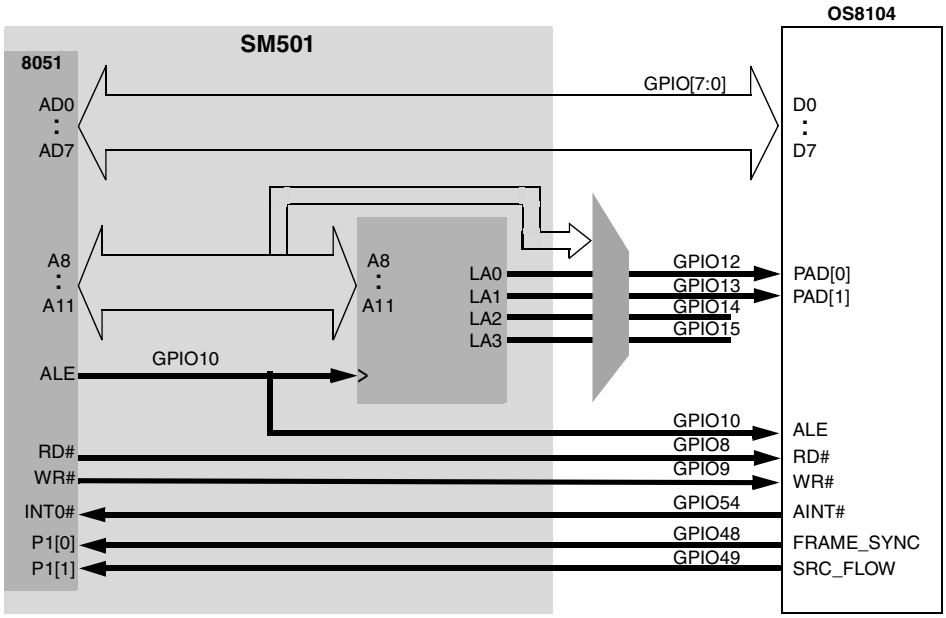
Example: Implementing MOST for Automotive Usage

The 8051 µ-controller and 8-bit parallel interface of the SM501 can be used to interface with an external MOST controller chip. The 8051 µ-controller will be programmed with firmware that can interface to an external MOST controller using either Asynchronous mode or Combined-Parallel mode.

The example shown in Figure 12-3 uses the Oasis MOST controller.

The 8051 µ-controller parallel interface is programmed in 12-bit address mode, with latching turned on. The WR# and RD# signals are directly connected to the MOST controller. The latched address bits A9:8 are connected to the MOST controller 2-bit address bus. The data bits AD7:0 are connected to the MOST controller 8-bit data bus. The MOST controller control and interrupt lines are connected to the SM501's GPIO pins.

Figure 12-3: 8051 µ-Controller to MOST Controller Connections

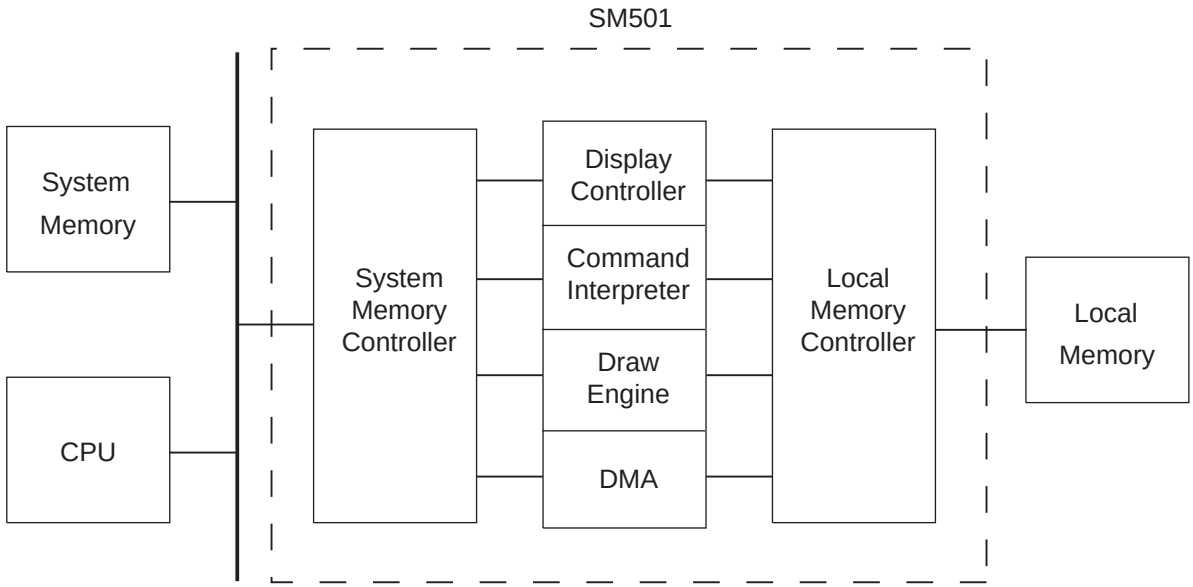


13 DMA Controller (DMAC)

Functional Overview

In the SM501, the Display Controller, Command Interpreter, Draw Engine, and DMA can access system memory through the on-chip system memory controller (see Figure 13-1).

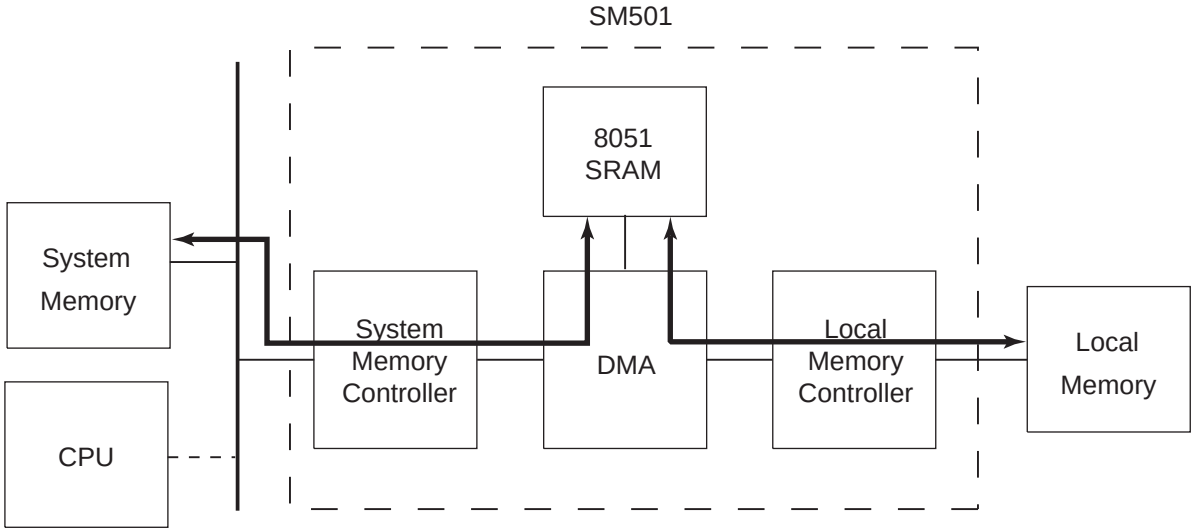
Figure 13-1: SM501 Functional Block Connections to Memory Controllers



Two DMA channels, DMA0 and DMA1, within the SM501 handle memory data transfers, thus offloading the CPU. One DMA channel moves data between external/internal SDRAM and the 8051's SRAM; the other DMA channel moves data between internal and external memory.

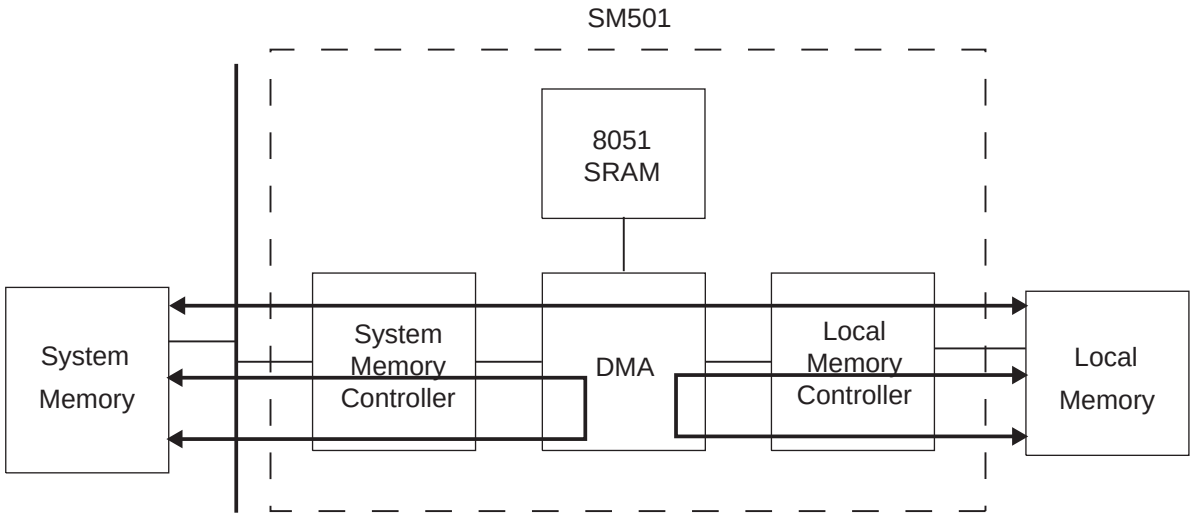
- DMA0—Moves data between local or system memory and the internal 8051's SRAM (see Figure 13-2)
  There are four ways to move data in DMA0:
  - System memory to 8051 SRAM
  - 8051 SRAM to system memory
  - Local memory to 8051 SRAM
  - 8051 SRAM to local memory

Figure 13-2: DMA Channel 0



- DMA1—Moves data between system memory and local memory (see Figure 13-3)
  There are four ways to transfer data in DMA1:
  - System memory to system memory
  - System memory to local memory
  - Local memory to system memory
  - Local memory to local memory

Figure 13-3: DMA Channel 1



Register Descriptions

Figure 13-4 shows how this 64kB region in the MMIO space is laid out. It controls the DMA registers.

Figure 13-4: DMA Register Space

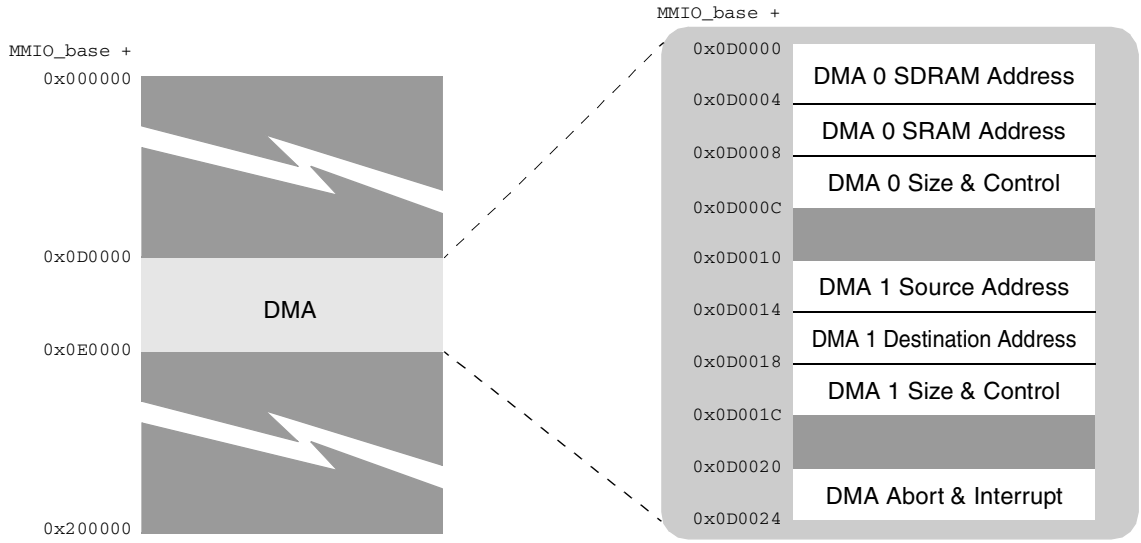


Table 13-1 shows the DMA Controller registers.

Table 13-1: DMA Controller Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Register Name
0x0D0000	R/W	32	0x00000000	DMA 0 SDRAM Address
0x0D0004	R/W	32	0x00000000	DMA 0 SRAM Address
0x0D0008	R/W	32	0x00000000	DMA 0 Size & Control
0x0D0010	R/W	32	0x00000000	DMA 1 Source Address
0x0D0014	R/W	32	0x00000000	DMA 1 Destination Address
0x0D0018	R/W	32	0x00000000	DMA 1 Size & Control
0x0D0020	R/W	32	0x00000000	DMA Abort & Interrupt

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.

DMA 0 SDRAM Address

Read/Write MMIO_base + 0x0D0000

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Memory Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Memory Address R/W														0	0

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:2	Memory Address	Memory address with 32-bit alignment.
1:0	00	These bits are hardwired to zeros.

DMA 0 SRAM Address

Read/Write MMIO_base + 0x0D0004

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:2	Memory Address	Memory address with 32-bit alignment.
1:0	00	These bits are hardwired to zeros.

DMA 0 Size & Control

Read/Write MMIO_base + 0x0D0008

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Act R/W	Dir R/W	Reserved													
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Size R/W														0	0

Bit(s)	Name	Description
31	Act	DMA Channel 0 Activation. The *Act* bit will be cleared to “0” by the hardware when the DMA is finished. 0: Idle. 1: Activate DMA channel 0.
30	Dir	Direction of DMA. 0: Read from SDRAM and write to 8051 SRAM. 1: Read from 8051 SRAM and write to SDRAM.
29:16	Reserved	These bits are reserved.
15:2	Size	Number of 32-bit aligned bytes to transfer.
1:0	00	These bits are hardwired to zeros.

DMA 1 Source Address

Read/Write MMIO_base + 0x0D0010

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Memory Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Memory Address R/W														0	0

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.

Bit(s)	Name	Description
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:2	Memory Address	Memory address with 32-bit alignment.
1:0	00	These bits are hardwired to zeros.

DMA 1 Destination Address

Read/Write MMIO_base + 0x0D0014

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				Ext R/W	CS R/W	Memory Address R/W									
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Memory Address R/W														0	0

Bit(s)	Name	Description
31:28	Reserved	These bits are reserved.
27	Ext	Memory Selection. 0: Local memory. 1: External memory.
26	CS	Chip Select for External Memory. 0: CS0 of external memory. 1: CS1 of external memory.
25:2	Memory Address	Memory address with 32-bit alignment.
1:0	00	These bits are hardwired to zeros.

DMA 1 Size & Control

Read/Write MMIO_base + 0x0D0018

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31	Act	DMA Channel 1 Activation. The *Act* bit will be cleared to "0" by the hardware when the DMA is finished. 0: Idle. 1: Activate DMA channel 1.
30:24	Reserved	These bits are reserved.
23:2	Size	Number of 32-bit aligned bytes to transfer (up to 16MB).
1:0	00	These bits are hardwired to zeros.

DMA Abort & Interrupt

Read/Write MMIO_base + 0x0D0020

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved										$Abort_{1:0}$ R/W		Reserved		$Int_{1:0}$ R/W	

Bit(s)	Name	Description
31:6	Reserved	These bits are reserved.
5:4	$Abort_{1:0}$	Enable or Abort DMA Channel. Aborting will reset the corresponding DMA controller. For normal operation, the *Abort* bits should be set to "0". 0: Enable corresponding DMA channel. 1: Abort corresponding DMA channel.

Bit(s)	Name	Description
3:2	Reserved	These bits are reserved.
1:0	$Int_{1:0}$	Interrupt Status Bit. The *Int* bit should be cleared to “0” by software when the interrupt has been serviced. Writing a “1” has no effect. 0: DMA is not active or still busy – no interrupt. 1: DMA is finished – interrupt.

14 UART

Functional Overview

UART0 and UART 1 perform these functions:

- Serial-to-parallel conversion on data received from a peripheral device
- Parallel-to-serial conversion on data transmitted to the peripheral device

The CPU reads and writes data and control/status information through the CPU interface. The transmit and receive paths are buffered with internal FIFO memories enabling up to 64 bytes to be stored independently in both transmit and receive modes.

The UARTs:

- Include a programmable baud rate generator that generates a common transmit and receive internal clock from the UART internal reference clock input, **UARTCLK**
- Offer similar functionality to the industry-standard 16C550 UART device
- Support baud rates of up to 460.8Kbits/s, subject to **UARTCLK** reference clock frequency

The UART operation and baud rate values are controlled by the Line Control register and the Baud Rate Divisor registers.

The UARTs can generate:

- Individually maskable interrupts from the receive (including timeout), transmit, modem status and error conditions
- A single combined interrupt so that the output is asserted if any of the individual interrupts are asserted and unmasked

If a framing, parity, or break error occurs during reception, the appropriate error bit is set, and is stored in the FIFO. If an overrun condition occurs, the overrun register bit is set immediately and FIFO data is prevented from being overwritten.

The input modem status signal, Clear To Send (nCTS), and output modem control line, Request To Send (nRTS), are supported.

There is a programmable hardware flow control feature that uses the nCTS input and the nRTS output to automatically control the serial data flow.

Modem Signals

The Request To Send (nRTS) output is controlled through the Modem Control register. Read the Modem Status register to get the status of the Clear To Send (nCTS) input. The UART can be programmed to generate an interrupt any time nCTS is asserted. The nDSR, nRI, and nDCD modem signals are tied to a logic 1.

Data Reception

Data is clocked into the RX Shift register according to the divided-by-16 Receive clock. The Baud Rate Generator creates this clock. When 32 bits have been clocked into the Receiver, they are sent to either the RX Buffer register or the RX FIFO (if enabled) for reading by the CPU. Bit 0 of this register contains the first bit of

data to be received. The Receiver also checks for the stop and parity bits within this data, as specified by the Line Control register.

Each UART can be programmed to generate several receive interrupts, such as:

- an RX Data Received interrupt when the data is transferred to the RX Buffer register or when the RX Trigger Level is reached (if FIFOs are enabled)
- incorrect parity
- RX FIFO character timeout
- missing stop bits (frame errors)
- line status errors

When the RX FIFO is enabled through the FIFO Control register, the RX FIFO can receive up to 64 bytes of data. IRQ is asserted when the RX FIFO contains one byte of data, is a quarter full, is half full, or has only two bytes empty.

Data Transmission

Data transmission begins when the transmit data is written to the TX Holding register or the TX FIFO (if enabled). The transmit data is subsequently sent to the TX Shift register with the addition of any applicable start, stop, or parity bits as determined by the Line Control register. The transmit bits are then shifted out of the TX Shift register in the following order:

1. Start bit
2. Data bits (least-significant bit first)
3. Parity bit
4. Stop bit

These bits are clocked according to the Baud Rate Generator clock, which is divided by 16.

Each UART can be programmed to generate a TX Holding Register Empty interrupt when the TX Holding register or the TX FIFO (if enabled) becomes empty.

When the TX FIFO is enabled through the FIFO Control register, the TX FIFO can store up to 64 bytes of data. Transmission proceeds until the TX FIFO is empty. The IRQ signal determines whether or not the FIFO can accept more data.

Hardware Flow Control

Each UART supports separate hardware transmission and reception flow control. The Enhanced Feature register handles the enabling of both functions. When hardware flow control is enabled, the nCTS and nRTS signals control the transmission and reception of data, respectively.

When hardware transmission flow control is enabled, the UART disables transmission of characters when the nCTS signal is sampled when it is inactive (logic 1). All character transmissions that are currently in progress will complete. Further data transmission will resume when nCTS is sampled active (logic 0).

When hardware reception flow control is enabled, the UART drives nRTS to its inactive state (logic 1) when the RX FIFO exceeds its trigger level. nRTS is driven active when the RX FIFO falls below its trigger level.

Note: Hardware reception flow control is enabled only when the RTS bit in the Modem Control register is set. The RTS bit can be used independent of this feature for flow control.

Software Flow Control

When software flow control is enabled, it controls data transmission and reception by the transfer of defined XON and XOFF characters. Each XON/XOFF character can be associated with up to two bytes; these can be chosen to represent either a single or a double character. The UART contains four XON/XOFF registers: XON 1, XON 2, XOFF 1, and XOFF 2. The Enhanced Features register determines which XON/XOFF registers are to be used. On reset, the characters in the XON/XOFF registers are defined by configuration constants. The default is all zeros for compatibility with existing driver software.

In software flow control, received data characters are compared with the XOFF character. Upon a match, the transmission channel is disabled when the current character finishes transmission. If the XOFF interrupt is enabled through the Interrupt Enable register, an interrupt is generated at this point. The transmission channel is re-enabled when an XON character is received.

Note: If a single character XON/OFF is selected, then received XON/XOFF characters are not placed in the RX FIFO. When a parity or framing error occurs during reception of an XON/XOFF character, the received character is treated as normal data, not as an XON/XOFF character.

When the XOFF character is received, the XOFF status bit in the Modem Control register is set (if software flow control is enabled). This read-only bit subsequently is cleared when an XON character is received. The XOFF status bit also is cleared upon a reset. Writing to this bit has no effect.

In software flow control, the XOFF character is sent when the RX FIFO exceeds its threshold level. When the RX FIFO falls below its threshold level, the XON character is sent, thus re-enabling transmission from the other end of the link.

Four bits in the Enhanced Features register determine the operation for software flow control as indicated in Table 14-1.

Table 14-1: Software Flow Control Configuration Bits

Enhanced Features Bits [3:0]	Software Flow Control Option
XX00	No RX flow control.
XX01	Receive XON2/XOFF2 as flow control bytes.
XX10	Receive XON1/XOFF1 as flow control bytes.
XX11	Receive XON1+XON2 and XOFF1+XOFF2 as flow control words.
00XX	No TX flow control.
01XX	Transmit XON2/XOFF2 as flow control bytes.
10XX	Transmit XON1/XOFF1 as flow control bytes.
11XX	Transmit XON1+XON2 and XOFF1+XOFF2 as flow control words.

UART Timings

Transmit Engine

The Transmit Engine begins operation two to three baud clocks from the time that the TX Holding register or the TX FIFO is written. The SOUT signal is driven low seven to eight baud clocks later.

FIFO Reset Timing

When bits 3:0 of the FIFO Control register are being used to clear the FIFOs, note the following with regards to some of these bits:

- Bit 0, FIFOE, enables/disables the FIFOs. When there is a master reset, both FIFOs are reset and remain in the reset state unless FIFOE is set to 1.
- Bit 1, CLRR, clears the RX FIFO. When this bit is set, the RX FIFO is cleared after at least one clock, as is this self-clearing bit.
- Bit 2, CLRT, clears the TX FIFO. When this bit is set, the TX FIFO is cleared after at least one clock, as is this self-clearing bit.

TX Holding Register Empty Interrupt Timing

A TX Holding Register Empty interrupt is generated 17 to 18 clocks after data is written to the TX Holding Register, assuming that the Transmit Engine was idle when the data was written. If the TX Holding register is empty and the TX Holding Register Empty interrupt is enabled, an interrupt is generated immediately.

Baud Rate Generator Timing

The output of the Baud Rate Generator differs from the 16550 operation as follows. A division by 1 produces a logic 1 signal.

IrDA Modulation/Demodulation

Each SM501 UART provides basic IrDA 1.0 SIR modulation and demodulation. The UARTs use the x16 transmit and receive clocks to generate 3/16 width pulses.

Register Descriptions

Table 14-2 shows the UART0/IrDA0 registers. Table 14-3 shows the UART1/IrDA1 registers.

Table 14-2: UART0 and IrDA0 Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Line Control Register Setting	Register Name
Normal Mode (DLAB = 0)					
0x030000	R	8	0x00	0b0XXXXXXX	RX Buffer
0x030000	W	8	0x00	0b0XXXXXXX	TX Holding
0x030004	R/W	8	0x00	0b0XXXXXXX	Interrupt Enable
0x030008	R	8	0x01	Not 0xBF	Interrupt Identification
0x030008	W	8	0x00	Not 0xBF	FIFO Control
0x03000C	R/W	8	0x00	—	Line Control
0x030010	R/W	8	0x00	Not 0xBF	Modem Control
0x030014	R	8	0x60	Not 0xBF	Line Status
0x030018	R	8	0x00	Not 0xBF	Modem Status
0x03001C	R/W	8	—	Not 0xBF	Scratch
Baud Rate Generator Divisor Registers (DLAB = 1)					
0x030000	R/W	8	0x01	0b1XXXXXXX	Divisor Latch (DLL)
0x030004	R/W	8	0x00	0b1XXXXXXX	Divisor Latch (DLM)
Enhanced UART Registers (Line Control Register = 0xBF)					
0x030008	R/W	8	0x00	0xBF	Enhanced Feature
0x030010	R/W	8	0x00	0xBF	XON1 Character
0x030014	R/W	8	0x00	0xBF	XON2 Character
0x030018	R/W	8	0x00	0xBF	XOFF1 Character
0x03001C	R/W	8	0x00	0xBF	XOFF2 Character

1. Refer to Table 1-5 on page 1-28 for MMIO_base values depending on the CPU.

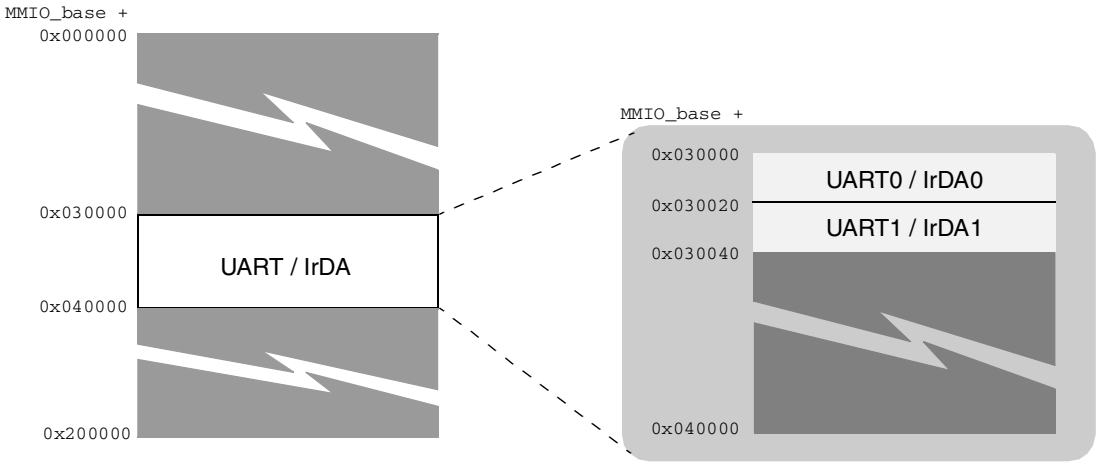
Table 14-3: UART1 and IrDA1 Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Line Control Register Setting	Register Name
Normal Mode (DLAB = 0)					
0x030020	R	8	0x00	0b0XXXXXXX	RX Buffer
0x030020	W	8	0x00	0b0XXXXXXX	TX Holding
0x030024	R/W	8	0x00	0b0XXXXXXX	Interrupt Enable
0x030028	R	8	0x01	Not 0xBF	Interrupt Identification
0x030028	W	8	0x00	Not 0xBF	FIFO Control
0x03002C	R/W	8	0x00	—	Line Control
0x030030	R/W	8	0x00	Not 0xBF	Modem Control
0x030034	R	8	0x60	Not 0xBF	Line Status
0x030038	R	8	0x00	Not 0xBF	Modem Status
0x03003C	R/W	8	—	Not 0xBF	Scratch
Baud Rate Generator Divisor Registers (DLAB = 1)					
0x030020	R/W	8	0x01	0b1XXXXXXX	Divisor Latch (DLL)
0x030024	R/W	8	0x00	0b1XXXXXXX	Divisor Latch (DLM)
Enhanced UART Registers (Line Control Register = 0xBF)					
0x030028	R/W	8	0x00	0xBF	Enhanced Feature
0x030030	R/W	8	0x00	0xBF	XON1 Character
0x030034	R/W	8	0x00	0xBF	XON2 Character
0x030038	R/W	8	0x00	0xBF	XOFF1 Character
0x03003C	R/W	8	0x00	0xBF	XOFF2 Character

1. Refer to Table 1-5 on page 1-28 for MMIO_base values depending on the CPU.

Figure 14-1 shows how the 64kB region in the MMIO space is laid out. It controls the UART0/1 and IrDA0/1 registers.

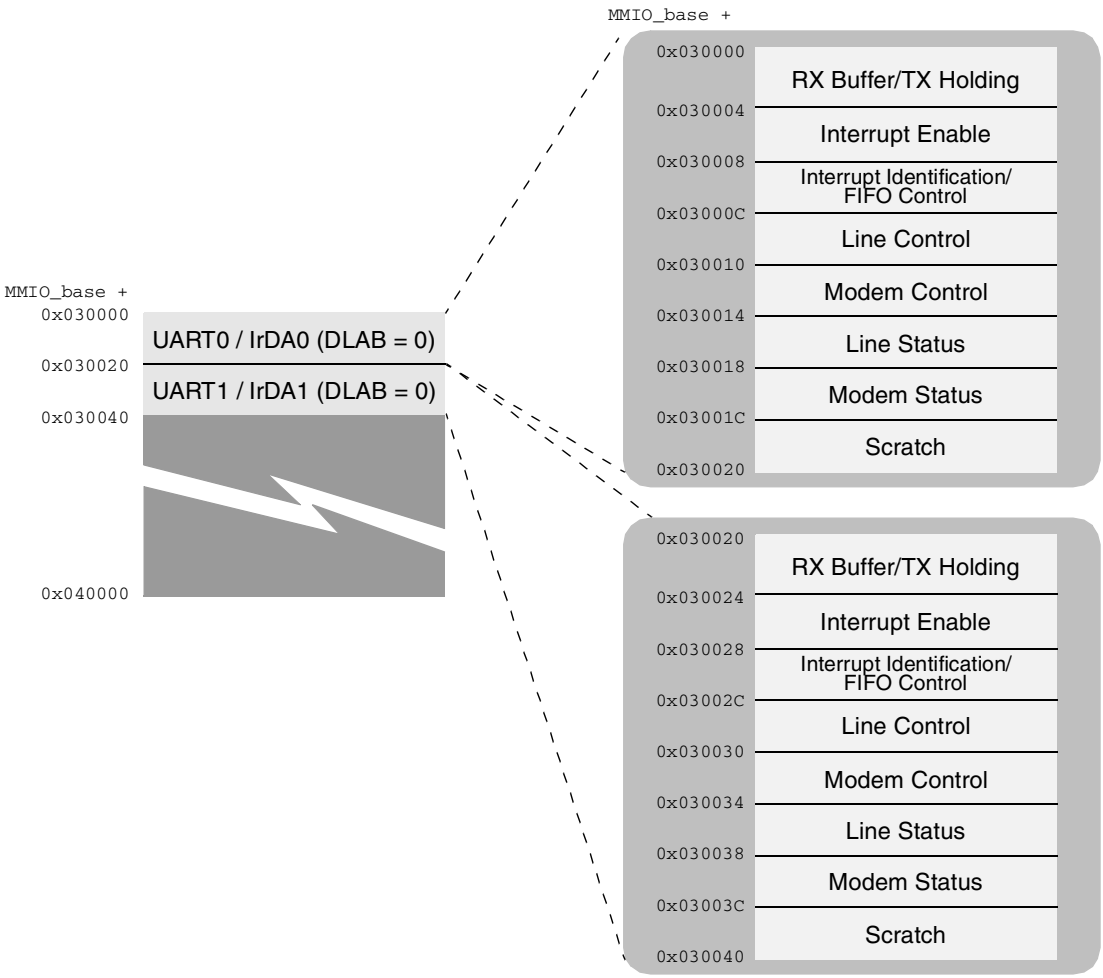
Figure 14-1: UART and IrDA Register Space



Normal Mode Registers (DLAB = 0)

The UART / IrDA is working in “normal mode” when DLAB (bit [7] of the Line Control register) is “0”. The UART / IrDA address space supports eight registers as defined in Figure 14-2.

Figure 14-2: UART / IrDA Normal Mode Registers (DLAB = 0)



RX Buffer

Read MMIO_base + 0x030000 / MMIO_base + 0x030020

Power-on Default 0x00

7	6	5	4	3	2	1	0
Data R							

Bit(s)	Name	Description
7:0	Data	Data received from RX Shift register or RX FIFO.

TX Holding

Write MMIO_base + 0x030000 / MMIO_base + 0x030020

Power-on Default 0x00

7	6	5	4	3	2	1	0
Data W							

Bit(s)	Name	Description
7:0	Data	Data to be sent to TX Shift register or TX FIFO.

Interrupt Enable

Read/Write MMIO_base + 0x030004 / MMIO_base + 0x030024

Power-on Default 0x00

7	6	5	4	3	2	1	0
CTSI R/W	RTSI R/W	XOFFI R/W	Reserved	EDSSI R/W	ELSI R/W	ETBEI R/W	ERBFI R/W

Bit(s)	Name	Description
7	CTSI	CTS Interrupt. An interrupt is generated when a rising edge is detected on the CTS modem control line. The CTS interrupt will only be generated if hardware flow control is enabled (Enhanced Mode). 0: Disable CTS interrupt. 1: Enable CTS interrupt.
6	RTSI	RTS Interrupt. An interrupt is generated when a rising edge is detected on the RTS modem control line. The RTS interrupt will only be generated if hardware flow control is enabled (Enhanced Mode). 0: Disable RTS interrupt. 1: Enable RTS interrupt.
5	XOFFI	XOFF Interrupt. An interrupt is generated when a XOFF character is received. The XOFF interrupt will only be generated if software flow control is enabled (Enhanced Mode). 0: Disable XOFF interrupt. 1: Enable XOFF interrupt.
4	Reserved	This bit is reserved.
3	EDDSI	Enable Modem Status Interrupt. An interrupt is generated if any of the DDCD, TERI, DDSR, or DCTS (bits [3:0] of the Modem Status register) becomes set. 0: Disable Modem Status interrupt. 1: Enable Modem Status interrupt.
2	ELSI	Enable RX Line Status Interrupt. An interrupt is generated if any of the *BI*, *FE*, *PE*, or *OE* (bits [4:1] of the Line Status register) becomes set. 0: Disable RX Line Status interrupt. 1: Enable RX Line Status interrupt.
1	ETBEI	Enable TX Holding Empty Interrupt. An interrupt will be generated when THRE (bit [5] of the Line Status register) becomes set. 0: Disable TX Holding Empty interrupt. 1: Enable TX Holding Empty interrupt.
0	ERBFI	Enable RX Buffer Interrupt. An interrupt is generated when DR (bit [0] of the Line Status register) becomes set. If the FIFOs are enabled, setting this bit also enables the RX FIFO Character Timeout Interrupt. 0: Disable RX Buffer Full interrupt. 1: Enable RX Buffer Full interrupt.

Interrupt Identification

Read MMIO_base + 0x030008 / MMIO_base + 0x030028

Power-on Default 0x01

7	6	5	4	3	2	1	0
FIFOE R		ID4 R	ID3 R	ID2 R	ID1 R	ID0 R	NINT R

Bit(s)	Name	Description
7:6	FIFOE	FIFO Enable. 00: FIFOs not enabled. 11: FIFOs enabled.
5:1	ID4 through ID0	Interrupt IDs. ID4 and ID3 are part of Enhanced Mode (that is, bit 4 in the Enhanced Features register is set).
0	NINT	No Interrupt Pending. 0: Interrupt pending. 1: No interrupt pending.

The following table gives the interrupt ID codes associated with the possible interrupts:

ID	Priority	Interrupt Source	Cleared by:
00011	1 (highest)	RX Line Status. BI, FE, PE, or OE bit set in Line Status register.	Reading the Line Status register.
00010	2	RX Data Received. RX data received or TX Trigger Level reached.	Reading the RX Buffer register or the RX FIFO (if enabled).
00110	2	RX FIFO Character Timeout. Timeout on character in RX FIFO.	Reading the RX FIFO.
00001	3	TX Holding Register Empty. TX Holding register empty or TX FIFO empty.	Writing to the TX Holding register or TX FIFO (if enabled) or reading Interrupt Identification register if event was priority 3 interrupt.
00000	4	Modem Status Change. DDCD, TERI, DDSR, or DCTS bit set in Modem Status register.	Reading the Modem Status register.
01000	5	Software Flow Control. XOFF character received.	Reading Interrupt Identification register if event was priority 5 interrupt.
10000	6 (lowest)	Hardware Flow Control. CTS or RTS rising edge.	Reading Interrupt Identification register if event was priority 6 interrupt.

FIFO Control

Write MMIO_base + 0x030008 / MMIO_base + 0x030028

Power-on Default 0x00

7	6	5	4	3	2	1	0
RFTL W		TFTL W		DMA1 W	CLRT W	CLRR W	FIFOE W

Bit(s)	Name	Description
7:6	RFTL	RX FIFO Trigger Level. 00: 1. 01: 16. 10: 32. 11: 62.
5:4	TFTL	TX FIFO Trigger Level. Enhanced Mode. 00: 1. 01: 16. 10: 32. 11: 62.
3	DMA1	DMA Mode 1. This bit is reserved.
2	CLRT	Clear TX FIFO. The *CLRT* bit is self clearing. 0: No action. 1: Empty TX FIFO.
1	CLRR	Clear RX FIFO. The *CLRR* bit is self clearing. 0: No action. 1: Empty RX FIFO.
0	FIFOE	FIFO Enabled. 0: Disable FIFOs. 1: Enable FIFOs.

Line Control

Read/Write MMIO_base + 0x03000C / MMIO_base + 0x03002C

Power-on Default 0x00

7	6	5	4	3	2	1	0
DLAB R/W	SB R/W	SP R/W	EPS R/W	PEN R/W	STB R/W	WLS R/W	

Bit(s)	Name	Description
7	DLAB	Divisor Latch Access Bit (DLAB). 0: Select “Normal Mode” registers. 1: Select “Configuration Mode” registers.
6	SB	Set Break. 0: This has no effect on SOUT. 1: The SOUT signal is forced into the “0” state.
5	SP	Stick Parity. 0: Disables stick parity. 1: Parity bit is forced into a defined state. If the EPS bit is 1 and the PEN bit is 1, then the parity bit is 0. If the EPS bit is 0 and the PEN bit is 1, then the parity bit is 1.
4	EPS	Even Parity Select. 0: When EPS = 0 and PEN =1, an odd number of ones is sent and checked. 1: When EPS = 1 and PEN =1, an even number of ones is sent and checked.
3	PEN	Parity Enable. 0: No parity. 1: Enable parity. The EPS and SP bits define even, odd, or stick parity.
2	STB	Number of Stop Bits. 0: One stop bit. 1: Two stop bits. In 5-bit words, 1½ stop bits.
1:0	WLS	Word Length Select. 00: 5 bit. 01: 6 bit. 10: 7 bit. 11: 8 bit.

Modem Control

Read/Write MMIO_base + 0x030010 / MMIO_base + 0x030030

Power-on Default 0x00

7	6	5	4	3	2	1	0
XOFF Status R	IR Enable R/W	Reserved	Loop R/W	OUT2 R/W	OUT1 R/W	RTS R/W	DTR R/W

Bit(s)	Name	Description
7	XOFF Status	This bit is read-only. Enhanced Mode. 0: Normal flow or XON character received. 1: XOFF character received.
6	IR Enable	IrDA Modulation/Demodulatoin Enable. Enhanced Mode. 0: Disable IrDA. 1: Enable IrDA.
5	Reserved	This bit is reserved.
4	Loop	Loopback Mode. 0: Disabled. 1: Enabled. The following conditions are implemented: i. SOUT is forced to “1”. ii. The RX Shift register input is connected to the TX Shift register output. iii. RTS is connected to nCTS, nDTR to nDSR, OUT1 to nRI, and OUT2 to nDCD.
3	OUT2	OUT2 Signal. This bit is reserved.
2	OUT1	OUT1 Signal. This bit is reserved.
1	RTS	Request To Send Signal. 0: The nRTS signal is forced into the “1” state. 1: The nRTS signal is forced into the “0” state.
0	DTR	Data Terminal Ready Signal. This bit is reserved.

Line Status

Read MMIO_base + 0x030014 / MMIO_base + 0x030034

Power-on Default 0x60

7	6	5	4	3	2	1	0
FIFOERR R	TEMT R	THRE R	BI R	FE R	PE R	OE R	DR R

Bit(s)	Name	Description
7	FIFOERR	RX Data Error in FIFO. When FIFOs are disabled, this bit is always “0”. When FIFOs are enabled, this bit is set to “1” when there is at least one PE, FE, or BI in the RX FIFO. It is cleared by a read from the Line Status register provided there are no subsequent errors in the FIFO.
6	TEMT	Transmitter Empty. If the FIFOs are disabled, the bit is set to “1” whenever the TX Holding register and the TX Shift register are empty. If the FIFOs are enabled, this bit is set whenever the TX FIFO and the TX Shift register are empty. In both cases, this bit is cleared when a byte is written to the TX data channel.
5	THRE	TX Holding Register Empty. If the FIFOs are disabled, this bit is set to “1” whenever the TX Holding register is empty and ready to accept new data and it is cleared when the CPU writes to the TX Holding register. If the FIFOs are enabled, this bit is set to “1” whenever the contents of the TX FIFO are reduced to its Trigger Level and it is cleared when at least one byte is written to the TX FIFO.
4	BI	Break Interrupt. RX data hold in “0” state for more than one character transmission time. 0: No break received. 1: Break received in RX Buffer register or current RX FIFO location.
3	FE	Framing Error. No valid stop bit. 0: No framing error received. 1: Framing error received in RX Buffer register or current RX FIFO location.
2	PE	Parity Error. 0: No parity error received. 1: Parity error received in RX Buffer register or current RX FIFO location.
1	OE	Overrun Error. If the FIFOs are disabled, this bit is set to “1” if the RX Buffer was not read by the CPU before new data from the RX Shift register overwrote the previous contents. It is cleared when the CPU reads the Line Status register. If the FIFOs are enabled, an overrun error occurs when the RX FIFO is full and the RX Shift register becomes full. This bit is set to “1” as soon as this happens. The character in the RX Shift register is then overwritten, but is not transferred to the FIFO.
0	DR	Data Ready. This bit is set by the RX Buffer becoming full or by a byte being transferred into the FIFO. It is cleared by the CPU reading the RX Buffer or by reading all of the FIFO bytes. This bit is also cleared whenever the FIFO enable bit is changed.

Modem Status

Read MMIO_base + 0x030018 / MMIO_base + 0x030038

Power-on Default 0x00

7	6	5	4	3	2	1	0
Reserved			CTS R	Reserved			DCTS R/W

Note: Reading the Modem Status Register clears bits [3:0].

Bit(s)	Name	Description
7:5	Reserved	These bits are reserved.
4	CTS	Clear To Send. 0: nCTS signal is in “1” state. 1: nCTS signal is in “0” state.
3:1	Reserved	These bits are reserved.
0	DCTS	Delta Clear To Send. Writing a “1” to bits [3:0] will trigger a Modem Status Interrupt. Writing a “0” will clear the Modem Status Interrupt. 0: No change detected in CTS signal. 1: Change detected in CTS signal since this register was last read.

Scratch

Read/Write MMIO_base + 0x03001C / MMIO_base + 0x03003C

Power-on Default –

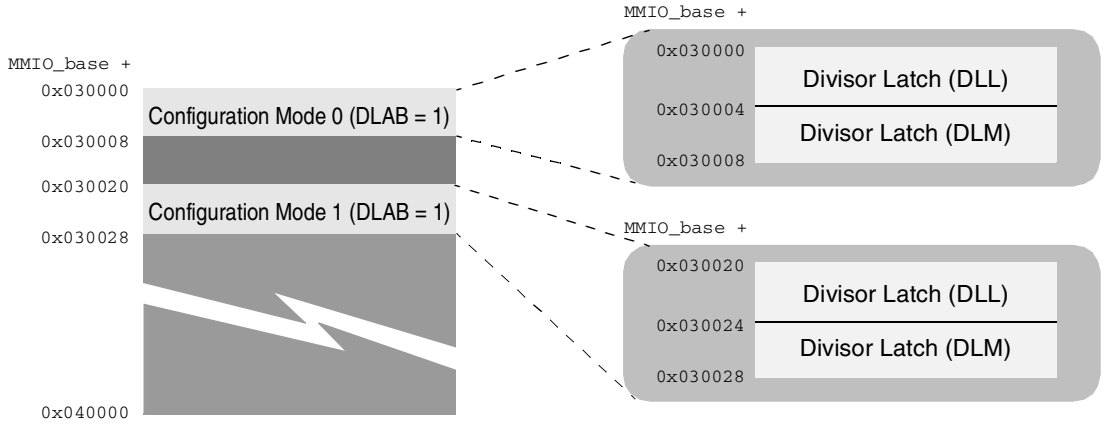
7	6	5	4	3	2	1	0
Scratch							

Bit(s)	Name	Description
7:0	Scratch	Scratch data that can be used by the user.

Configuration Mode Registers (DLAB = 1)

The UART / IrDA is working in "configuration mode" when DLAB (bit [7] of the Line Control register) is "1". The Divisor Latch registers are available as defined in Figure 14-3.

Figure 14-3: Configuration Mode Registers (DLAB = 1)



The incoming clock is divided by the value in the Divisor Latch registers (1 – 65535) to generate the Baud Rate Generator output signal (BAUD). These registers are accessible only when the DLAB bit in the Line Control register is set.

Note: Division by 1 generates a constant-high BAUD signal.

Table 14-4 shows the required divisor to generate a given baud rate with an 8 MHz input clock. The generated clock enable is 16x the required baud rate. Use the following equation to calculate clock frequencies (f_{clock}) not listed here:

$$\text{Divisor value} = f_{clock} / (16 \times \text{desired baud rate})$$

Table 14-4: Baud Rate Divisor

Desired Baud Rate	8 MHz Divisor	Desired Baud Rate	8 MHz Divisor
50	10000	2400	208
75	6667	3600	139
110	4545	4800	104
134.5	3717	7200	69
150	3333	9600	52
300	1667	19200	26
600	833	38400	13
1200	417	56000	9
1800	277	128000	4
2000	250	256000	2

Divisor Latch (DLL)

Read/Write MMIO_base + 0x030000 / MMIO_base + 0x030020

Power-on Default 0x01

7	6	5	4	3	2	1	0
Baud Rate Generator [7:0]							

Divisor Latch (DLM)

Read/Write MMIO_base + 0x030004 / MMIO_base + 0x030024

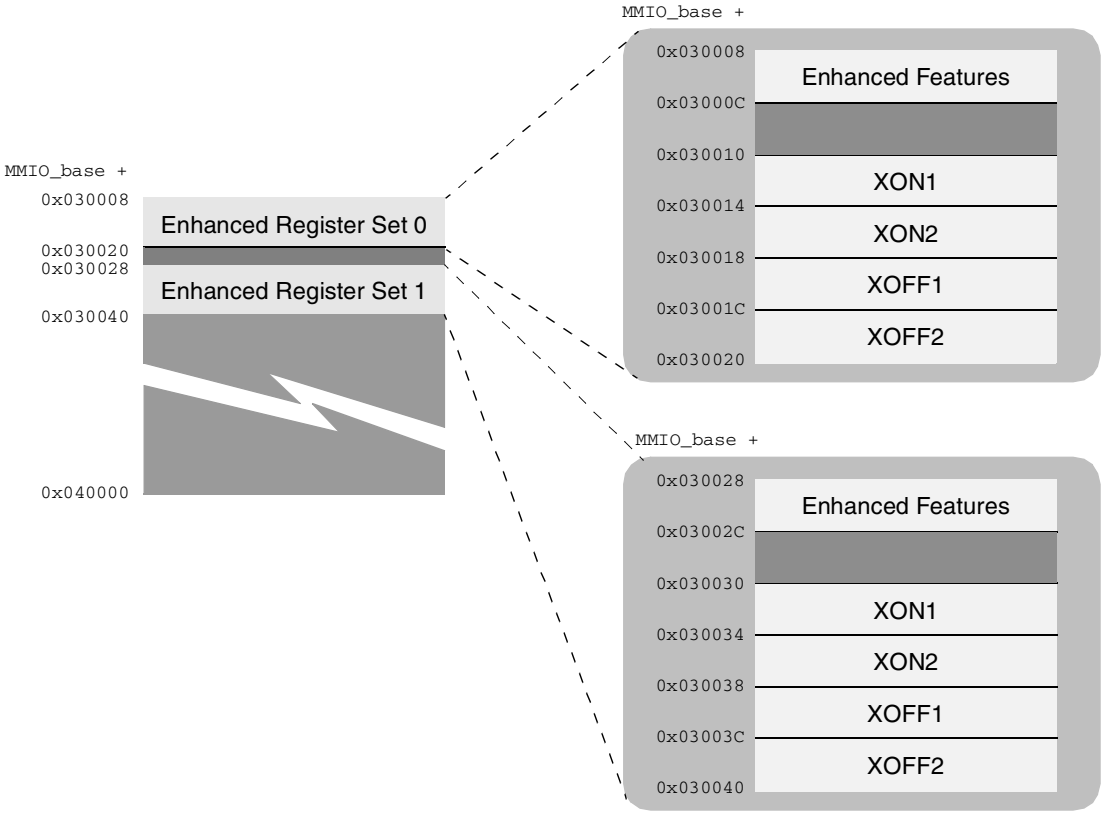
Power-on Default 0x00

7	6	5	4	3	2	1	0
Baud Rate Generator [15:8]							

Enhanced Register Set

The Enhanced Features, XON*n*, and XOFF*n* registers are only available when the value of the Line Control register is `0xBF`.

Figure 14-4: Enhanced Register Set



Enhanced Features

Read/Write MMIO_base + `0x030008` / MMIO_base + `0x030028`

Power-on Default `0x00`

7	6	5	4	3	2	1	0
AutoCTS	AutoRTS	0	EnableE	SWFlowCont[3:0]			

Bit(s)	Name	Description
7	AutoCTS	Setting this bit enables hardware transmission flow control.
6	AutoRTS	Setting this bit enables hardware reception flow control.
5	0	This bit always reads as 0.

Bit(s)	Name	Description
4	EnableE	Enable Enhancements. Setting this bit enables the enhanced register set.
3:0	SWFlowCont[3:0]	Software Flow Control Bits. These are the software flow control bits. Refer to Table 14-1 on page 14-3 for their encoding.

XON1

Read/Write MMIO_base + 0x030010 / MMIO_base + 0x030030

Power-on Default 0x00

7	6	5	4	3	2	1	0
XON1[7:0]							

Bit(s)	Name	Description
7:0	XON1	This register contains the XON1 character used in software flow control.

XON2

Read/Write MMIO_base + 0x030014 / MMIO_base + 0x030034

Power-on Default 0x00

7	6	5	4	3	2	1	0
XON2[7:0]							

Bit(s)	Name	Description
7:0	XON2	This register contains the XON2 character used in software flow control.

XOFF1

Read/Write MMIO_base + 0x030018 / MMIO_base + 0x030038
Power-on Default 0x00

7	6	5	4	3	2	1	0
XOFF1[7:0]							

Bit(s)	Name	Description
7:0	XOFF1	This register contains the XOFF1 character used in software flow control.

XOFF2

Read/Write MMIO_base + 0x03001C / MMIO_base + 0x03003C
Power-on Default 0x00

7	6	5	4	3	2	1	0
XOFF2[7:0]							

Bit(s)	Name	Description
7:0	XOFF2	This register contains the XOFF2 character used in software flow control.

15 PWM Specification

Functional Overview

The Pulse Width Modulation (PWM) module is a simple counter that can pulse with programmable pulse widths. The pulse optionally can be tied to the PWM interrupt signal to generate a periodic interrupt for timing or watchdog support. The input clock is the peripheral clock at 96 MHz. The output pulse starts high.

The following subsections provide some different samples of PWM usage.

Delay Counter with Interrupt

Any of the three available PWM circuits can be programmed to perform a single shot delay. Once the delay is finished, an interrupt is triggered. The required delay is assumed to be 20 ms.

The values for the PWM *n* register are calculated as follows:

- Delay * clock = 20 ms * 96 MHz = 1,920,000
- A 50% duty cycle means 960,000 clocks are LOW and 960,000 clocks are HIGH
- The shift to a 12-bit value is done by dividing by 2^8: (3,750 – 1) clocks LOW and (3,750 – 1) clocks HIGH
- The value for the PWM *n* register is: `0xEA5EA585`

Internal Timer with Interrupt

Any of the three available PWM circuits can be programmed to act as a periodic timer to support a clock. The periodic timer generates an interrupt after each cycle. The required periodic interval is assumed to be 1 s. For this example, there is a 30/70% duty cycle.

The values for the PWM *n* register are calculated as follows:

- Delay * clock = 1 s * 96 MHz = 96,000,000
- A 30% duty cycle means 28,800,000 clocks are LOW and 67,200,000 clocks are HIGH
- The shift to a 12-bit value is done by dividing by 2^{15}: (879 – 1) clocks LOW and (2,051 – 1) clocks HIGH
- The value for the PWM *n* register is: `0x80236EF5`

External Pulse

In this example, the PWM is programmed for an external pulse with a frequency of 44.1 kHz and a duty cycle of 15%.

The values for the PWM *n* register are calculated as follows:

- Delay * clock = (1 / 44.1 kHz) * 96 MHz = 2,177
- A 15% duty cycle means (327 – 1) clocks are LOW and (1,850 – 1) clocks are HIGH
- The value for the PWM *n* register is: `0x73914601`

Register Descriptions

The PWM registers are shown in Table 15-1.

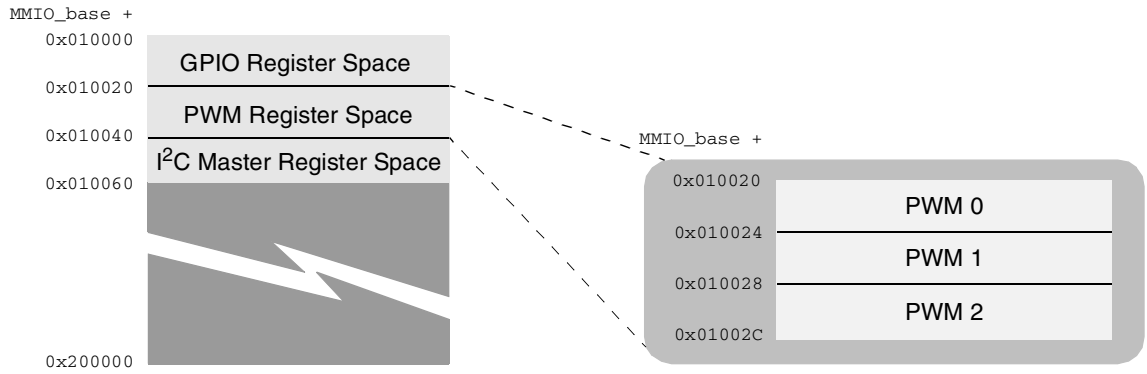
Table 15-1: PWM Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value	Register Name
0x010020	R/W	32	0x00000000	PWM 0
0x010024	R/W	32	0x00000000	PWM 1
0x010028	R/W	32	0x00000000	PWM 2

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.

The PWM registers control the three PWM pins. It contains three registers, one for each PWM pin. Figure 15-1 defines the register layout for the PWM registers.

Figure 15-1: PWM Register Space



PWM 0

Read/Write Address 0x010020

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
High Counter R/W												Low Counter R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Low Counter R/W								Clock Divide R/W				IP R/W	I R/W	Res	E R/W

Bit(s)	Name	Description
31:20	High Counter	Number of clocks - 1 the PWM should remain high before switching to low.
19:8	Low Counter	Number of clocks - 1 the PWM should remain low before switching to high.
7:4	Clock Divide	Select divisor for 96MHz input clock.
3	IP	PWM Interrupt Pending. In order to clear a pending interrupt, write a “1” in the IP bit. 0: No interrupt pending. 1: Interrupt pending.
2	I	Enable or Disable PWM Interrupt. 0: Disable PWM interrupt. 1: Enable PWM interrupt whenever a single cycle is completed.
1	Res	This bit is reserved.
0	E	Enable or Disable the PWM. 0: Disabled. 1: Enabled.

0000	÷ 1	96MHz	1000	÷ 256	375kHz
0001	÷ 2	48MHz	1001	÷ 512	187.5kHz
0010	÷ 4	24MHz	1010	÷ 1,024	93.75kHz
0011	÷ 8	12MHz	1011	÷ 2,048	46.875kHz
0100	÷ 16	6MHz	1100	÷ 4,096	23.438kHz
0101	÷ 32	3MHz	1101	÷ 8,192	11.719kHz
0110	÷ 64	1.5MHz	1110	÷ 16,384	5.859kHz
0111	÷ 128	750kHz	1111	÷ 32,768	2.93kHz

PWM 1

Read/Write Address 0x010024

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
High Counter R/W												Low Counter R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Low Counter R/W								Clock Divide R/W				IP	I	Res	E

Bit(s)	Name	Description
31:20	High Counter	Number of clocks - 1 the PWM should remain high before switching to low.
19:8	Low Counter	Number of clocks - 1 the PWM should remain low before switching to high.
7:4	Clock Divide	Select divisor for 96MHz input clock.
3	IP	PWM Interrupt Pending. In order to clear a pending interrupt, write a “1” in the IP bit. 0: No interrupt pending. 1: Interrupt pending.
2	I	Enable or Disable PWM Interrupt. 0: Disable PWM interrupt. 1: Enable PWM interrupt whenever a single cycle is completed.
1	Res	This bit is reserved.
0	E	Enable or Disable the PWM. 0: Disabled. 1: Enabled.

0000	÷ 1	96MHz	1000	÷ 256	375kHz
0001	÷ 2	48MHz	1001	÷ 512	187.5kHz
0010	÷ 4	24MHz	1010	÷ 1,024	93.75kHz
0011	÷ 8	12MHz	1011	÷ 2,048	46.875kHz
0100	÷ 16	6MHz	1100	÷ 4,096	23.438kHz
0101	÷ 32	3MHz	1101	÷ 8,192	11.719kHz
0110	÷ 64	1.5MHz	1110	÷ 16,384	5.859kHz
0111	÷ 128	750kHz	1111	÷ 32,768	2.93kHz

PWM 2

Read/Write Address 0x010028
Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
High Counter R/W												Low Counter R/W			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Low Counter R/W								Clock Divide R/W				IP R/W	I R/W	Res	E R/W

Bit(s)	Name	Description
31:20	High Counter	Number of clocks - 1 the PWM should remain high before switching to low.
19:8	Low Counter	Number of clocks - 1 the PWM should remain low before switching to high.
7:4	Clock Divide	Select divisor for 96MHz input clock.
3	IP	PWM Interrupt Pending. In order to clear a pending interrupt, write a “1” in the IP bit. 0: No interrupt pending. 1: Interrupt pending.
2	I	Enable or Disable PWM Interrupt. 0: Disable PWM interrupt. 1: Enable PWM interrupt whenever a single cycle is completed.
1	Res	This bit is reserved.
0	E	Enable or Disable the PWM. 0: Disabled. 1: Enabled.

0000	÷ 1	96MHz	1000	÷ 256	375kHz
0001	÷ 2	48MHz	1001	÷ 512	187.5kHz
0010	÷ 4	24MHz	1010	÷ 1,024	93.75kHz
0011	÷ 8	12MHz	1011	÷ 2,048	46.875kHz
0100	÷ 16	6MHz	1100	÷ 4,096	23.438kHz
0101	÷ 32	3MHz	1101	÷ 8,192	11.719kHz
0110	÷ 64	1.5MHz	1110	÷ 16,384	5.859kHz
0111	÷ 128	750kHz	1111	÷ 32,768	2.93kHz

16 Synchronous Serial Port

Functional Overview

The Synchronous Serial Port (SSP) consists of two SPI-like interfaces that support devices such as CAN controllers. Its main functional blocks are described in the following subsections.

Register Block

The register block stores data written or to be read across the SM501 interface.

Clock Prescaler

When configured as a master, an internal prescaler, comprising two free-running reloadable serially linked counters, is used to provide the serial output clock SCLKOUT.

You can program the clock prescaler, through the Clock Prescale register, to divide SSPCLK by a factor of 2 to 254 in steps of two. By not using the least significant bit of the Clock Prescale register, division by an odd number is not possible, thus ensuring generation of a symmetrical (equal mark space ratio) clock.

The output of the prescaler is further divided by a factor of 1 to 256, through the programming of the SSP Control 0 register, to give the final master output clock SCLKOUT.

Transmit FIFO

The common transmit FIFO is a 16-bit wide, eight lines deep, first-in, first-out memory buffer. CPU data written across the SM501 interface are stored in the buffer until read out by the transmit logic.

When configured as a master or a slave parallel data is written into the transmit FIFO prior to serial conversion and transmission to the attached slave or master respectively, through the SSPTXD pin.

Receive FIFO

The common receive FIFO is a 16-bit wide, 8-locations deep, first-in, first-out memory buffer. Received data from the serial interface are stored in the buffer until read out by the CPU.

When configured as a master or slave, serial data received through the SSPRXD pin is registered prior to parallel loading into the attached slave or master receive FIFO, respectively.

Transmit/Receive Logic

When configured as a master, the clock to the attached slaves is derived from a divided down version of SSPCLK through the prescaler operations described previously. The master transmit logic successively reads a value from its transmit FIFO and performs parallel to serial conversion on it. Then the serial data stream and frame control signal, synchronized to SCLKOUT, are output through the SSPTXD pin to the attached slaves. The master receive logic performs serial to parallel conversion on the incoming synchronous SSPRXD data stream, extracting and storing values into its receive FIFO, for subsequent reading through the SM501 interface.

When configured as a slave, the SSPCLKIN clock is provided by an attached master and used to time its transmission and reception sequences. The slave transmit logic, under control of the master clock, successively reads a value from its transmit FIFO, performs parallel to serial conversion, then output the serial data stream and frame control signal through the slave SSPTXD pin. The slave receive logic performs serial to parallel conversion on the incoming SSPRXD data stream, extracting and storing values into its receive FIFO, for subsequent reading through the SM501 interface.

Interrupt Generation Logic

The SSP generates three individual maskable, active HIGH interrupts. It also generates a combined interrupt output that is the OR function of the three individual interrupt requests.

The single combined interrupt can be used with a system interrupt controller that provides another level of masking on a per-peripheral basis. This option allows use of modular device drivers that always know where to find the interrupt source control register bits.

The individual interrupt requests also can be used with a system interrupt controller that provides masking for the outputs of each peripheral. In this way, a global interrupt controller service routine would be able to read the entire set of sources from one wide register in the system interrupt controller. This is useful when the time to read from the peripheral registers is significant compared to the CPU clock speed in a real-time system.

The SSP supports both the above methods because the overhead is small.

The transmit and receive dynamic data-flow interrupts, SSPTXINTR and SSPRXINTR, are separated from the status interrupts so that data can be read or written in response to the FIFO trigger levels.

Synchronizing Registers and Logic

The SSP supports both asynchronous and synchronous operation of the clocks, PCLK and SSPCLK. The SSP implements synchronization registers and handshaking logic, which are active at all times. This implementation has a minimal impact on performance and area. The SSP synchronizes control signals in both directions of data flow—from the PCLK to the SSPCLK domain and from the SSPCLK to the PCLK domain.

Operation

The operation of the SSP is described in the following subsections.

Configuring the SSP

Following reset, the SSP logic is disabled and should be configured when in this state.

The SSP Control 0 and Control 1 registers configure the peripheral as a master or slave operating under one of the following protocols:

- Motorola SPI
- Texas Instruments SSI
- National Semiconductor

The bit rate, derived from the external SSPCLK, requires the programming of the Clock Prescale register.

Enable SSP Operation

You can either prime the transmit FIFO, by writing up to eight 16-bit values when the SSP is disabled, or allow the transmit FIFO service request to interrupt the CPU. Once enabled, transmission or reception of data begins on the transmit (SSPTXD) and receive (SSPRXD) pins.

Clock Signals

The frequency selected for SSPCLK must accommodate the desired range of bit clock rates.

$F_{\mathbf{SSPCLK}}(\text{min}) >= 2 \times F_{\mathbf{SCLKOUT}}(\text{max})$

$F_{\mathbf{SSPCLK}}(\text{max}) <= 254 \times 256 \times F_{\mathbf{SCLKOUT}}(\text{min})$

For example, for a range of bit clocks from 7.2 kHz to 1.8432 MHz, the SSPCLK frequency must be within the range 3.6864 MHz to 468 MHz.

The frequency of SSPCLK must be within the required error limits for all baud rates to be used.

There is also a constraint on the ratio of clock frequencies for PCLK to SSPCLK. The frequency of SSPCLK must be less than or equal to the frequency of PCLK.

$F_{\mathbf{SSPCLK}} <= F_{\mathbf{PCLK}}$

Programming the SSP Control 0 Register

See “Register Descriptions” on page 14 for more details on the bit assignment of this register.

The SSP Control 0 register is used to:

- Program the serial clock rate
- Select one of the three protocols
- Select the data word size (where applicable)

The Serial Clock Rate value, in conjunction with the clock prescale divisor value in the Clock Prescale register, is used to derive the SSP transmit and receive bit rate from the external SSPCLK.

The frame format is programmed through the Format bits and the data word size through the DataSize bits.

Bit phase and polarity, applicable to Motorola SPI format only, are programmed through the Ph and Pol bits.

Programming the SSP Control 1 Register

See “Register Descriptions” on page 14, for more details on the bit assignment of this register.

The SSP Control 1 register is used to:

- Select master or slave mode
- Control interrupt enabling or disabling
- Enable a loopback test feature
- Enable the SSP peripheral

To configure the SSP as a master, clear the SSP Control 1 register master or slave selection bit (M) to 0, which is the default value on reset.

Setting the SSP Control 1 register M bit to 1 configures the SSP as a slave. When configured as a slave, enabling or disabling of the SSP SSPTXD signal is provided through the slave mode output select bit (O). This can be used in some multi-slave environments where masters might parallel broadcast.

Transmit and receive FIFO level interrupts can be enabled by setting the respective Transmit Interrupt Enable (TI), Receive Interrupt Enable (RI), and Overflow Interrupt Enable (OI) bits within the SSP Control 1 register.

To enable the operation of the SSP, set the Synchronous Serial Port Enable (E) bit to 1.

Bit Rate Generation

The serial bit rate is derived by dividing down the input clock SSPCLK. The clock is first divided by an even prescale value CPSDVSR from 2 to 254, which is programmed in the Clock Prescale register. The clock is further divided by a value from 1 to 256, which is 1 + SCR, where SCR is the serial clock rate value programmed in the SSP Control 0 register.

The frequency of the output signal bit clock SCLKOUT is defined below:

$$F_{SCLKOUT} = \frac{F_{SSPCLK}}{CPSDVR \times (1+SCR)}$$

For example, if SSPCLK is 3.6864 MHz, and CPSDVSR = 2, then SCLKOUT has a frequency range from 7.2 kHz to 1.8432 MHz.

Frame Formats

Each data frame is between 4 and 16 bits long depending on the size of data programmed, and is transmitted starting with the MSB. There are three basic frame types that can be selected:

- Texas Instruments synchronous serial
- Motorola SPI
- National Semiconductor Microwire

For all three formats, the serial clock (SCLKOUT) is held inactive while the SSP is idle, and transitions at the programmed frequency only during active transmission or reception of data. The idle state of SCLKOUT is utilized to provide a receive timeout indication that occurs when the receive FIFO still contains data after a timeout period.

For Motorola SPI and National Semiconductor Microwire frame formats, the serial frame (SFRMOUT) pin is active LOW, and is asserted (pulled down) during the entire transmission of the frame.

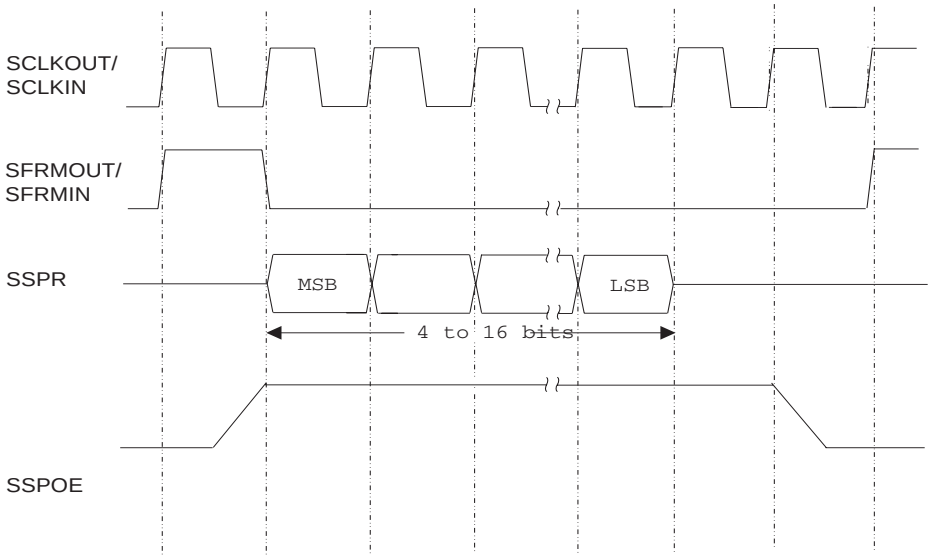
For the Texas Instruments synchronous serial frame format, the SFRMOUT pin is pulsed for one serial clock period starting at its rising edge, prior to the transmission of each frame. For this frame format, both the SSP and the off-chip slave device drive their output data on the rising edge of SCLKOUT, and latch data from the other device on the falling edge.

Unlike the full-duplex transmission of the other two frame formats, the National Semiconductor Microwire format uses a special master-slave messaging technique, which operates at half-duplex. In this mode, when a frame begins, an 8-bit control message is transmitted to the off-chip slave. During this transmit, no incoming data is received by the SSP. After the message has been sent, the off-chip slave decodes it and, after waiting one serial clock after the last bit of the 8-bit control message has been sent, responds with the requested data. The returned data can be 4 to 16 bits in length, making the total frame length anywhere from 13 to 25 bits.

Texas Instruments Synchronous Serial Frame Format

Figure 16-1 shows the Texas Instruments synchronous serial frame format for a single transmitted frame.

Figure 16-1: Texas Instruments Synchronous Serial Frame Format (Single Transfer)

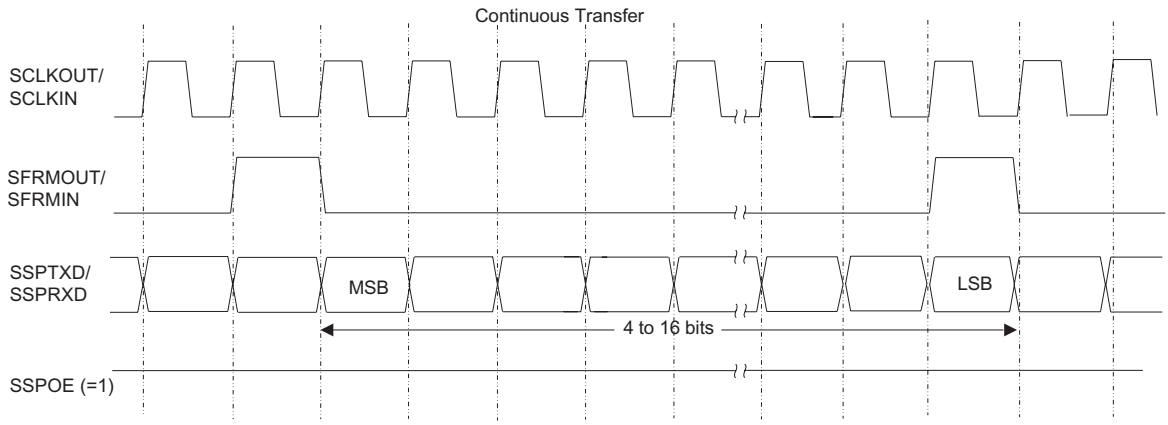


In this mode, SCLKOUT and SFRMOUT are forced LOW, and the transmit data line SSPTXD is 3-stated whenever the SSP is idle. Once the bottom entry of the transmit FIFO contains data, SFRMOUT is pulsed HIGH for one SCLKOUT period. The value to be transmitted is also transferred from the transmit FIFO to the serial shift register of the transmit logic. On the next rising edge of SCLKOUT, the MSB of the 4-bit to 16-bit data frame is shifted out on the SSPTXD pin. Likewise, the MSB of the received data is shifted onto the SSPRXD pin by the off-chip serial slave device.

Both the SSP and the off-chip serial slave device then clock each data bit into their serial shifter on the falling edge of each SCLKOUT. The received data is transferred from the serial shifter to the receive FIFO on the first rising edge of SCLKOUT after the LSB has been latched.

Figure 16-2 shows the Texas Instruments synchronous serial frame format when back-to-back frames are transmitted.

Figure 16-2: TI Synchronous Serial Frame Format (Continuous Transfer)



Motorola SPI Frame Format

Figure 16-3 shows the Motorola SPI frame format for a single frame. Figure 16-4 shows the same format when back to back frames are transmitted.

In this mode, SCLKOUT is forced LOW, SFRMOUT is forced HIGH, and the transmit data line SSPTXD is 3-stated whenever the SSP is idle. Once the bottom entry of the transmit FIFO contains data, SFRMOUT is pulsed LOW and remains LOW for the duration of the frame transmission. The falling edge of SFRMOUT causes the value for transmission to be transferred from the bottom transmit FIFO entry to the serial shift register of the transmit logic. The MSB of the 4-bit to 16-bit data frame is then shifted out on the SSPTXD pin half an SCLKOUT period later. The SCLKOUT pin does not transition at this point.

The MSB of the received data is shifted onto the SSPRXD pin by the off-chip slave device as soon as the serial framing signal goes LOW. Both the SSP and the off-chip serial slave device then latch each data bit into their serial shifter on the rising edge of each SCLKOUT. At the end of the frame, the SFRMOUT pin is pulled HIGH one SCLKOUT period after the last bit has been latched in the receive serial shifter. This causes the data to be transferred to the receive FIFO.

Note: The off-chip slave device can 3-state the receive line either on the falling edge of SCLKOUT after the receive shifter has latched the LSB, or when the SFRMOUT pin goes HIGH.

Figure 16-3: Motorola SPI Frame Format (Single Transfer)

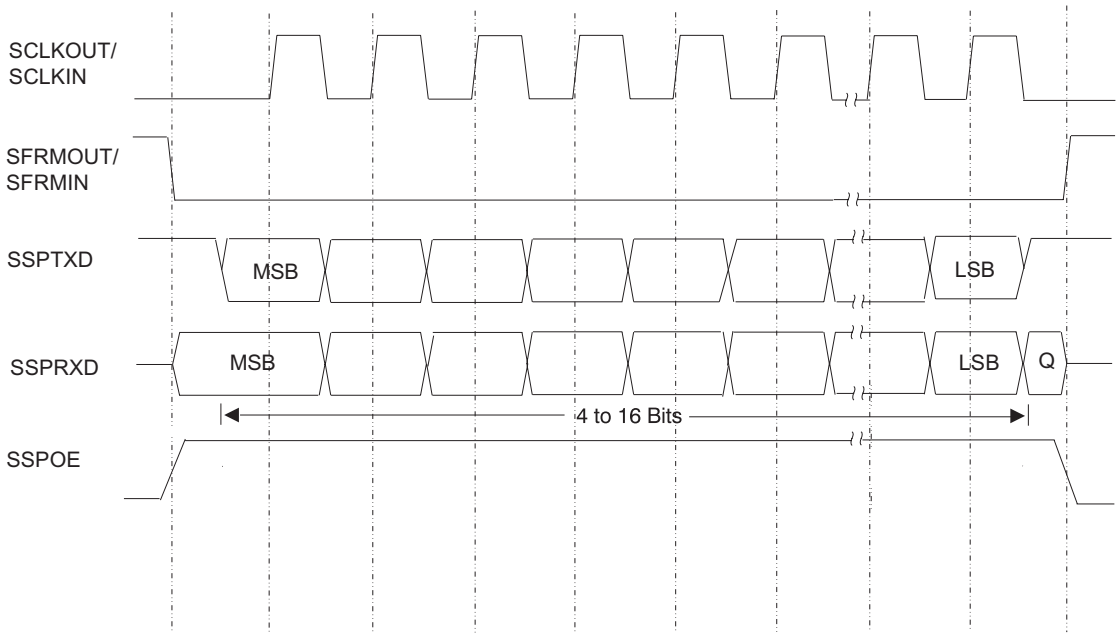
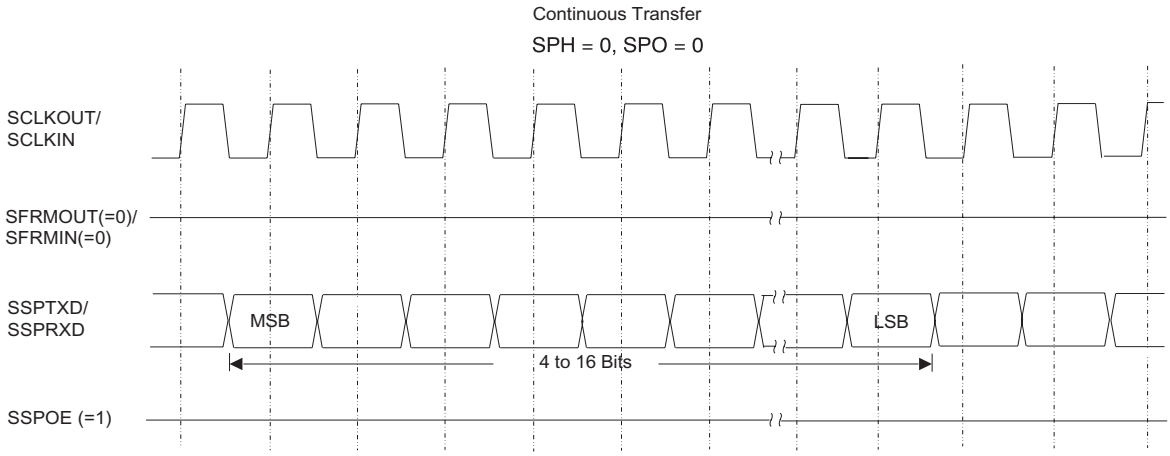


Figure 16-4 shows the Motorola SPI frame format when back to back frames are transmitted.

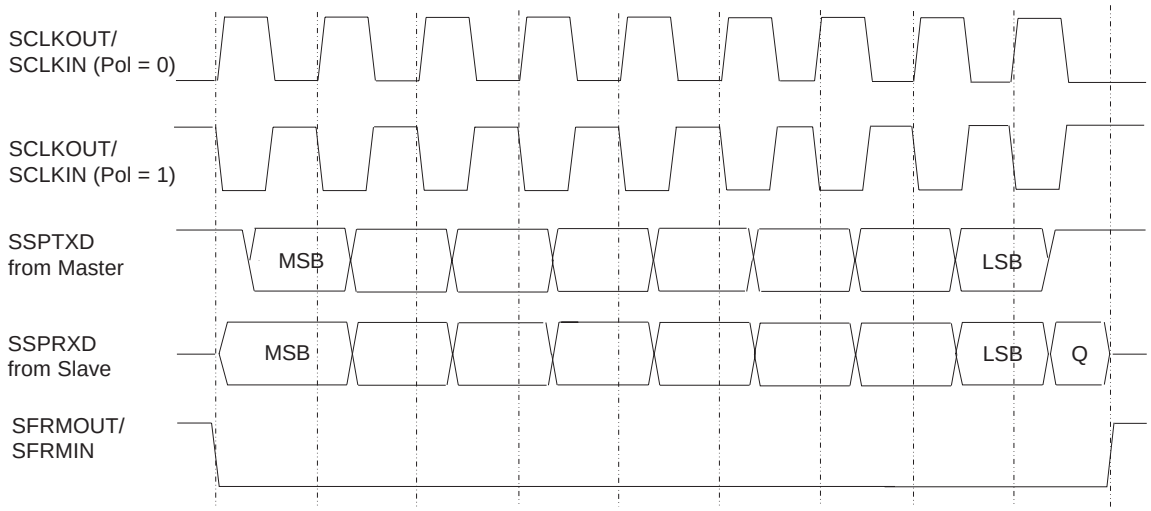
Figure 16-4: Motorola SPI Frame Format (Continuous Transfer) SPO = 0, SPH = 0



It is possible to change the Motorola clock phase and polarity by selecting the appropriate value of the Pol and Ph bits in the SSP Control 0 register.

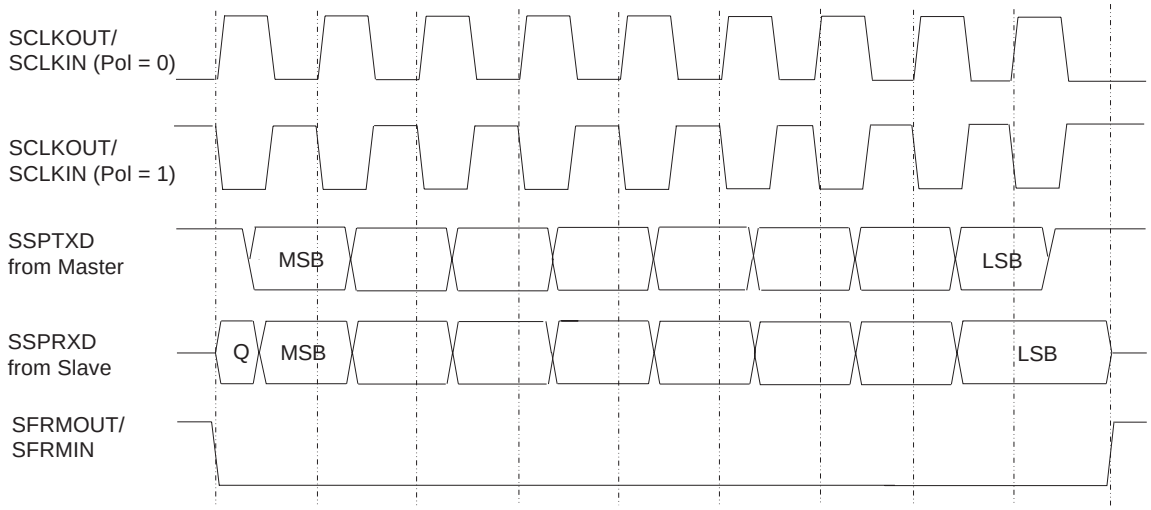
Figure 16-5 and Figure 16-6 show the operation of Pol and Ph.

Figure 16-5: Motorola SPI Frame Format with Ph = 0



Note: In Figure 16-5, Q is an undefined signal.

Figure 16-6: Motorola SPI Frame Format with Ph = 1



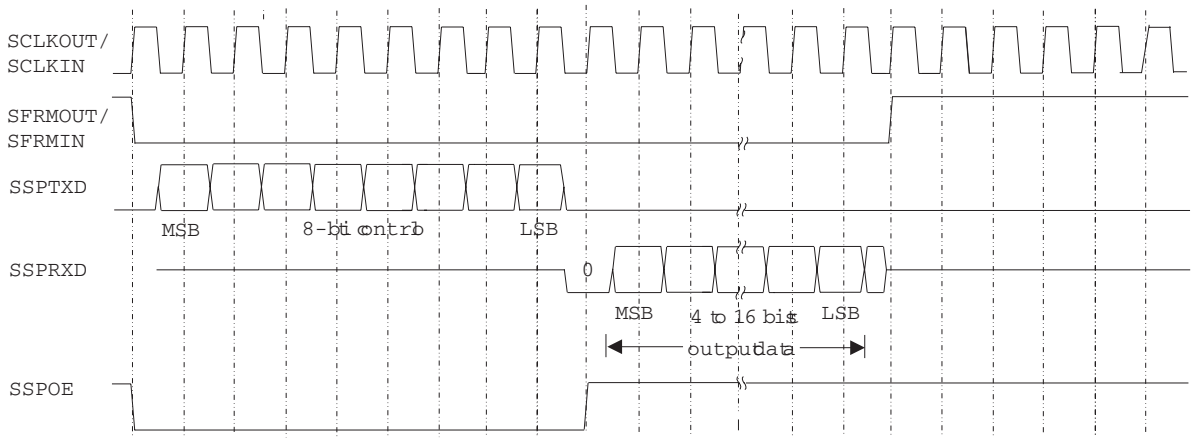
Note: In Figure 16-6, Q is an undefined signal.

For continuous transfers, data transmission begins and ends in the same manner as a single transfer. However, the SFRMOUT line is continuously asserted (held LOW). The transmission of data also occurs back to back (the MSB of the next frame follows directly after the LSB of the current frame.) Each of the received data values is transferred from the receive shifter to the receive FIFO on the falling edge of SCLKOUT, after the LSB of the frame has been latched into the SSP.

National Semiconductor Microwire Frame Format

Figure 16-7 shows the National Semiconductor Microwire frame format, again for a single frame. Figure 16-8 shows the same format when back to back frames are transmitted.

Figure 16-7: Microwire Frame Format, Single Transfer



Microwire format is very similar to SPI format, except that transmission is half-duplex instead of full-duplex, using a master-slave message passing technique. Each serial transmission begins with an 8-bit control word that is transmitted from the SSP to the off-chip slave device. During this transmission, no incoming data is received by the SSP. After the message has been sent, the off-chip slave decodes it and, after waiting one serial clock after the last bit of the 8-bit control message has been sent, responds with the required data. The returned data is 4 to 16 bits in length, making the total frame length anywhere from 13 to 25 bits.

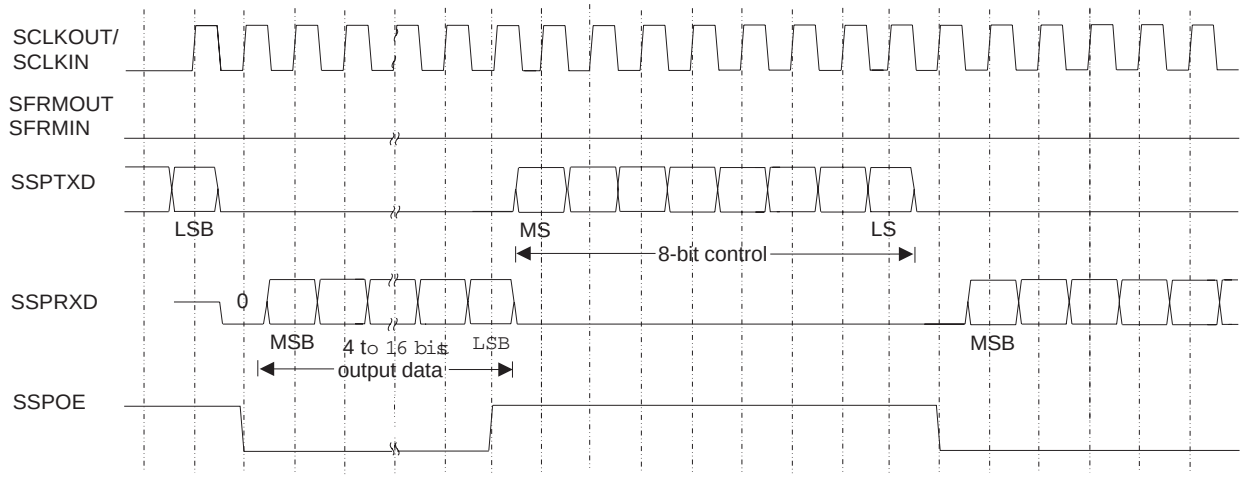
Like SPI mode, SCLKOUT is forced LOW, SFRMOUT is forced HIGH, and the transmit data line SSPTXD is 3-stated whenever the SPMSS is idle. A transmission is triggered by writing a control byte to the transmit FIFO. The falling edge of SFRMOUT causes the value contained in the bottom entry of the transmit FIFO to be transferred to the serial shift register of the transmit logic, and the MSB of the 8-bit control frame to be shifted out onto the SSPTXD pin. SFRMOUT remains LOW for the duration of the frame transmission. The SSPRXD pin remains 3-stated during this transmission.

The off-chip serial slave device latches each control bit into its serial shifter on the rising edge of each SCLKOUT. After the last bit is latched by the slave device, the control byte is decoded during a one clock wait state, and the slave responds by transmitting data back to the SSP. Each bit is driven onto SSPRXD on the falling edge of SCLKOUT. The SSP in turn latches each bit on the rising edge of SCLKOUT. At the end of the frame, for single transfers, SFRMOUT is driven HIGH one clock period after the last bit has been latched in the receive serial shifter, causing the data to be transferred to the receive FIFO.

Note: The off-chip slave device can 3-state the receive line either on the falling edge of SCLKOUT after the LSB has been latched by the receive shifter, or when SFRMOUT goes HIGH.

For continuous transfers, data transmission begins and ends in the same manner as a single transfer. However, the SFRMOUT line is continuously asserted (held LOW) and transmission of data occurs back to back. The control byte of the next frame follows directly after the LSB of the received data from the current frame. Each of the received values is transferred from the receive shifter on the falling edge SCLKOUT, after the LSB of the frame has been latched into the SSP.

Figure 16-8: Microwire Frame Format, Continuous Transfers

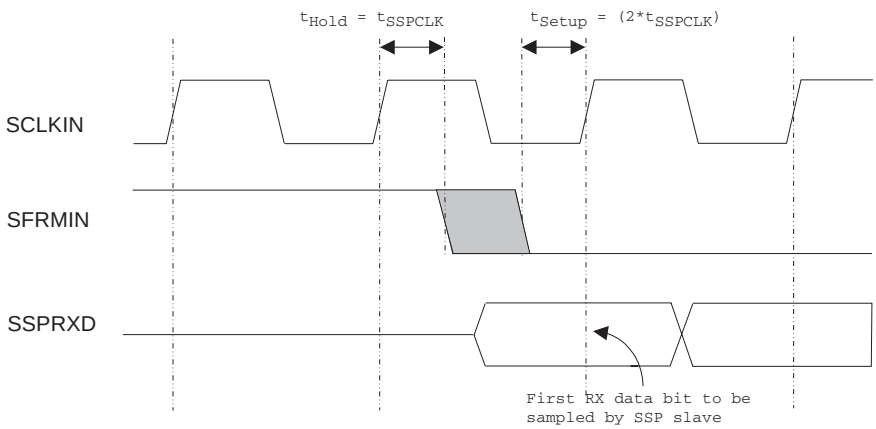


Setup and Hold Time Requirements—SFRMIN with Respect to SSPCLKIN, Microwire Mode

In Microwire mode, the SSP slave samples the first bit of receive data on the rising edge of SCLKIN after SFRMIN has gone LOW. Masters that drive a free-running SCKLIN must ensure that the SFRMIN signal has sufficient setup and hold margins with respect to the rising edge of SCLKIN.

Figure 16-9 illustrates these setup and hold time requirements. With respect to the SCLKIN rising edge on which the first bit of receive data is to be sampled by the SSP slave, SFRMIN must have a setup of at least two times the period of SSPCLK on which the SSP operates. With respect to the SCLKIN rising edge previous to this edge, SFRMIN must have a hold of at least one SSPCLK period.

Figure 16-9: Microwire Frame Format, SFRMIN Input Setup and Hold Requirements



Examples of Master and Slave Configurations

Figure 16-10 through Figure 16-12 show how to connect the SSP peripheral to other synchronous serial peripherals, when it is configured as a master or slave.

Note: The SSP does not support dynamic switching between master and slave in a system. Each instance is configured and connected either as a master or slave.

Figure 16-10 shows the SSP instanced three times, as a single master and two slaves. The master can broadcast to the two slaves through the master SSPTXD line. In response, only one slave drives its nSSPOE signal HIGH, thereby enabling its SSPTXD data onto the SSPRXD line of the master.

Figure 16-10: SSP Master Coupled to Two Slaves

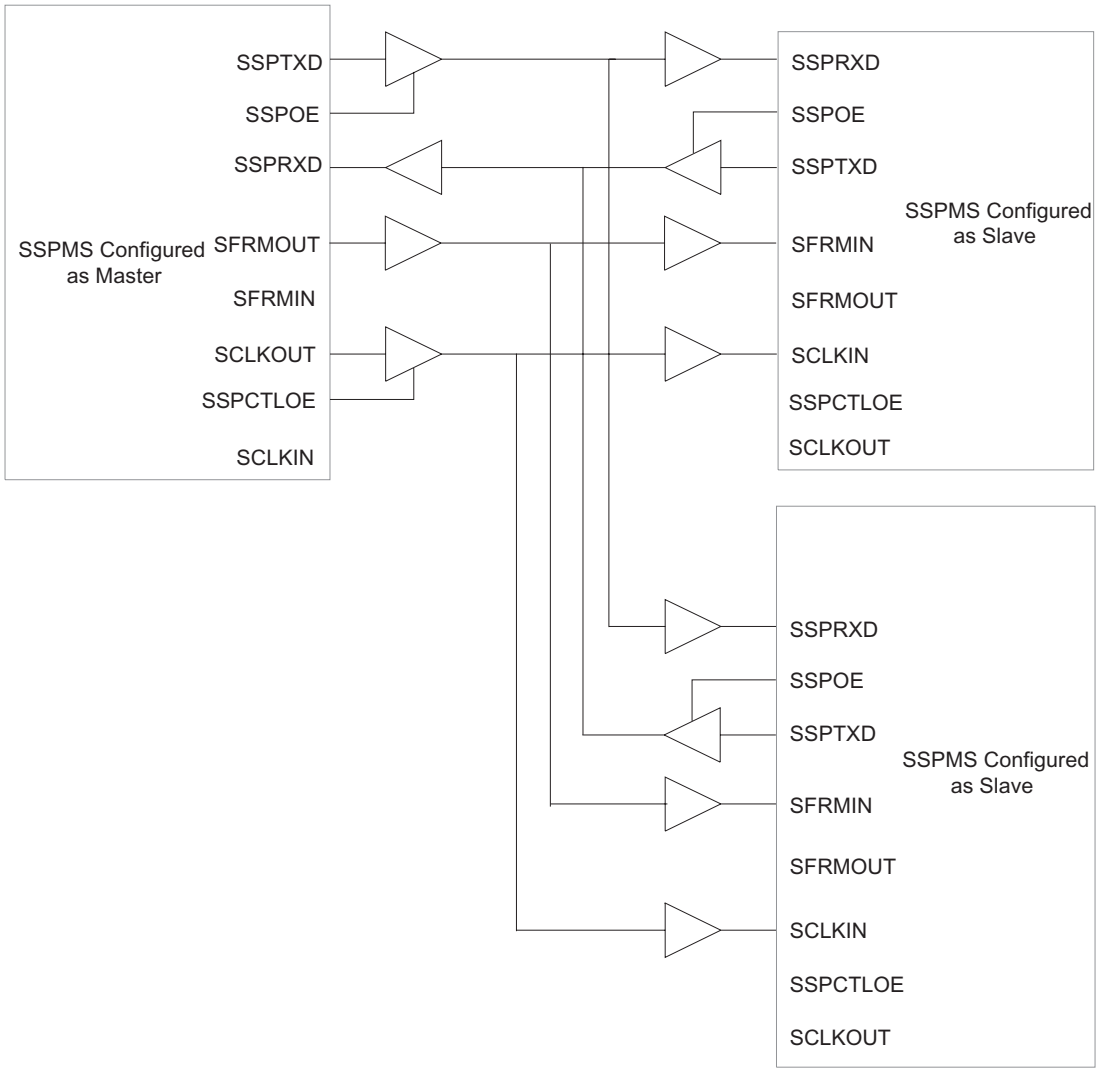


Figure 16-11 shows how an SSP, configured as master, interfaces to two Motorola SPI slaves. Each SPI Slave Select ($\overline{SS}$) signal is permanently tied LOW and configures them as slaves. Similar to the above operation, the master can broadcast to the two slaves through the master SSP SSPTXD line. In response, only one slave drives its SPI MISO port onto the master's SSPRXD line.

Figure 16-11: SSPMS (PL021) Master Coupled to Two SPI Slaves

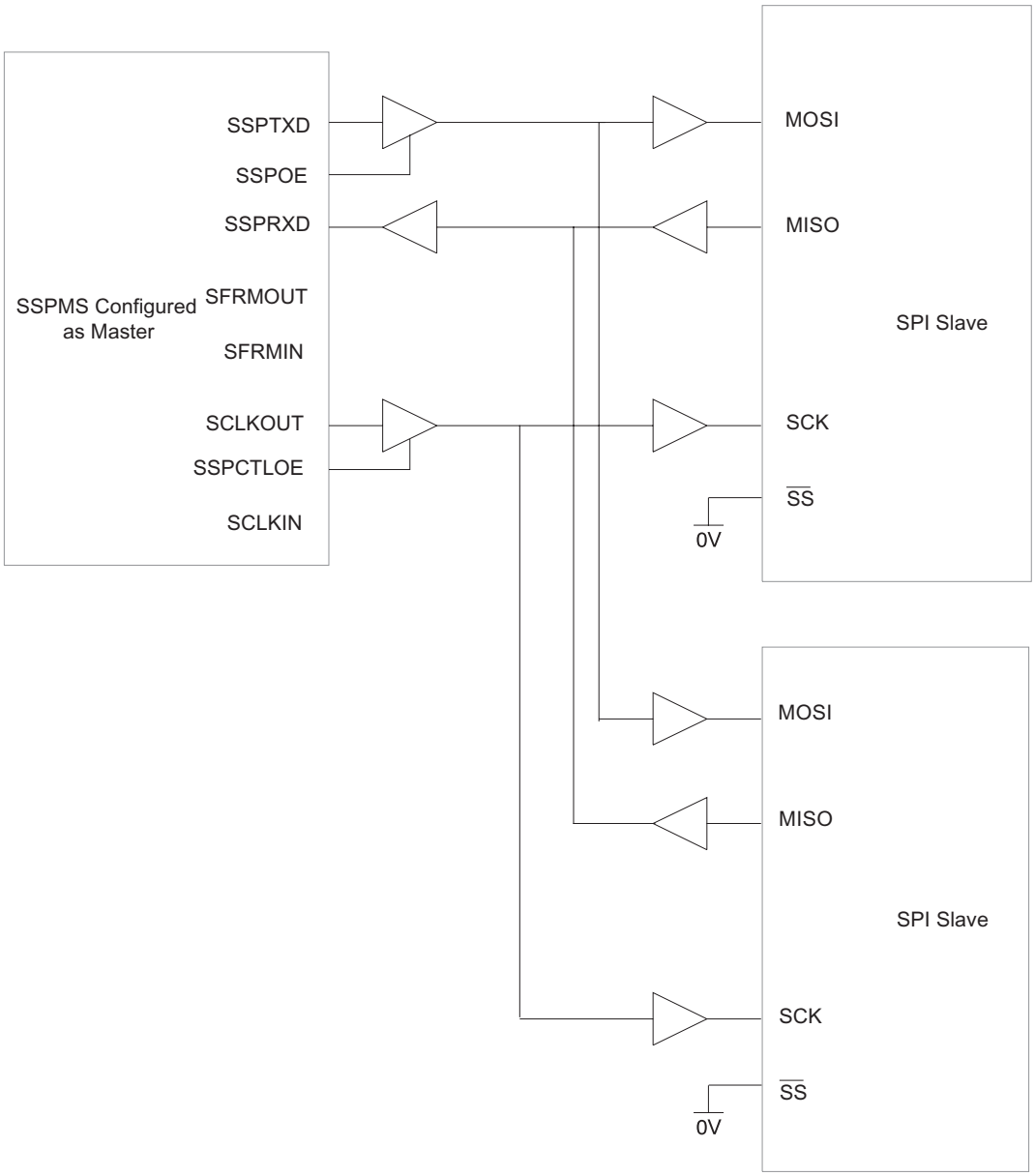
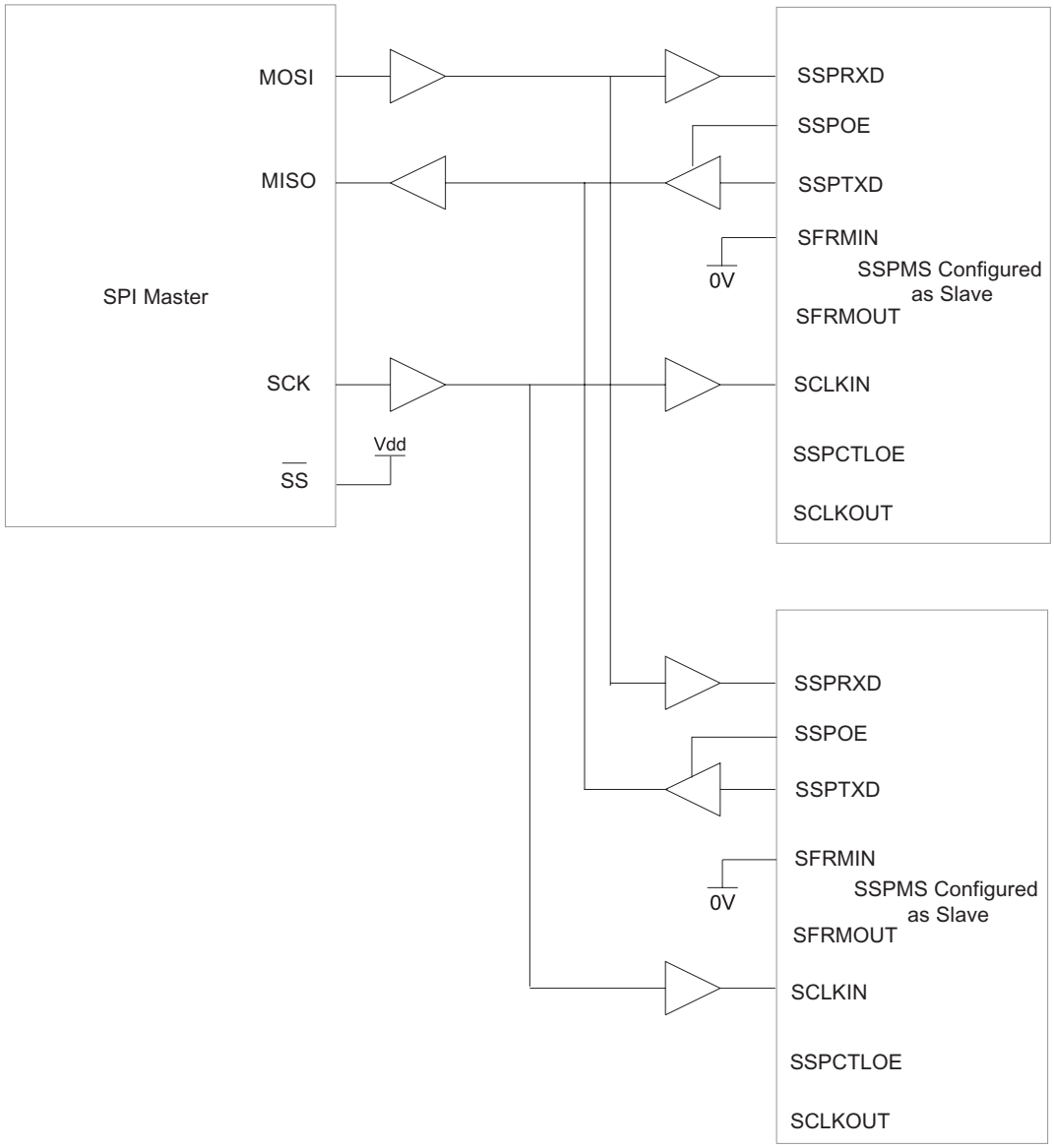


Figure 16-12 shows a Motorola SPI configured as a master and interfaced to two instances of SSP configured as slaves. In this case, the slave Select Signal ($\overline{SS}$) signal is permanently tied HIGH and configures it as a master. It is possible for the master to broadcast to the two slaves through the master SPI MOSI line and in response, only one slave will drive its SSPOE signal HIGH, thereby enabling in SSPTXD data onto the master's MISO line.

Figure 16-12: SPI Master Coupled to Two SSP Slaves



Register Descriptions

The base address of the SSP is not fixed, and might be different for any particular system implementation. The offset of any particular register from the base address, however, is fixed.

Locations at offsets `0x018-0xFF` and `0x94-0xFF` are reserved.

Table 16-1 shows the SSP registers.

Table 16-1: SSP Register Summary

Offset from MMIO_base[1]	Type	Width	Reset Value[2]	Register Name
0x020*n*00	R/W	32	0x00000000	Control 0
0x020*n*04	R/W	32	0x00000000	Control 1
0x020*n*08	R/W	32	0x00000000	Data
0x020*n*0C	R	32	0x00000003	Status
0x020*n*10	R/W	32	0x00000000	Clock Prescale
0x020*n*14	R/W	32	0b0000.0000.0000.0000. 0000.0000.0000.00XX	Interrupt Status

1. Refer to Table 1-5 on page 28 for MMIO_base values depending on the CPU.
2. In the reset values, “X” indicates don’t care.

Figure 16-13 shows how this 64kB region in the MMIO space is laid out. It controls the SSP registers.

Figure 16-13: SSP Register Space

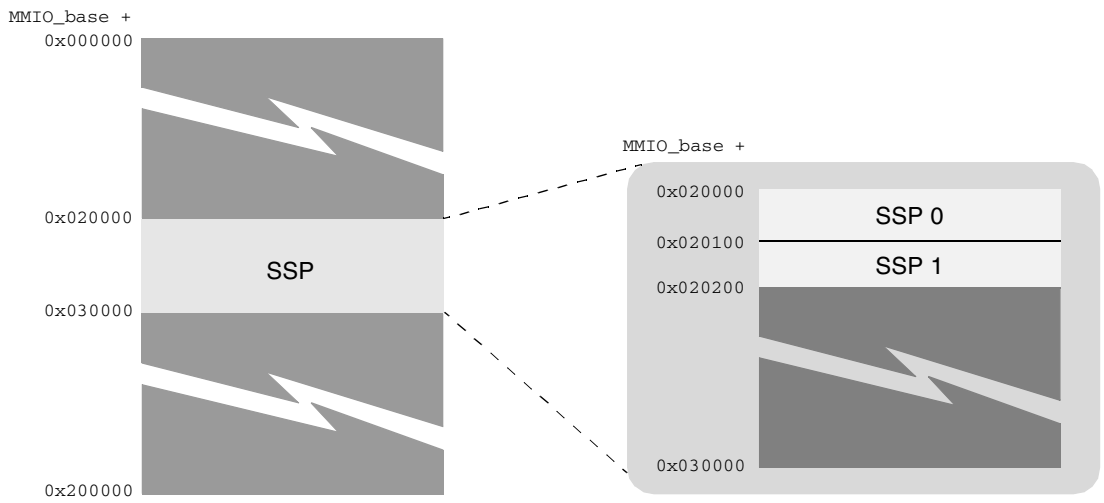
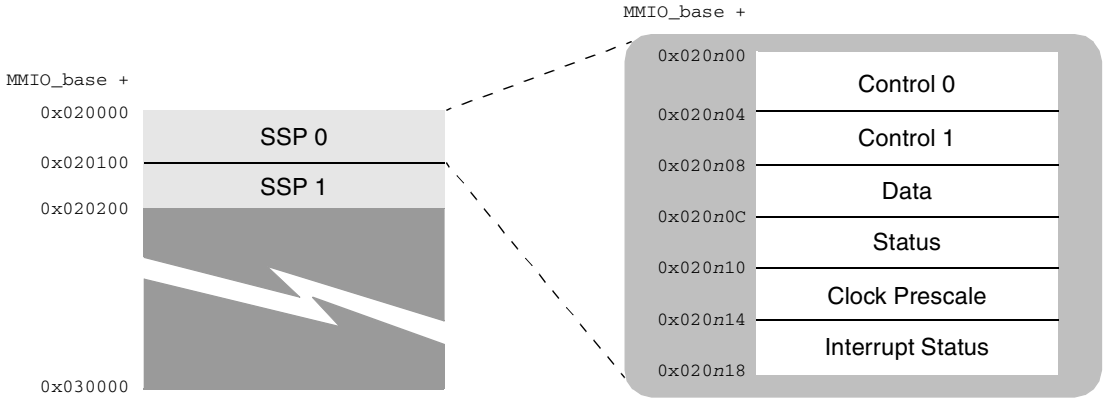


Figure 16-14 shows how each SSP memory region is laid out.

Figure 16-14: Single SSP Register Space



Control 0

Read/Write MMIO_base + 0x020*n*00

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Clock R/W								Ph R/W	Pol R/W	Format R/W		DataSize R/W			

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:8	Clock	Serial Clock Rate. The clock rate is calculated as follows: $\frac{F_{SSPCLK}}{CPSDVSR \times (1 + SCR)}$ where CPSDVSR is an even value from 2 to 254, programmed via the Clock Prescale register.
7	Ph	SCLKOUT Phase for Motorola SPI Frame.
6	Pol	SCLKOUT Polarity for Motorola SPI Frame. 0: Rising edge. 1: Falling edge.

Bit(s)	Name	Description
5:4	Format	Frame Format. 00: Motorola SPI frame format. 01: Texas Instruments serial frame format. 10: National Microwire frame format. 11: Reserved
3:0	DataSize	Data Size Select.

0000	Reserved	0100	5-bit	1000	9-bit	1100	13-bit
0001	Reserved	0101	6-bit	1001	10-bit	1101	14-bit
0010	Reserved	0110	7-bit	1010	11-bit	1110	15-bit
0011	4-bit	0111	8-bit	1011	12-bit	1111	16-bit

Control 1

Read/Write MMIO_base + 0x020*n*04

Power-on Default 0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									O R/W	M R/W	E R/W	L R/W	OI R/W	TI R/W	RI R/W

Bit(s)	Name	Description
31:7	Reserved	These bits are reserved.
6	O	Slave-Mode Output Select. 0: Enable. 1: Disable.
5	M	Mode Select. 0: Master mode. 1: Slave mode.
4	E	SSP Enable. 0: Disable. 1: Enable.
3	L	Loopback Mode. 0: Normal. 1: Internal loopback.
2	OI	Overflow Interrupt Enable. 0: Disable. 1: Enable.

Bit(s)	Name	Description
1	TI	Transmit Interrupt Enable. 0: Disable. 1: Enable.
0	RI	Receive Interrupt Enable. 0: Disable. 1: Enable.

Data

Read/Write MMIO_base + 0x020*n*08

Power-on Default 0x00000000

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

When the SSP is programmed for National Microwire frame format, the default transmit data size is eight bits (the most significant byte is ignored).

Bit(s)	Name	Description
31:16	Reserved	These bits are reserved.
15:0	Data	Data read from receive FIFO or written to transmit FIFO.

Status

Read MMIO_base + 0x020*n*0C

Power-on Default 0x00000003

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

Bit(s)	Name	Description
31:5	Reserved	These bits are reserved.
4	B	SSP Busy Flag. 0: Idle. 1: Busy.
3:2	R	Receive FIFO Status. 00: Empty. 01: Not empty. 11: Full.
1:0	T	Transmit FIFO Status. 00: Full. 10: Not full. 11: Empty.

Clock Prescale

Read/Write MMIO_base + 0x020*n*10

Power-on Default 0x00000000

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

Bit(s)	Name	Description
31:8	Reserved	These bits are reserved.
7:0	Prescale	Clock Prescale Value. This must be an even number, so bit 0 is always “0”.

Interrupt Status

Read/Write MMIO_base + 0x020*n*14

Power-on Default 0b0000.0000.0000.0000.0000.0000.0000.00XX

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

Bit(s)	Name	Description
31:3	Reserved	These bits are reserved.
2	O	Receive FIFO Overflow Interrupt Status. Writing any value to this bit clears the interrupt. 0: Inactive. 1: Active.
1	T	Transmit FIFO Interrupt Status. This bit is read-only. 0: Inactive. 1: Active.
0	R	Receive FIFO Interrupt Status. This bit is read-only. 0: Inactive. 1: Active.

Interrupts

The SSP can generate four interrupts. The following three of these interrupts are individual, maskable, active HIGH interrupts:

- **SSPRXINTR -** SSP receives FIFO service interrupt request

  This receive interrupt is asserted when there are four or more valid entries in the receive FIFO.

- **SSPTXINTR** - SSP transmits FIFO service interrupt request

  This transmit interrupt is asserted when there are four or less valid entries in the transmit FIFO. This interrupt is not qualified with the SSP enable signal, which allows operation in one of two ways. Data can be written to the transmit FIFO prior to enabling the SSP and the interrupts. Alternatively, the SSP and interrupts can be enabled so that data can be written to the transmit FIFO by an interrupt service routine.

- **SSPRORINTR** - SSP receives overrun interrupt request

  This receive overrun interrupt is asserted when the FIFO is already full and an additional data frame is received, causing an overrun of the FIFO. Data is overwritten in the receive shift register but not in the FIFO.

The fourth interrupt, SSPINTR, is a combined single interrupt. The three interrupts also are combined into the single output SSPINTR, which is an OR function of the individual masked sources. This output can be connected to the system interrupt controller to provide another level of masking on an individual per-peripheral basis. The combined SSP interrupt is asserted if any of the three individual interrupts are asserted and enabled.

Each of the three individual maskable interrupts can be enabled or disabled by changing the mask bits in the SSP Control 1 register. Setting the appropriate mask bit HIGH enables the interrupt.

Provision of the individual outputs as well as a combined interrupt output allows use of either a global interrupt service routine, or modular device drivers to handle interrupts.

The transmit and receive dynamic dataflow interrupts SSPTXINTR and SSPRXINTR have been separated from the status interrupts, so that data can be read or written in response to just the FIFO trigger levels.

The status of the individual interrupt sources can be read from the SSP Interrupt Status register.

17 Pin & Packaging Information

Signal List

Table 17-1 summarizes the pins for the SM501 chip. The highlighted lines contain multiplexed pins.

Table 17-1: Signal Summary for the SM501

Signal Category	Number of Pins	Signal Name
GPIO Related Signals (64)		
GPIO	7	GPIO / Interrupt
SSP0	5	SSP0
UART0 / IrDA	4	TXD, RXD, CTS, RTS
UART1 / SSP1 / IrDA	5	TXD, RXD, CTS, RTS / SSP1
I^2C	2	TXD, RXD / IrDA
8051	16	Parallel Interface + Control Signals (8051 u-controller)
Zoom Video	8	ZV[7:0]
ACLink	5	RST, SYNC_48K, BIT_CLK, SDOUT, SDIN
PWM	3	PWM[0:2]
Digital CRT / Zoom Video / FP	9	Digital CRT[7:0], CLK / ZV[15:8] / FP[17:16, 9:8, 1:0]
Panel (25)		
	18	FP[23:18], FP[15:10], FP[7:2]
	7	FP_HSYNC, FP_VSYNC, M/DE, FPCLK, FPEN, VDEN, BIAS, FP_DISP
CPU Interface (75)		
Address	24	CA[25:2]
Data	32	D[31:0]
Control	19	RST#, INTR, HCS#, BS#, HRDY#, HCLK, HCKE, BREQ#, ACK#, MCS#[1:0], HWE#, HRAS#, HCAS#, BE[3:0], CPURD#
External Memory (60)		

Table 17-1: Signal Summary for the SM501

Signal Category	Number of Pins	Signal Name
Data	32	MD[31:0]
Addr	13	MA[12:0]
Control	15	RAS#, CAS#, WE#, CKE, SCK+, SCK-, DSF, DQS, DQM[3:0], BA[1:0], CS#
CRT Output	6	R, G, B, CRT_HSYNC, CRT_VSYNC, IREF
Zoom Video	3	VP_VSYNC, VP_HREF, VP_CLK
USB	4	USB+, USB-, USBGND, USBPWR
Clocks + Signals	6	XTALIN, XTALOUT, Power, GND, CLKOFF, TESTCLK
Test	2	TEST[1:0]
Miscellaneous (1)		
No Connection	1	MVREF
Subtotal (246)		
Power & Ground (51)		
Power & GND	51	
Grand Total (297)		

Pinout

Figure 17-1 shows the pinout for the SM501 chip.

Figure 17-1:Pinout for the SM501

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
A	CA5	CA10	CD4	CD1	CD5	CD8	CD12	CA18	CA24	CA19	CD17	CD21	CD23	CD25	CD31	HCLK	BE3	HCKE	CPURD#	BS#	HCAS#
B	CA4	CA9	CA14	CD6	CD3	CD7	CD10	CD14	CA21	CD16	CD20	CD19	CA20	CD27	CD29	BREQ#	BE2	MCS1#	HRDY#	HRAS#	HWE#
C	CA3	CA8	CA13	CD2	CD9	CD13	CA16	CA22	CA25	CD18	CD22	CD24	CD26	CD30	ACK#	RST#	BE1	MCS0#	HCS#	CLK OFF	XTAL IN
D	CA2	CA7	CA12	MVDD2	CA17	CD11	CD15	VDD	CA23	HVDD	CA15	HVDD	CD28	VDD	INTR	HVDD	BE0	HVDD	PLL GND	TEST CLK	XTAL OUT
E	MD0	CA6	CA11	CD0														GPIO 0	GPIO 1	GPIO 2	PLL PWR
F	MD1	MD2	MD3	MVDD														VDD	GPIO 3	GPIO 4	GPIO 5
G	MD4	MD5	MD6	MD7														GPIO 6	GPIO 7	GPIO 8	GPIO 9
H	MD15	MD14	MD13	MVDD														XTAL PWR	GPIO 10	GPIO 11	GPIO 12
J	MD12	MD11	MD10	MD9					MVDD2	GND	GND	GND	VDD					GPIO 13	GPIO 14	GPIO 15	GPIO 16
K	DQM0	DQM1	MD8	MVREF					GND	GND	GND	GND	GND					GVDD	GPIO 17	GPIO 18	GPIO 19
L	WE#	CAS#	RAS#	CS#					GND	GND	GND	GND	GND					GPIO 20	GPIO 21	GPIO 22	GPIO 23
M	BA1	BA0	MA0	MVDD					GND	GND	GND	GND	GND					VDD	GPIO 24	GPIO 25	GPIO 26
N	MA1	MA2	MA3	MA4					MVDD2	GND	GND	GND	VDD					GPIO 27	GPIO 28	GPIO 29	GPIO 30
P	MA5	MA6	MA7	MVDD														GVDD	GPIO 31	GPIO 32	GPIO 33
R	SCK+	MA12	DSF	MA11														GPIO 34	GPIO 35	GPIO 36	GPIO 37
T	SCK-	MA10	MA9	MVDD														GVDD	GPIO 38	GPIO 39	GPIO 40
U	MA8	CKE	DQM2	DQM3														GPIO 41	GPIO 42	GPIO 43	GPIO 44
V	DQS	MD24	MD25	MVDD2	FP3	PVDD	FP12	PVDD	FP21	VDD	FPEN	PVDD	BIAS	AVDD	TEST 0	AVSS	GPIO 45	VDD	GPIO 46	GPIO 47	GPIO 48
W	MD26	MD27	MD28	MD29	FP2	FP6	FP11	FP15	FP20	FP23	FP_VSYNC	VDEN	CRT_HSYNC	CRT_VSYNC	TEST 1	GPIO 49	GPIO 50	GPIO 51	GPIO 52	GPIO 53	GPIO 54
Y	MD30	MD31	MD23	MD22	MD21	FP5	FP10	FP14	FP19	FP22	FP_HSYNC	FP_DISP	R	USB GND	USB PWR	GPIO 55	GPIO 56	GPIO 57	GPIO 58	GPIO 59	GPIO 60
AA	MD20	MD19	MD18	MD17	MD16	FP4	FP7	FP13	FP18	FPCLK	IREF	B	G	USB-	USB+	VP_VSYNC	VP_HREF	VP_CLK	GPIO 61	GPIO 62	GPIO 63

Pin Descriptions

Each BGA ball of the MMCC is one of the following types:

- I: Input signal
- O: Output signal
- I/O: Input or Output signal

All outputs and I/O signals are 3-stated. Internal pull-ups for I/O pads are all 100KΩ resistors. Internal pull-downs for I/O pads are all 100KΩ resistors.

Table 17-2: Pin Descriptions

Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
Host/PCI Interface (75)					
ACK#	C15	I	CMOS 3.3V		CPU Bus Acknowledge / PCI Bus Grant.
BE[3:0]	A17, B17, C17, D17	I/O	CMOS 3.3V	8	CPU Byte Enable / SDRAM Data Mask.
BREQ#	B16	O	CMOS 3.3V	8, 12, 24	CPU Bus Request / PCI Bus Request.
BS#	A20	I	CMOS 3.3V		SH4 Cycle Start / XScale or NEC CPU Write Enable.
CA[11:2]	E3, A2, B2, C2, D2, E2, A1, B1, C1, D1	I/O	CMOS 3.3V	8	CPU Address [11:2] / SDRAM MA.
CA[13:12]	C3, D3	I/O	CMOS 3.3V	8	CPU Address [13:12] / SDRAM BA / SDRAM MA.
CA[20:17]	B13, A10, A8, D5	I/O	CMOS 3.3V	8, 12, 24	CPU Address [20:17] / PCI C/BE#[3:0] / SDRAM MA.
CA14	B3	I/O	CMOS 3.3V	8	CPU Address 14 / PCI CLKRUN# / SDRAM BA / SDRAM MA.
CA15	D11	I/O	CMOS 3.3V	8, 12, 24	CPU Address 15 / PCI IDSEL / SDRAM BA / SDRAM MA.
CA16	C7	I/O	CMOS 3.3V	8, 12, 24	CPU Address 16 / PCI PAR / SDRAM BA / SDRAM MA.
CA21	B9	I/O	CMOS 3.3V	8, 12, 24	CPU Address 21 / PCI DEVSEL# / SDRAM MA.
CA22	C8	I/O	CMOS 3.3V	8, 12, 24	CPU Address 22 / PCI STOP# / SDRAM BA / SDRAM MA.
CA23	D9	I/O	CMOS 3.3V	8, 12, 24	CPU Address 23 / PCI TRDY# / SDRAM BA / SDRAM MA.
CA24	A9	I/O	CMOS 3.3V	8, 12, 24	CPU Address 24 / PCI IRDY# / SDRAM BA.

Table 17-2: Pin Descriptions

Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
CA25	C9	I/O	CMOS 3.3V	8, 12, 24	CPU Address 25 / PCI FRAME#.
CD[31:0]	A15, C14, B15, D13, B14, C13, A14, C12, A13, C11, A12, B11, B12, C10, A11, B10, D7, B8, C6, A7, D6, B7, C5, A6, B6, B4, A5, A3, B5, C4, A4, E4	I/O	CMOS 3.3V	8, 12, 24	CPU Data Bus / PCI AD Bus / SDRAM Data Bus.
CPURD#	A19	I/O	CMOS 3.3V		XScale or NEC Read Enable.
HCAS#	A21	O	CMOS 3.3V	8	SDRAM Column Address select.
HCKE	A18	I/O	CMOS 3.3V	8	SDRAM Clock Enable.
HCLK	A16	I/O	CMOS 3.3V	8, 12, 24	CPU clock / PCI clock.
HCS#	C19	I	CMOS 3.3V		SM501 chip select. This signal has a weak internal pull-up.
HRAS#	B20	O	CMOS 3.3V	8	SDRAM Row Address select.
HRDY#	B19	O	CMOS 3.3V	8	CPU Ready. This signal must be externally pulled-up for SH4. For all other CPUs, this signal must be pulled-down.
HWE#	B21	I/O	CMOS 3.3V	8	SH4 Write Enable / SDRAM Write Enable.
INTR	D15	O	CMOS 3.3V	8, 12, 24	CPU Interrupt / PCI Interrupt.
MCS#[1:0]	B18, C18	O	CMOS 3.3V	8	SDRAM Chip Select.
RST#	C16	I	CMOS 3.3V		SM501 reset.
Power Down Interface (1)					
CLKOFF	C20	I	CMOS 3.3V		Used to shut off host interface clock. This signal has a weak internal pull-down.
Clock Interface (5)					
PLLGND	D19	I			PLL ground.
PLLPWR	E21	I	1.8 V		PLL power.
TESTCLK	D20	I	CMOS 3.3V		For testing purposes.
XTALIN	C21	I	CMOS 3.3V		12/24MHz crystal input connection.

Table 17-2: Pin Descriptions

Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
XTALOUT	D21	O	CMOS 3.3V		12/24MHz crystal output connection.
Test Interface (2)					
TEST[1:0]	W15, V15	I	CMOS 3.3V		Test mode selection. Both signals have a weak internal pull-down.
Internal Memory Interface (60)					
BA[1:0]	M1, M2	O	CMOS 3.3V	4	SDRAM Bank Address.
CAS#	L2	O	CMOS 3.3V	4	SDRAM Column Address Strobe.
CKE	U2	O	CMOS 3.3V	4	SDRAM Clock Enable.
CS#	L4	O	CMOS 3.3V	4	SDRAM Chip Select.
DQM[3:0]	U4, U3, K2, K1	O	CMOS 3.3V	4	SDRAM Data Mask.
DQS	V1	I/O	CMOS 3.3V or 2.5V	4	For DDR Memory
DSF	R3	O	CMOS 3.3V	4	SGRAM Block Write.
MA[12:0]	R2, R4, T2, T3, U1, P3, P2, P1, N4, N3, N2, N1, M3	O	CMOS 3.3V	4	SDRAM Address bus.
MD[31:0]	Y2, Y1, W4, W3, W2, W1, V3, V2, Y3, Y4, Y5, AA1, AA2, AA3, AA4, AA5, H1, H2, H3, J1, J2, J3, J4, K3, G4, G3, G2, G1, F3, F2, F1, E1	I/O	CMOS 3.3V	4	SDRAM Data bus. These signals have weak, internal pull-down resistors.
MVREF	K4	A	1.25V		For DDR Memory
RAS#	L3	O	CMOS 3.3V	4	SDRAM Row Address Strobe.
SCK-	T1	O	CMOS 3.3V or 2.5V	8	For DDR Memory
SCK+	R1	O	CMOS 3.3V	8	SDRAM Positive Clock.
WE#	L1	O	CMOS 3.3V	4	SDRAM Write Enable.

Table 17-2: Pin Descriptions

Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
Flat Panel Interface (25)					
BIAS	V13	O	CMOS 3.3V	4	Flat panel voltage bias enable.
FP[23:18], FP[15:10], FP[7:2]	W10, y10, V9, W9, Y9, AA9, W8, Y8, AA8, V7, W7, Y7, AA7, W6, Y6, AA6, V5, W5	O	CMOS 3.3V	4	Flat panel data bus {23:18,15:10,7:2}.
FP_DISP	Y12	O	CMOS 3.3V	4	Flat panel display enable.
FP_HSYNC	Y11	O	CMOS 3.3V	4	Flat panel TFT horizontal sync / STN line pulse.
FP_VSYNC	W11	O	CMOS 3.3V	4	Flat panel TFT vertical sync / STN frame pulse.
FPCLK	AA10	O	CMOS 3.3V	4	Flat panel pixel clock.
FPEN	V11	O	CMOS 3.3V	4	Flat panel enable.
VDEN	W12	O	CMOS 3.3V	4	Flat panel VDD enable.
CRT Interface (6)					
B	AA12	O	Analog		CRT blue output.
CRT_HSYNC	W13	O	CMOS 3.3V	4	CRT vertical sync.
CRT_VSYNC	W14	O	CMOS 3.3V	4	CRT horizontal sync.
G	AA13	O	Analog		CRT green output.
IREF	AA11	I	Analog		CRT IREF input.
R	Y13	O	Analog		CRT red output.
USB Interface (4)					
USB+	AA15	I/O	USB Transceiver 3.3V		USB transceiver plus.
USB-	AA14	I/O	USB Transceiver 3.3V		USB transceiver minus.
Video Port Interface (3)					
VP_CLK	AA18	I	CMOS 3.3V		Video port clock.

Table 17-2: Pin Descriptions

Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
VP_HREF	AA17	I	CMOS 3.3V		Video port horizontal reference.
VP_VSYNC	AA16	I	CMOS 3.3V		Video port vertical sync.
GPIO Interface (64) — All 64 GPIO pins have weak, internal pull-down resistors					
GPIO[7:0]	G19, G18, F21, F20, F19, E20, E19, E18	I/O	CMOS 3.3V	4	GPIO[7:0] / 8051 AD[7:0]. Note that these signals also function as strap pins. Refer to Table 1-4 on page 1-21 for more information.
GPIO8	G20	I/O	CMOS 3.3V	4	GPIO8 / 8051 RD#. This signal must be externally pulled-up when used as 8051 RD#.
GPIO9	G21	I/O	CMOS 3.3V	4	GPIO9 / 8051 WR#. This signal must be externally pulled-up when used as 8051 WR#.
GPIO10	H19	I/O	CMOS 3.3V	4	GPIO10 / 8051 ALE#. This signal must be externally pulled-up when used as 8051 ALE#.
GPIO11	H20	I/O	CMOS 3.3V	4	GPIO11 / 8051 WAIT#. This signal must be externally pulled-up when used as 8051 WAIT#.
GPIO[15:12]	J20, J19, J18, H21	I/O	CMOS 3.3V	4	GPIO[15:12] / 8051 A[11:8]. Note that these signals also function as strap pins. Refer to Table 1-4 on page 1-21 for more information.
GPIO[23:16]	L21, L20, L19, L18, K21, K20, K19, J21	I/O	CMOS 3.3V	4	GPIO[23:16] / Video Port D[7:0] / Test Bus [7:0].
GPIO24	M19	I/O	CMOS 3.3V	4	GPIO24 / AC97 RST#. Use external pull-up when using AC97 RST# function.
GPIO25	M20	I/O	CMOS 3.3V	4	GPIO25 / AC97 SYNC / I^2S WS. Use external pull-down when using AC97 SYNC or I^2S WS function.
GPIO26	M21	I/O	CMOS 3.3V	4	GPIO26 / AC97/I^2S BITCLK. Use external pull-up when using AC97/I^2S BITCLK function.
GPIO27	N18	I/O	CMOS 3.3V	4	Can be used for GPIO / AC97 / IIS. When configured for AC97 interface, GPIO27 connects to SDOUT pin of AC97 codec. When configured for IIS interface, GPIO27 connects to DATAIN pin of IIS codec.

Table 17-2: Pin Descriptions

Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
GPIO28	N19	I/O	CMOS 3.3V	4	Can be used for GPIO / AC97 / IIS. When configured for AC97 interface, GPIO28 connects to SDIN pin of AC97 codec. When configured for IIS interface, GPIO28 connects to DATAOUT pin of IIS codec.
GPIO29	N20	I/O	CMOS 3.3V	4	GPIO29 / PWM 0 output / Test Bus [8]. This signal also functions as a strap pin (refer to Table 1-4 on page 1-21).
GPIO30	N21	I/O	CMOS 3.3V	4	Output: GPIO30 / PWM 1 output / Test Bus [9]. Input: GPIO30 / XScale input clock.
GPIO31	P19	I/O	CMOS 3.3V	4	GPIO31 / PWM 2 output / Test Bus [10]. This signal also functions as a strap pin (refer to Table 1-4 on page 1-21).
GPIO32	P20	I/O	CMOS 3.3V	4	GPIO32 / SSP 0 TXD.
GPIO33	P21	I/O	CMOS 3.3V	4	GPIO33 / SSP 0 RXD.
GPIO34	R18	I/O	CMOS 3.3V	4	GPIO34 / SSP 0 FRMOUT.
GPIO35	R19	I/O	CMOS 3.3V	4	GPIO35 / SSP 0 FRMIN.
GPIO36	R20	I/O	CMOS 3.3V	4	GPIO36 / SSP 0 CLK.
GPIO37	R21	I/O	CMOS 3.3V	4	GPIO37 / UART/IrDA 0 TXD.
GPIO38	T19	I/O	CMOS 3.3V	4	GPIO38 / UART/IrDA 0 RXD.
GPIO39	T20	I/O	CMOS 3.3V	4	GPIO39 / UART 0 CTS.
GPIO40	T21	I/O	CMOS 3.3V	4	GPIO40 / UART 0 RTS.
GPIO41	U18	I/O	CMOS 3.3V	4	GPIO41 / SSP 1 TXD / UART/IrDA 1 TXD.
GPIO42	U19	I/O	CMOS 3.3V	4	GPIO42 / SSP 1 RXD / UART/IrDA 1 RXD.
GPIO43	U20	I/O	CMOS 3.3V	4	GPIO43 / SSP 1 FRMIN / UART 1 CTS.
GPIO44	U21	I/O	CMOS 3.3V	4	GPIO44 / SSP 1 FRMOUT / UART 1 RTS.
GPIO45	V17	I/O	CMOS 3.3V	4	GPIO45 / SSP1 CLK.
GPIO46	V19	I/O	CMOS 3.3V	4	GPIO46 / I^2C Clock. Use external pull-up when using I^2C Clock function.

Table 17-2: Pin Descriptions

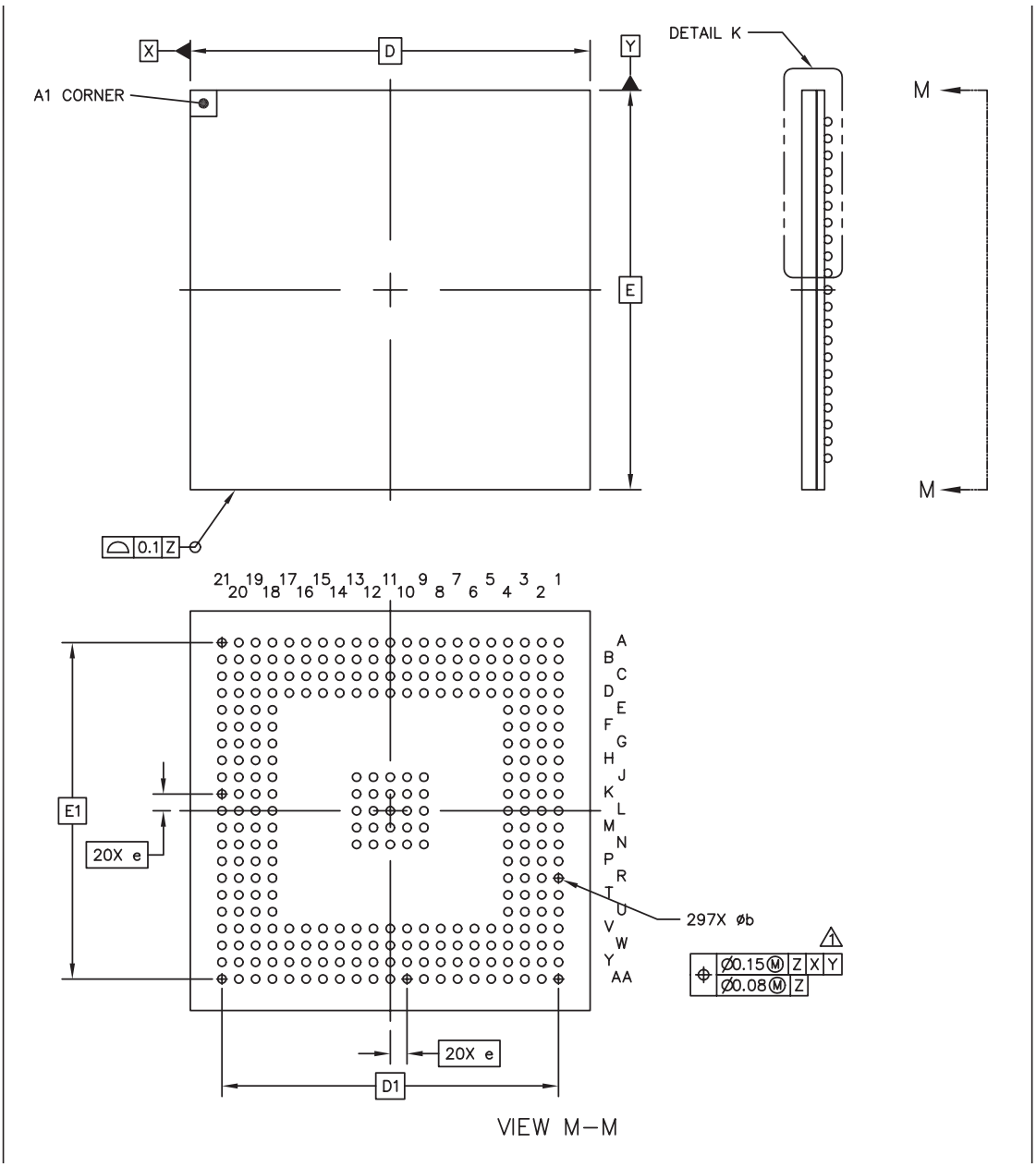
Signal Name	Pin Number	Type	Pad	IOL (ma)	Description
GPIO47	V20	I/O	CMOS 3.3V	4	GPIO47 / I^2C Data. Use external pull-up when using I^2C Data function.
GPIO[53:48]	W20, W19, W18, W17, W16, V21	I/O	CMOS 3.3V	4	GPIO[53:48] / 8051 P1[5:0].
GPIO54	W21	I/O	CMOS 3.3V	4	GPIO54 / 8051 INTR#.
GPIO55	Y16	I/O	CMOS 3.3V	4	GPIO55 / Digital CRT Clock / Test Bus [11].
GPIO[63:54]	Z21, Z20, Z19, Y21, Y20, Y19, Y18, Y17, Y16, W21 (GPIO54 repeated twice)	I/O	CMOS 3.3V	4	GPIO[63:54] / Digital CRT Data [7:0] / Test Bus [19:12] / Video Port Data [15:8] / FPDATA {17:16,9:8,1:0} in lower 6 bits / Embedded Memory Test [7:0].
Power and Ground (51)					
AVDD	V14	I	3.3V		DAC
AVSS	V16	I	0 V		DAC Ground.
GND	K9, L9, M9, J10, K10, L10, M10, N10, J11, K11, L11, M11, N11, J12, K12, L12, M12, N12, K13, L13, M13	I	0 V		Core Ground.
GVDD	K18, P18, T18	I	3.3V		GPIO
HVDD	D10, D12, D16, D18	I	3.3V		Host/PCI I/O
MVDD	F4, H4, M4, P4, T4	I	2.5V/3.3V		SDRAM Core
MVDD2	D4, J9, N9, V4	I	CMOS 3.3V or 2.5V		SDRAM or DDR
PVDD	V6, V8, V12	I	3.3V		LCD Panel I/O
USBGND	Y14	I	0 V		USB ground
USBPWR	Y15	I	3.3V		USB power
VDD	D8, D14, F18, J13, M18, N13, V10, V18	I	1.8 V		Core
XTALPWR	H18	I	3.3 V		Crystal

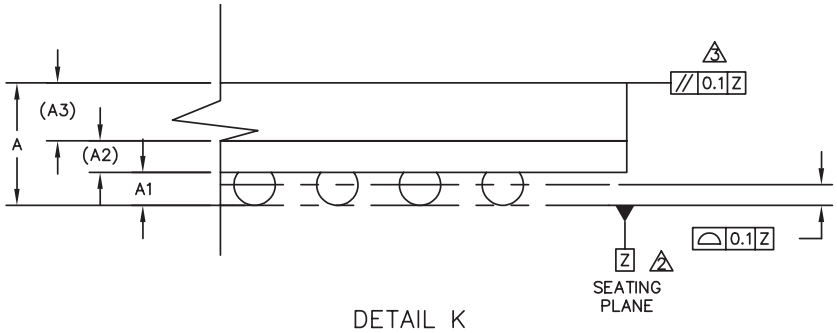
Note: All output pins can be 3-stated.

Packaging

The SM501's initial package is a 297-pin BGA using 0.8mm ball spacing (see Figure 17-2). The total package size is 19x19mm, which allows the I/O and external memory interfaces to be present. The SM501 is available in configurations with internal memory included in the package as well as without memory. Please refer to Figure 17-1 for package pinout details.

Figure 17-2:SM501 Package Outline (two sheets)





DETAIL K
(ROTATE 90°)

DIM	MIN	NOM	MAX
A	---		1.5
A1	0.3		0.4
A2		0.38 REF	
A3		0.7 REF	
b	0.35		0.45
D		19 BSC	
E		19 BSC	
e		0.8 BSC	
D1		16 BSC	
E1		16 BSC	

NOTES

1. DIMENSION b IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER, PARALLEL TO DATUM PLANE Z.
2. DATUM Z IS DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
3. PARALLELISM MEASUREMENT SHALL EXCLUDE ANY EFFECT OF MARK ON TOP SURFACE OF PACKAGE.

UNIT	DIMENSION AND TOLERANCES	REFERENCE DOCUMENT
MM	ASME Y14.5M	JEDEC MO-219

18 Specifications

Environmental

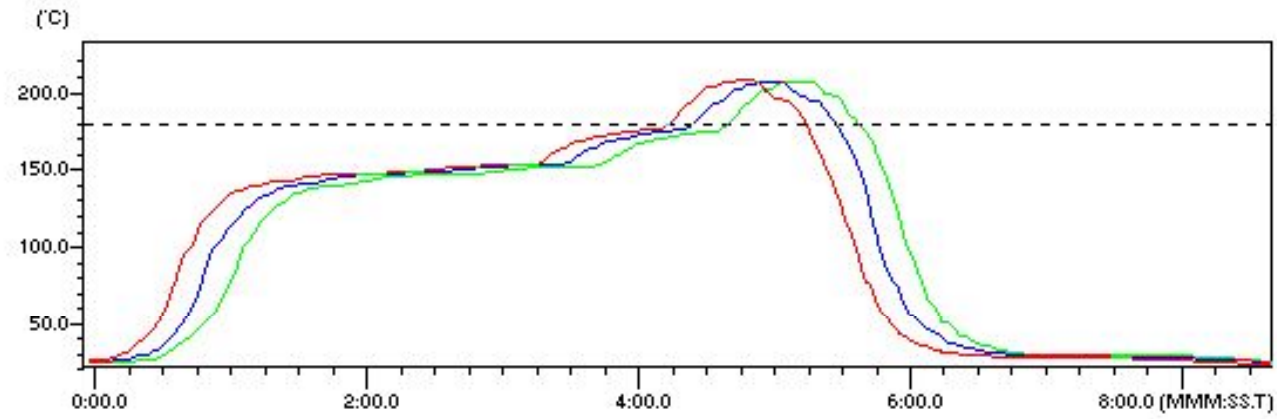
The SM501 meets the following temperature ranges:

- **Storage:** -65 to +150 °C.
- **Operating:** -40 to +85 °C.

Soldering Profile

Figure 18-1 shows the soldering profile for the SM501 device. This profile is designed for use with Sn63 or Sn62 (tin measurements in the PCB) and can serve as a general guideline in establishing a reflow profile.

Figure 18-1: Temperature Profile



The reflow profile is defined as follows;

- Heat up at 1 to 3°C/sec to 140 °C.
- Preheat at 140 to 150 °C for 120 to 160 seconds.
- Ramp at 2 to 3 °C/sec to peak temperature (220 to 225 °C), temperature over 183 °C for 45 to 75 seconds.
- Cool down to room temperature at 4 to 2 °C/sec to avoid undesired intermetallic compound layer.

DC Characteristics

Table 18-1: Absolute Maximum Ratings

Parameter	Maximum Rating	Units
Ambient Temperature	0 to 75	degrees C
Storage Temperature	-40 to 125	degrees C
VDD I/O[1] with respect to VSS	3.3 +/- 5%	V
VDD core with respect to VSS	1.8 +/- 5%	V
ESD Rating (human body), all signal pins	>2000	V
ESD Rating (human body), all VDD pins (AVDD, PVDD, VDD)	>1500	V

1. VDD I/O refers to AVDD, GVDD, HVDD, MVDD, MVDD2, PVDD, USBPWR, XTALPWR.

Table 18-2: DC Characteristics

Symbol	Parameter	Min	Max	Units
V_{IL}	Input Low Voltage		0.8	V
V_{IH}	Input High Voltage	2.4		V
V_{OL}	Output Low Voltage		0.4	V
V_{OH}	Output High Voltage	2.8	VDD + 0.5	V
I_{OZL}	Output 3-State Current		10	uA
I_{OZH}	Output 3-State Current		10	uA
I_{OZL} (pull up pins)	Output 3-State Current	-130	-10	uA
I_{OZH} (pull up pins)	Output 3-State Current		10	uA
I_{OZL} (pull down pins)	Output 3-State Current		10	uA
I_{OZH} (pull down pins)	Output 3-State Current	10	130	uA
C_{IN}	Input Capacitance		7	pF
C_{OUT}	Output Capacitance		7	pF

AC Timing

This section provides the AC timing waveforms and parameters:

- "PCI Interface Timing" on page 18-3
- "Host Interface Timing" on page 18-7
- "Display Controller Timing" on page 18-16
- "USB Interface Timing" on page 18-18
- "ZV Port Timing" on page 18-19
- "UART Timing" on page 18-20
- "AC97-Link and I^{2}S Timing" on page 18-22
- "8051 µ-Controller Timing" on page 18-25
- "Local SDRAM Timing" on page 18-28

PCI Interface Timing

Figure 18-2 shows the PCI clock and its timing parameters. Table 18-3 provides the values for the PCI clock timing parameters shown in Figure 18-2.

Figure 18-2: PCI Clock Timing

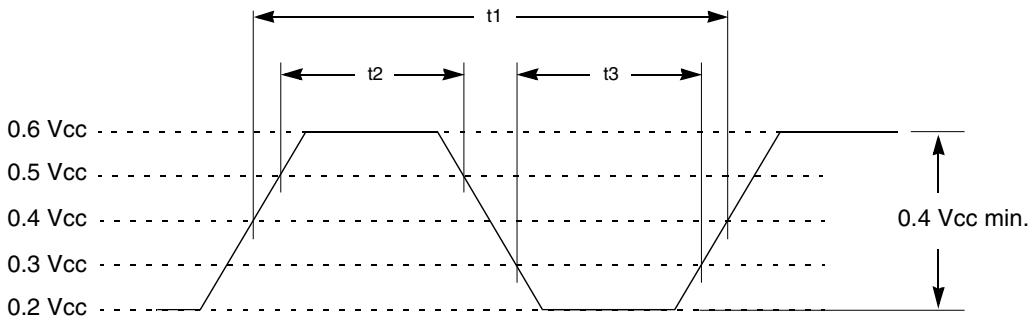


Table 18-3: PCI Clock Timing Parameters

Sym	Parameter	66 MHz		33 MHz		Unit
		Min	Max	Min	Max	
t1	CLK cycle time	15	30	30		ns
t2	CLK high time	6		11		ns
t3	CLK low time	6		11		ns
–	CLK skew rate	1.5	4	1	4	V/ns

Figure 18-3 and Figure 18-4 show the PCI outputs and inputs, respectively, and their relationship to the PCI clock. Table 18-4 provides the values for the timing parameters shown in the two figures.

Figure 18-3: PCI Clock to Output Timing

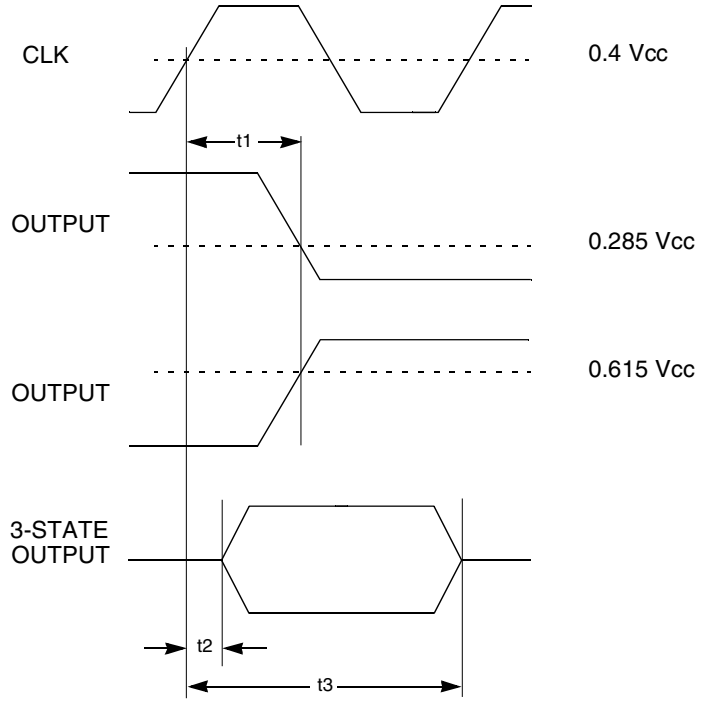


Figure 18-4: PCI Clock to Input Timing

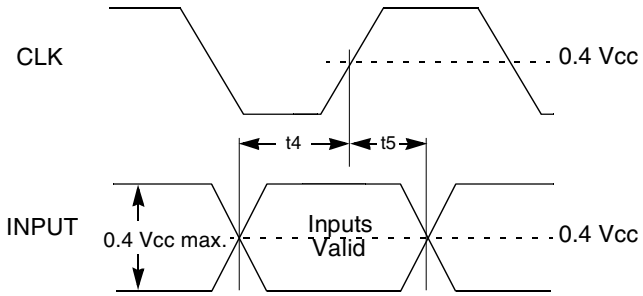


Table 18-4: PCI I/O Timing Parameters

Sym	Parameter	66 MHz		33 MHz		Unit
		Min	Max	Min	Max	
t1	CLK to signal valid delay	2	6	2	11(12)[1]	ns
t2	Float to active delay	2		2		ns
t3	Active to float delay		14		28	ns
t4	Input setup time to CLK	3(5)		7(10.12)		ns
t5	Input hold time from CLK	0		0		ns

1. Values shown in parentheses are for point-to-point signals.

Figure 18-5 shows the remaining timing PCI bus waveforms. Table 18-5 provides the values for the timing parameters shown in Figure 18-5.

Figure 18-5: PCI Bus Timing Diagram

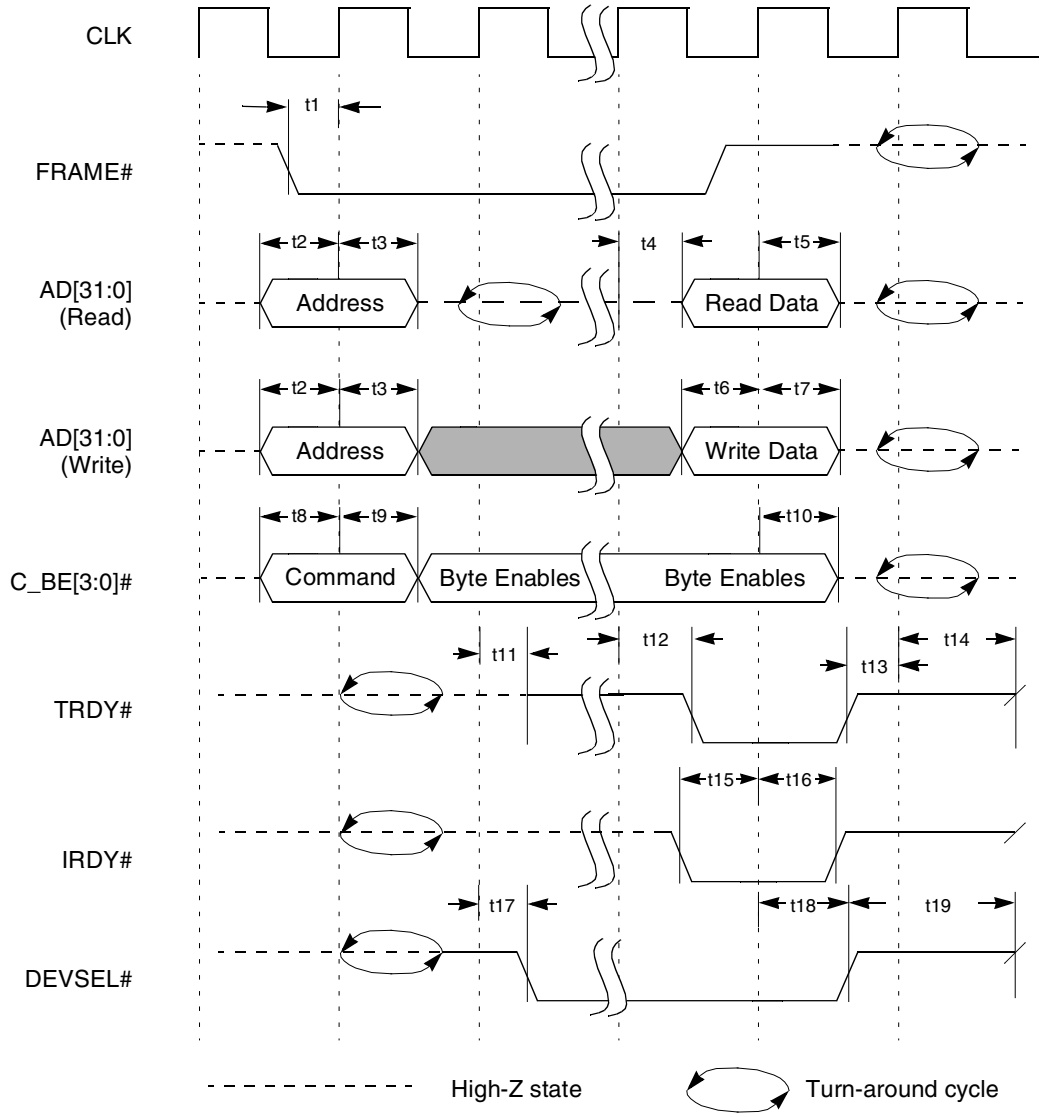


Table 18-5: PCI Bus Timing Parameters (33 MHz)

Symbol	Parameter	Min	Max	Unit
t1	FRAME# setup to CLK	7	-	ns
t2	AD[31:0] (address) setup to CLK	7	-	ns
t3	AD[31:0] (address) hold from CLK	0	-	ns
t4	AD[31:0] (Read Data) valid from CLK	2	11	ns
t5	AD[31:0] (Read Data) hold from CLK	0	-	ns
t6	AD[31:0] (Write Data) setup to CLK	7	-	ns
t7	AD[31:0] (Write Data) hold from CLK	0	-	ns
t8	C/BE[3:0]# (Command) setup to CLK	7	-	ns
t9	C/BE[3:0]# (Command) hold from CLK	0	-	ns
t10	C/BE[3:0]# (Byte Enable) hold from CLK	0	-	ns
t11	TRDY High-Z to High from CLK	2	-	ns
t12	TRDY# active from CLK	2	11	ns
t13	TRDY# inactive from CLK	2	11	ns
t14	TRDY# High before High-Z	1T	-	CLK
t15	IRDY# setup to CLK	7	-	ns
t16	IRDY# hold from CLK	0	-	ns
t17	DEVSEL# active from CLK	2	11	ns
t18	DEVSEL# inactive from CLK	2	11	ns
t19	DEVSEL# High before High-Z	1T	-	CLK

Host Interface Timing

Intel XScale Host

Figure 18-6 shows the timing waveforms for XScale System DRAM operations. Figure 18-7 shows the timing waveforms for XScale read operations. Figure 18-8 shows the timing waveforms for XScale write operations. Table 18-6 lists the AC timing values for the parameters shown in the three XScale figures.

Figure 18-6: XScale System DRAM (Master) Timing

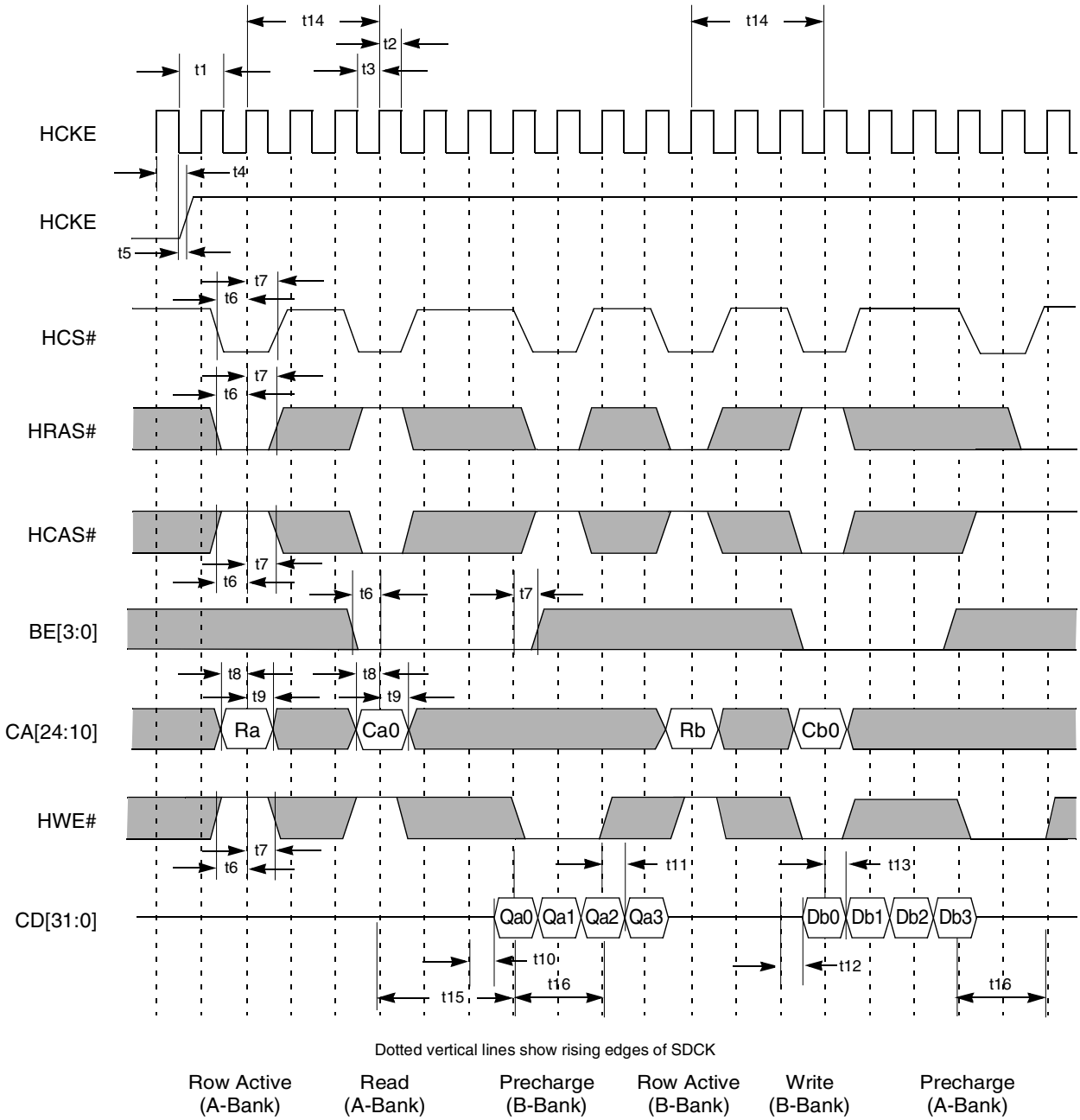


Figure 18-7: XScale Read Timing

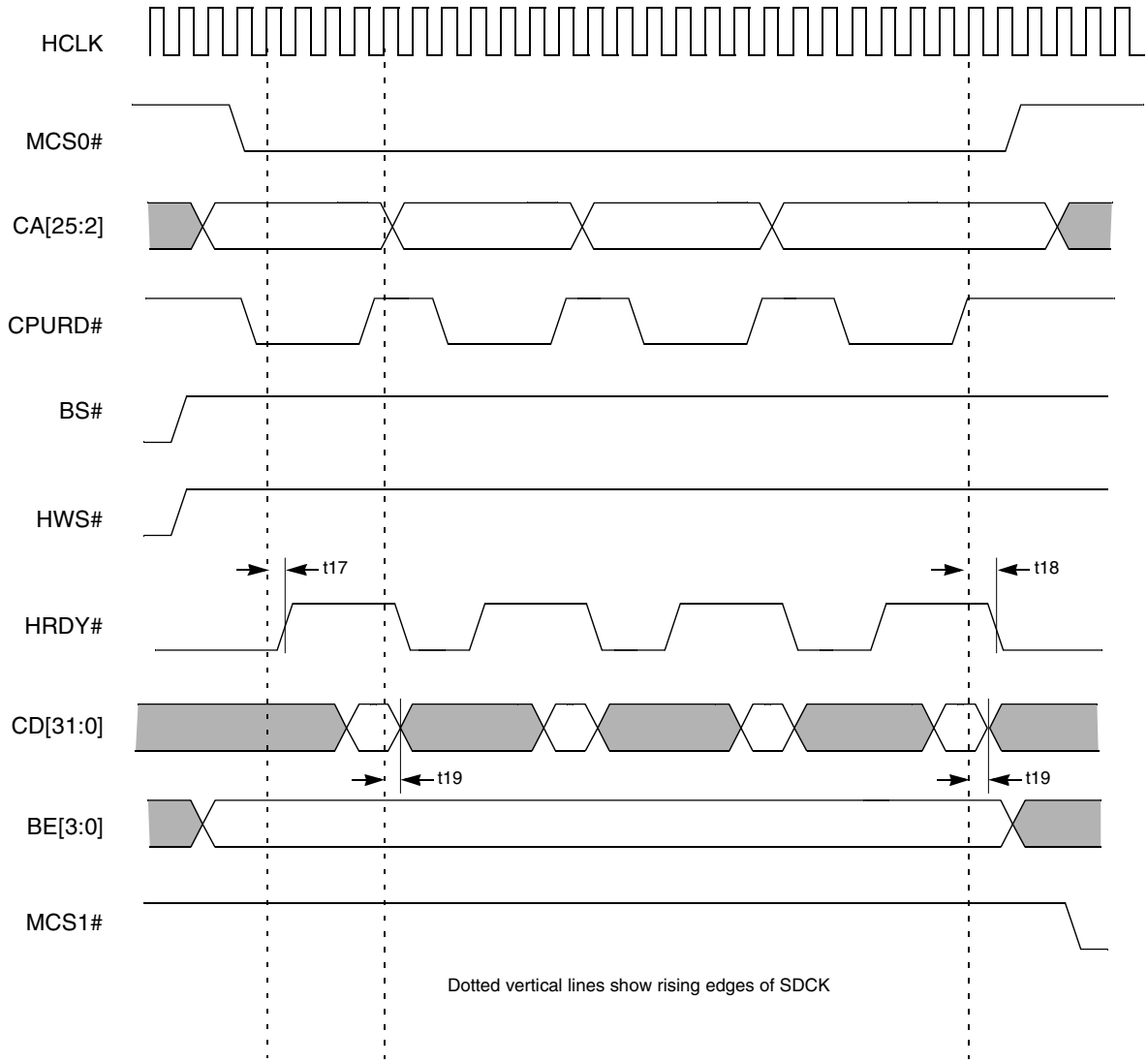


Figure 18-8: XScale Write Timing

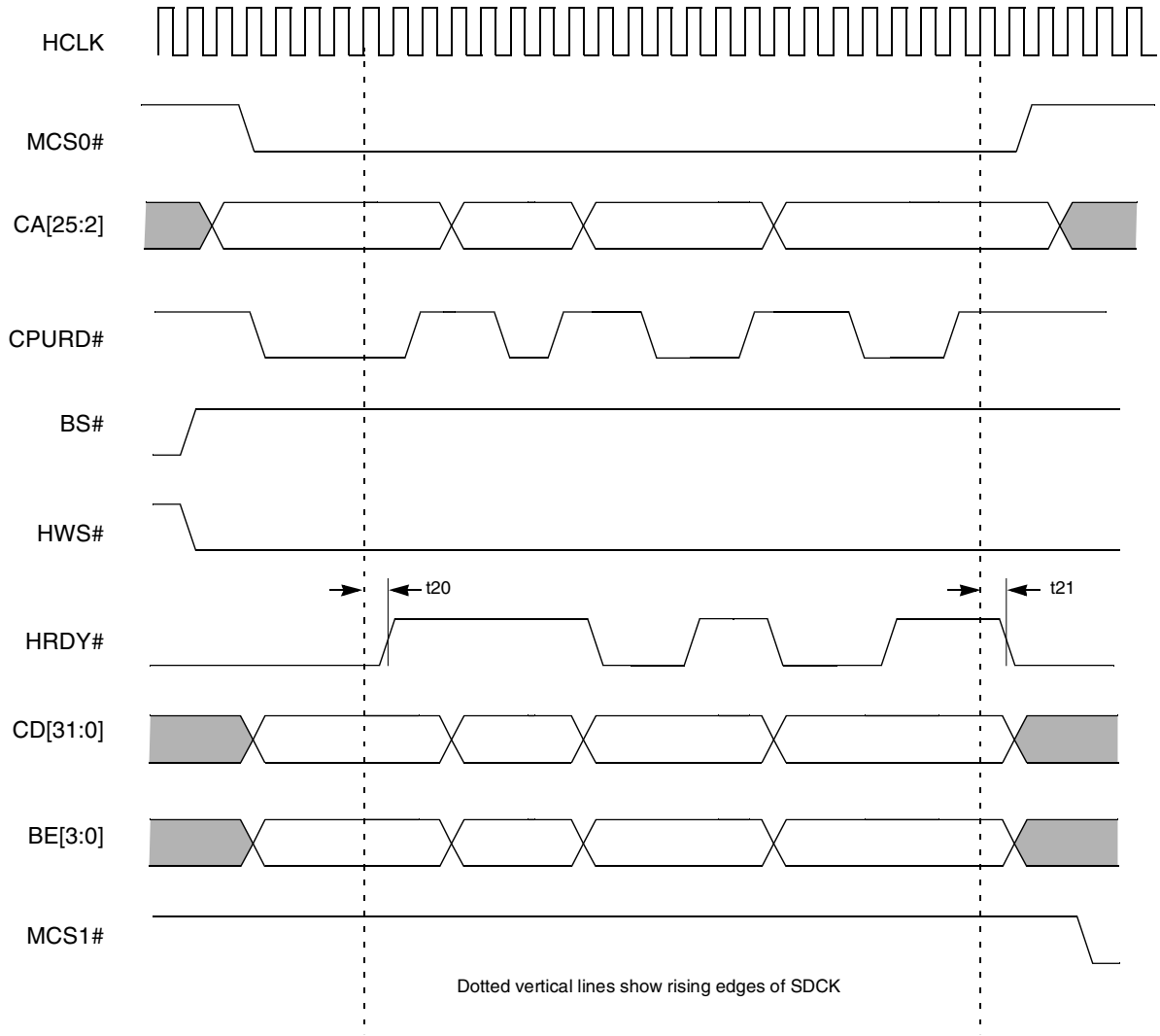


Table 18-6: XScale Timing Parameters

Symbol	Parameter	Min	Max	Units
XScale SDRAM Timing				
t1	HCLK Cycle Time	12		ns
t2	HCLK High Time	4		ns
t3	HCLK Low Time	4		ns
t4	HCKE hold time	3.5		ns
t5	HCKE setup time	3.5		ns
t6	Command setup time	3.5		ns
t7	Command hold time	3.5		ns
t8	Address setup time	3.5		ns
t9	Address hold time	2.5		ns
t10	Access time from HCLK		t1 - 2	ns
t11	CD Out hold time from HCLK	4		ns
t12	CD In setup time from HCLK	3.5		ns
t13	CD In hold time from HCLK	3.5		ns
t14	Active to READ, WRITE delay (Row Active)	3* t1		ns
t15	READ Latency	3* t1 or 2* t1[1]		ns
t16	Write recovery time (Precharge)	2* t1		ns
XScale Read Timing				
t17	Ready latency		4	ns
t18	Read valid delay		2	ns
t19	Data hold time		2	ns
XScale Write Timing				
t20	Ready latency		4	ns
t21	Read valid delay		2	ns

1. Programmable value in DRAM Control register, bit 27 (see “DRAM Control” on page 2-12).

Hitachi SH4 Host

Figure 18-9 shows the timing waveforms for SH4 System DRAM operations. Figure 18-10 shows the timing waveforms for SH4 read operations. Figure 18-11 shows the timing waveforms for SH4 write operations. Table 18-7 lists the AC timing values for the parameters shown in the three SH4 figures.

Figure 18-9: SH4 System DRAM (Master) Timing

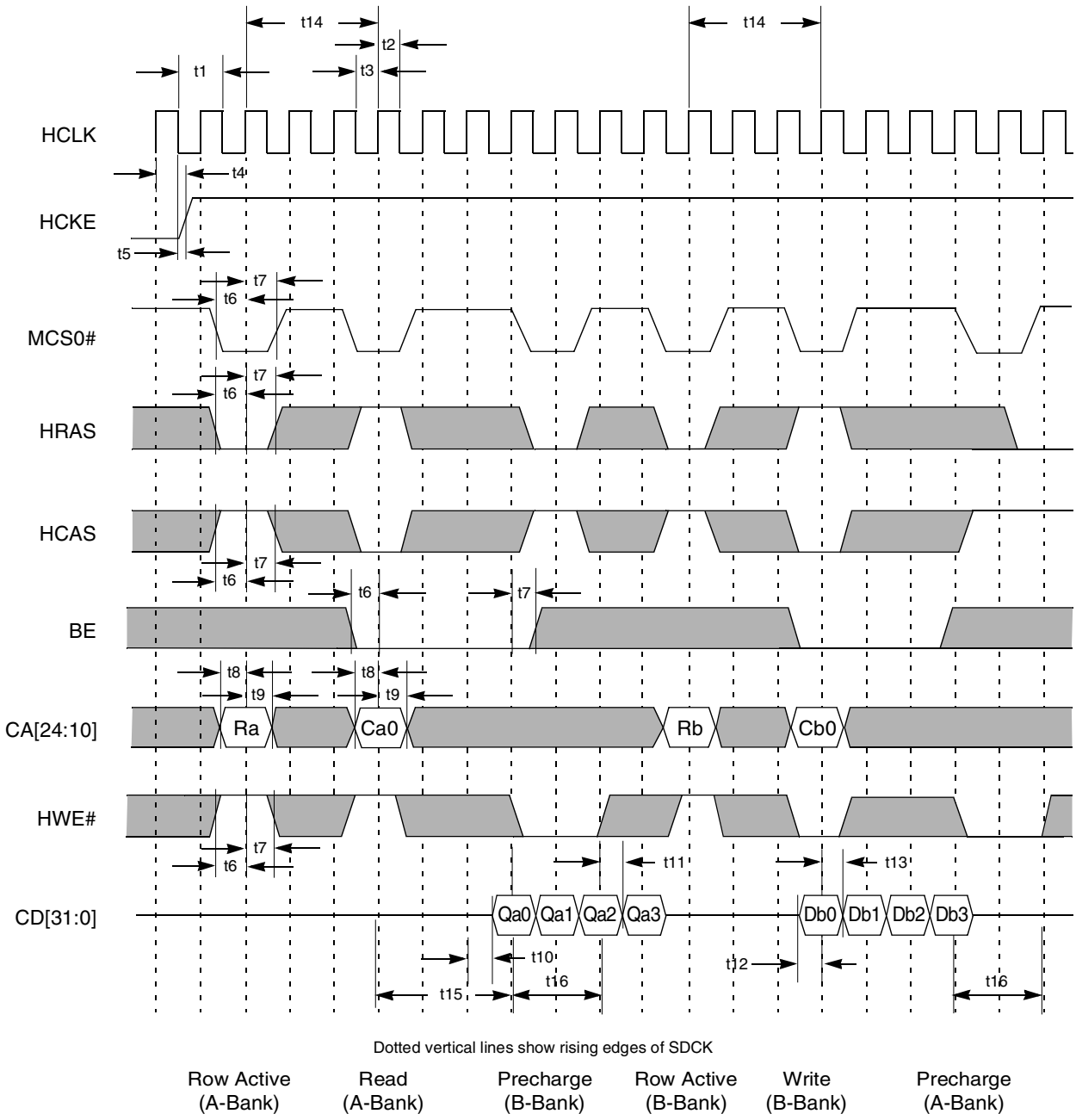


Figure 18-10: SH4 Read Timing

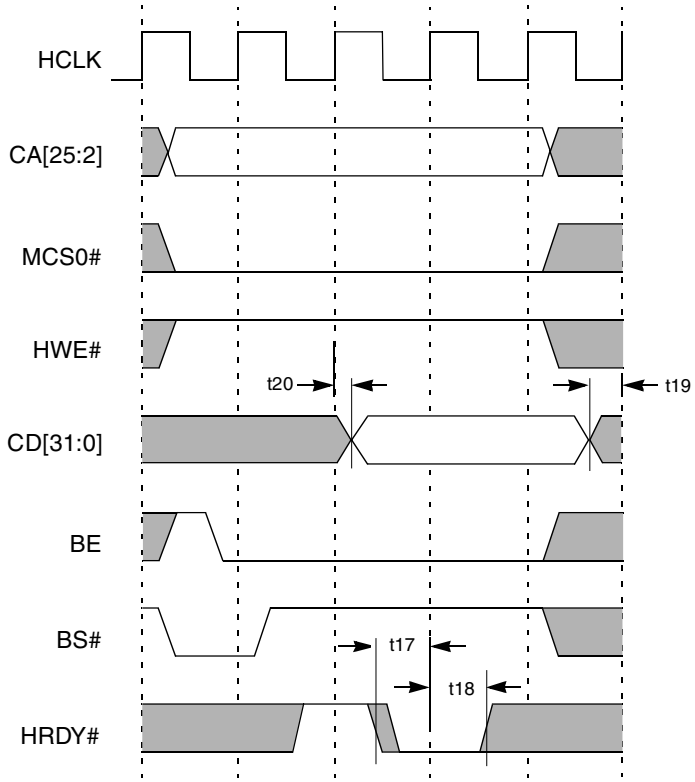


Figure 18-11: SH4 Write Timing

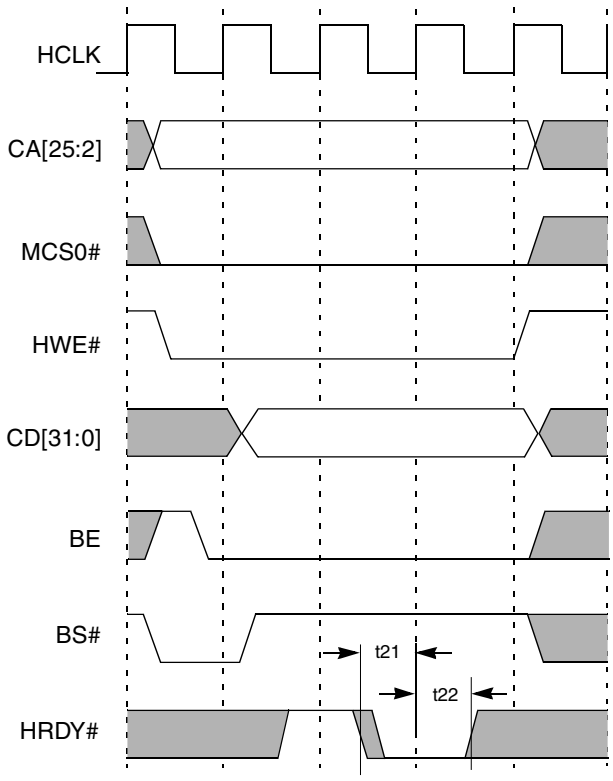


Table 18-7: SH4 Timing Parameters

Symbol	Parameter	Min	Max	Units
SH4 SDRAM Timing				
t1	HCLK Cycle Time	12		ns
t2	HCLK High Time	4		ns
t3	HCLK Low Time	4		ns
t4	HCKE hold time	3.5		ns
t5	HCKE setup time	3.5		ns
t6	Command setup time		6	ns
t7	Command hold time		2	ns
t8	Address/BA setup time	3.5		ns
t9	Address/BA hold time	2.5		ns
t10	Access time from SDCK		t1 - 2	ns
t11	Data Out hold time from HCLK	4		ns
t12	Data In setup time from HCLK	3.5		ns
t13	Data In hold time from HCLK	3.5		ns
t14	Active to READ, WRITE delay (Row Active)	3* t1		ns
t15	READ Latency	3* t1 or 2* t1[1]		ns
t16	Write recovery time (Precharge)	2* t1		ns
SH4 Read Timing				
t17	Ready setup time		6	ns
t18	Ready hold time		2	ns
t19	Read hold time		2	ns
t20	Data valid delay time		4	ns
SH4 Write Timing				
t21	Ready setup time		6	ns
t22	Ready hold time		2	ns

1. Programmable value in DRAM Control register, bit 27 (see “DRAM Control” on page 2-12).

NEC MIPS VR4122/4131 Host

Figure 18-12: NEC System DRAM (Master) Timing

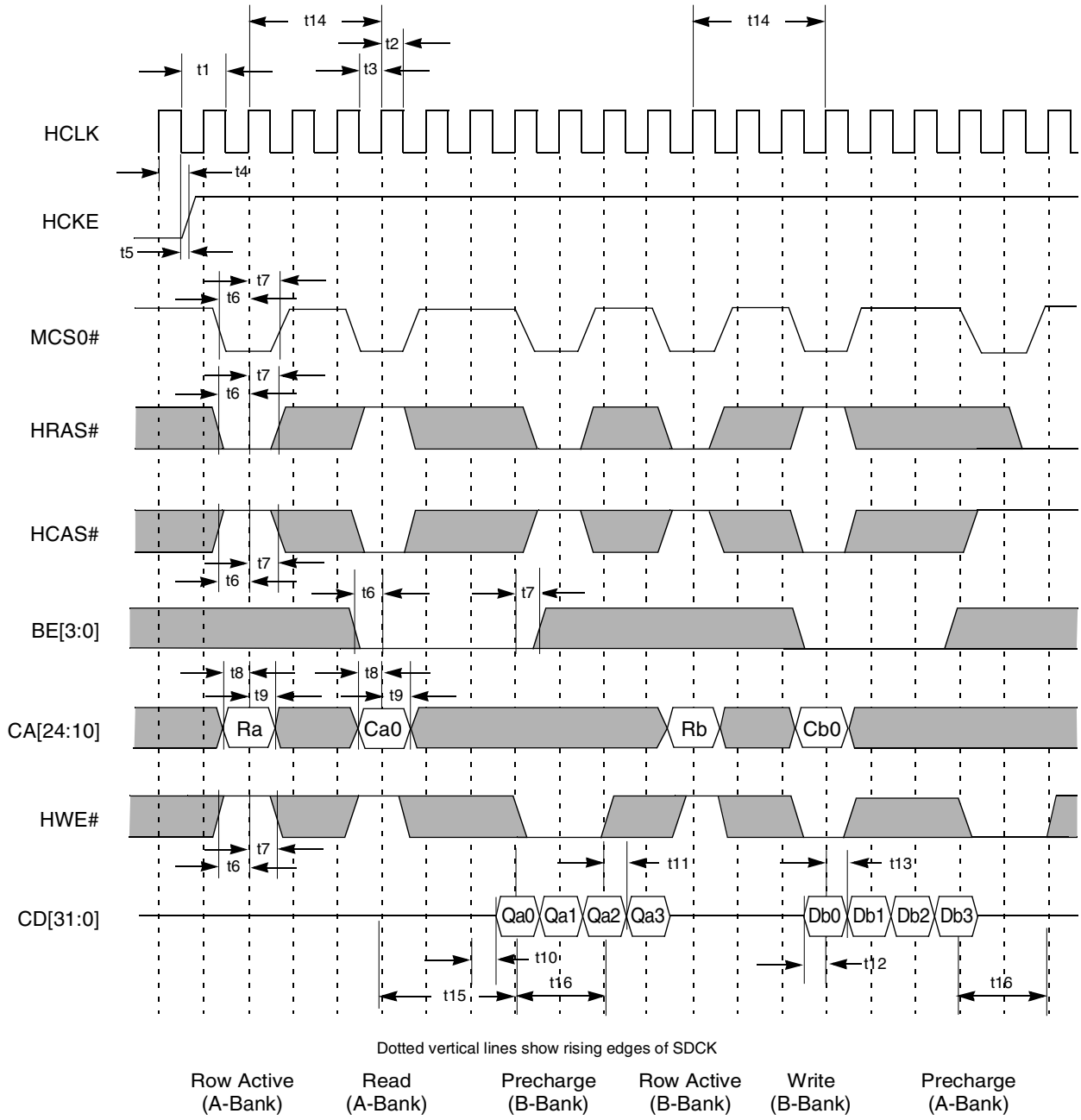


Figure 18-13: NEC DRAM (Slave) Timing

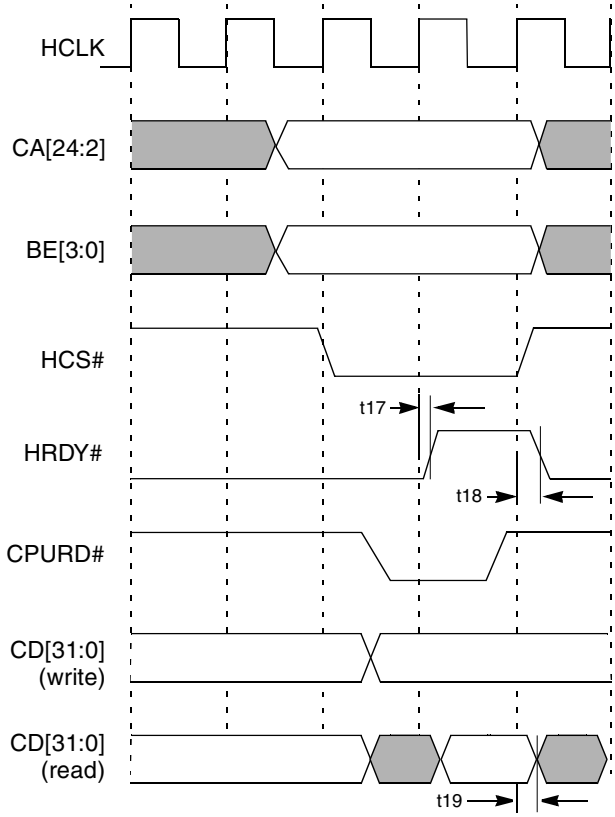


Table 18-8: NEC MIPS Timing Parameters

Symbol	Parameter	Min	Max	Units
NEC SDRAM Master Timing				
t1	HCLK Cycle Time	12		ns
t2	HCLK High Time	4		ns
t3	HCLK Low Time	4		ns
t4	HCKE hold time	3.5		ns
t5	HCKE setup time	3.5		ns
t6	Command setup time		6	ns
t7	Command hold time		2	ns
t8	Address/BA setup time	3.5		ns
t9	Address/BA hold time	2.5		ns
t10	Access time from HCLK		t1 - 2	ns
t11	Data Out hold time from HCLK	4		ns
t12	Data In setup time from HCLK	3.5		ns
t13	Data In hold time from HCLK	3.5		ns
t14	Active to READ, WRITE delay (Row Active)	3* t1		ns

Table 18-8: NEC MIPS Timing Parameters (Continued)

Symbol	Parameter	Min	Max	Units
t15	READ Latency	3* t1 or 2* t1[1]		ns
t16	Write recovery time (Precharge)	2* t1		ns
NEC DRAM Slave Timing				
t17	Ready latency		4	ns
t18	Ready hold time		2	ns
t19	Data hold time		2	ns

1. Programmable value in DRAM Control register, bit 27 (see "DRAM Control" on page 2-12).

Display Controller Timing

Color TFT Interface

Figure 18-14 shows the timing waveforms for the FP and FP_DISP signals. Figure 18-15 shows the timing waveforms for the FP_HSYNC and FP_VSYNC signals. Table 18-9 lists the AC timing values for the parameters shown in the two figures.

Figure 18-14: FP and FP_DISP Timing

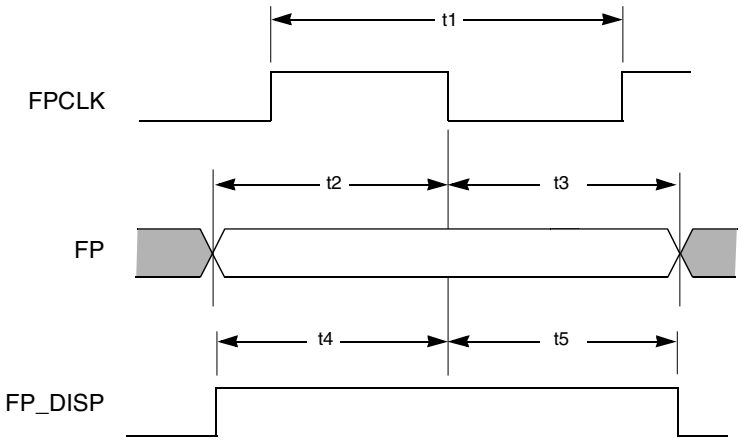
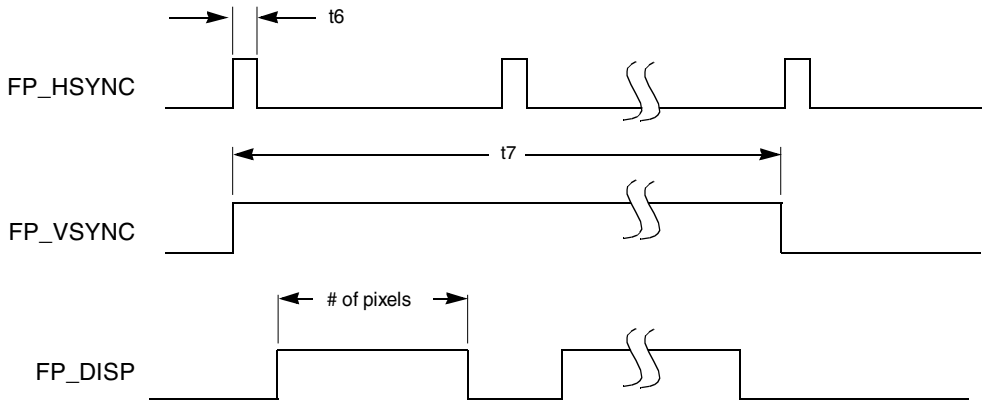


Figure 18-15: FHSYNC and FVSYNC Timing



Note: Number of pixels is programmed in the Panel Horizontal Total register, bits [11:0] (see "Panel Horizontal Total" on page 5-15).

Table 18-9: Color TFT Interface Timing Parameters

Symbol	Parameter	Min	Max	Units
t1	TFT FPCLK cycle time	12		ns
t2	FP setup to FPCLK falling edge	0.5*T[1] - 2		ns
t3	FP hold from FPCLK falling edge	0.5*T - 2		ns
t4	FP_DISP setup to FPCLK falling edge	0.5*T - 2		ns
t5	FP_DISP hold from FPCLK falling edge	0.5*T - 2		ns
t6	FP_HSYNC pulse width	8	16	T
t7	FP_VSYNC pulse width	1		FP_HSYNC

1. T is pixel clock rate on LCD.

USB Interface Timing

The timing for the USB interface is compliant with the USB 1.1 Specification. Figure 18-16 shows the USB timing waveforms. Table 18-10 lists the AC timing values for the parameters shown in the USB figure.

Figure 18-16: USB Timing

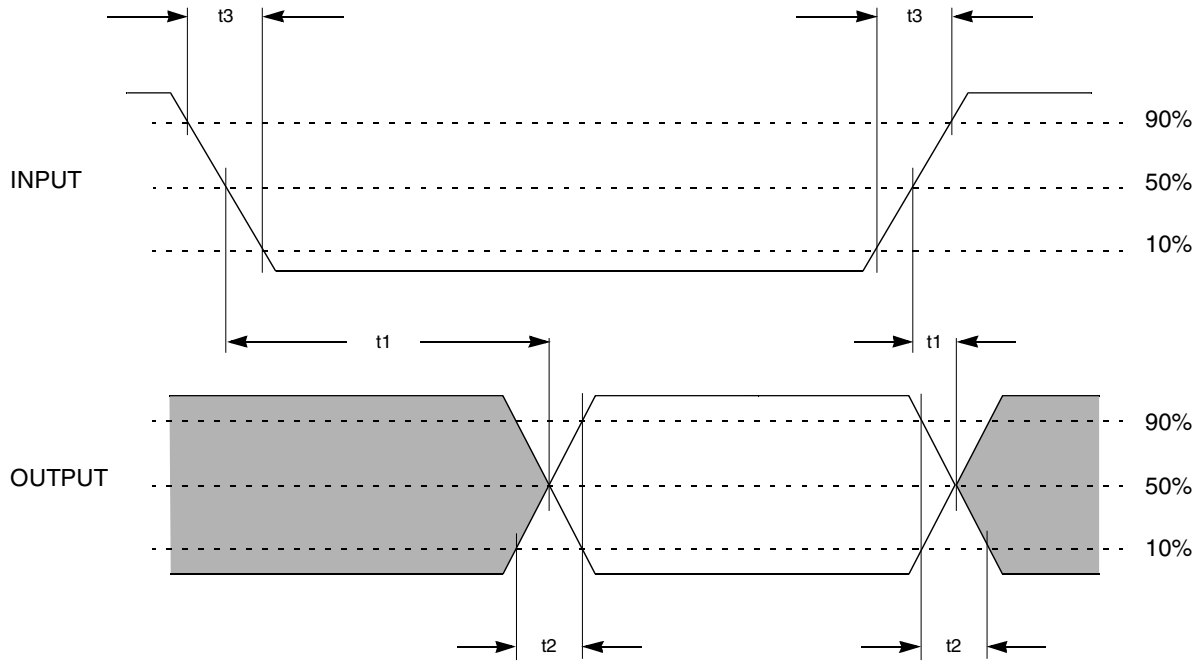


Table 18-10: USB Timing Parameters

Sym	Parameter	Low Speed			Full Speed			Unit
		Min	Typ	Max	Min	Typ	Max	
t1	VIP/VIN to DP/DN	75	150	220	5	8	10	ns
t2	Rise/fall times	75	130	300	4	10	20	ns
t3	Input rise and fall times		3			3		ns

ZV Port Timing

Figure 18-17 depicts the relationship amongst the ZV Port signals. Table 18-11 shows the AC parameters associated with the ZV Port signals when the ZV Port custom interface is in use at 50 MHz.

Figure 18-17: ZV Port Timing

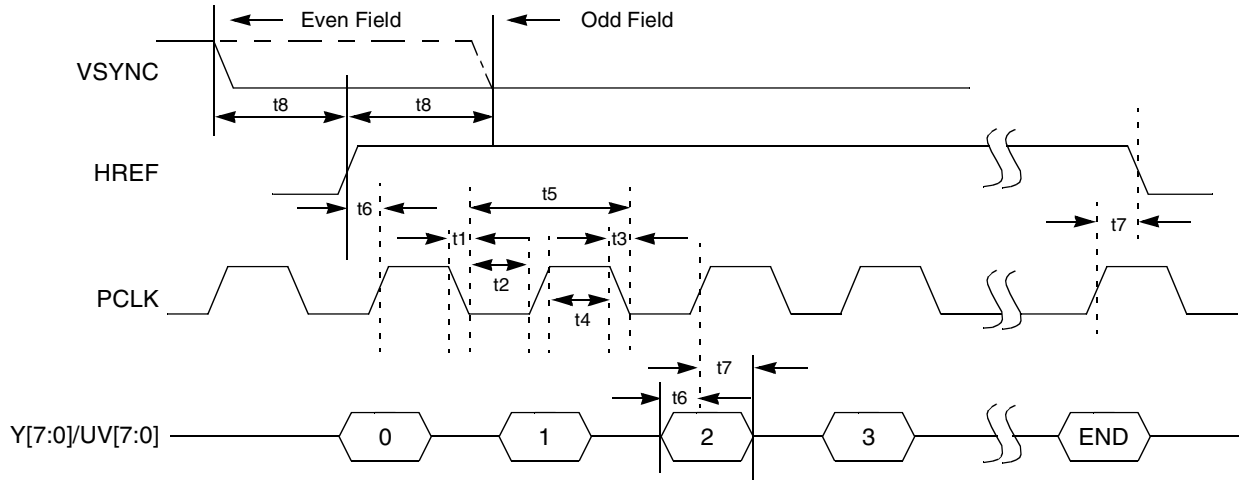


Table 18-11: ZV Port Timing Parameters

Symbol	Parameter	Min	Max	Units
t1	PCLK fall time	2		ns
t2	PCLK low time	7		ns
t3	PCLK rise time	2		ns
t4	PCLK high time	7		ns
t5	PCLK cycle time	20		ns
t6	Y[7:0] / UV[7:0] / HREF setup time	10		ns
t7	Y[7:0] / UV[7:0] / HREF hold time	3		ns
t8	VSYNC setup / hold time to HREF	30		ns

Note: All video signals have minimum rise and fall times of 4 ns and maximum rise and fall times of 8 ns. Non-interlaced data asserts VSYNC at the Odd Field timing.

UART Timing

Figure 18-18 shows the timing waveforms for UART transmit operations. Figure 18-19 shows the timing waveforms for UART receive operations. Figure 18-20 shows the timing waveforms for UART modem operations. Table 18-12 lists the AC timing values for the parameters shown in the three UART figures.

Figure 18-18: UART Transmit Timing

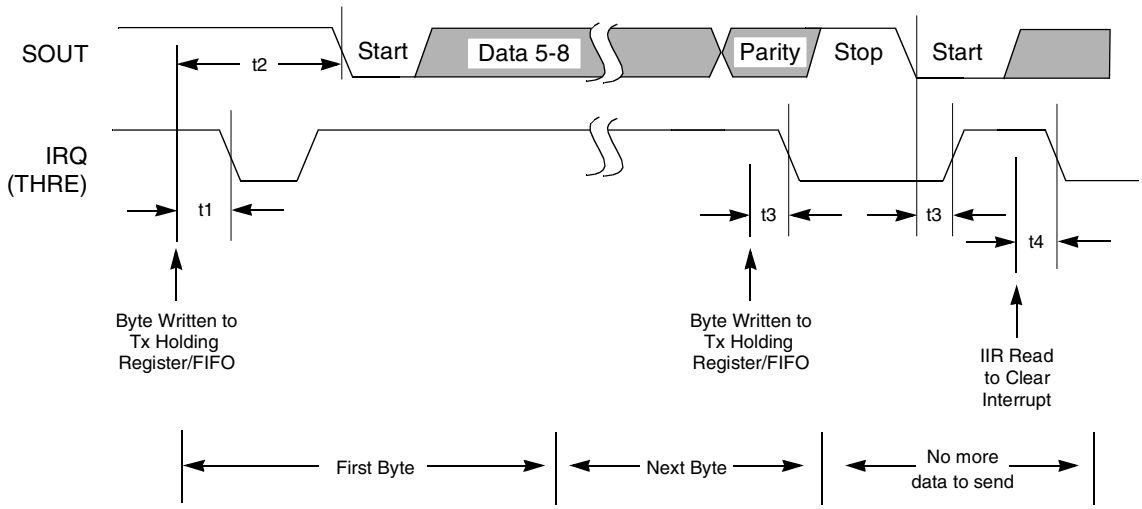


Figure 18-19: UART Receive Timing

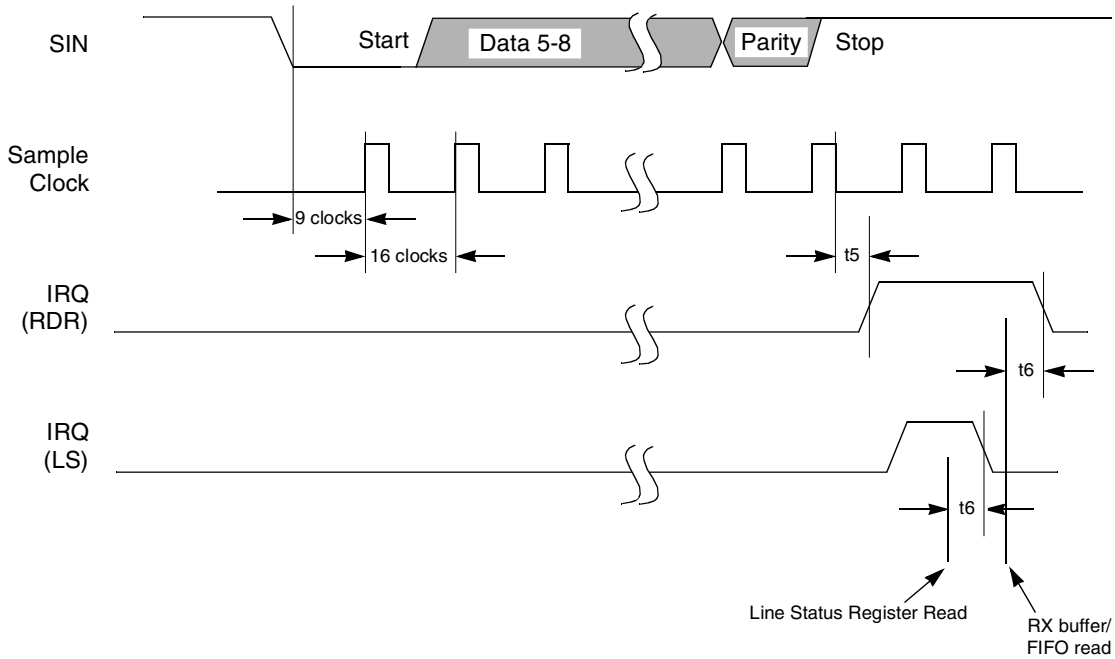


Figure 18-20: UART Modem Timing

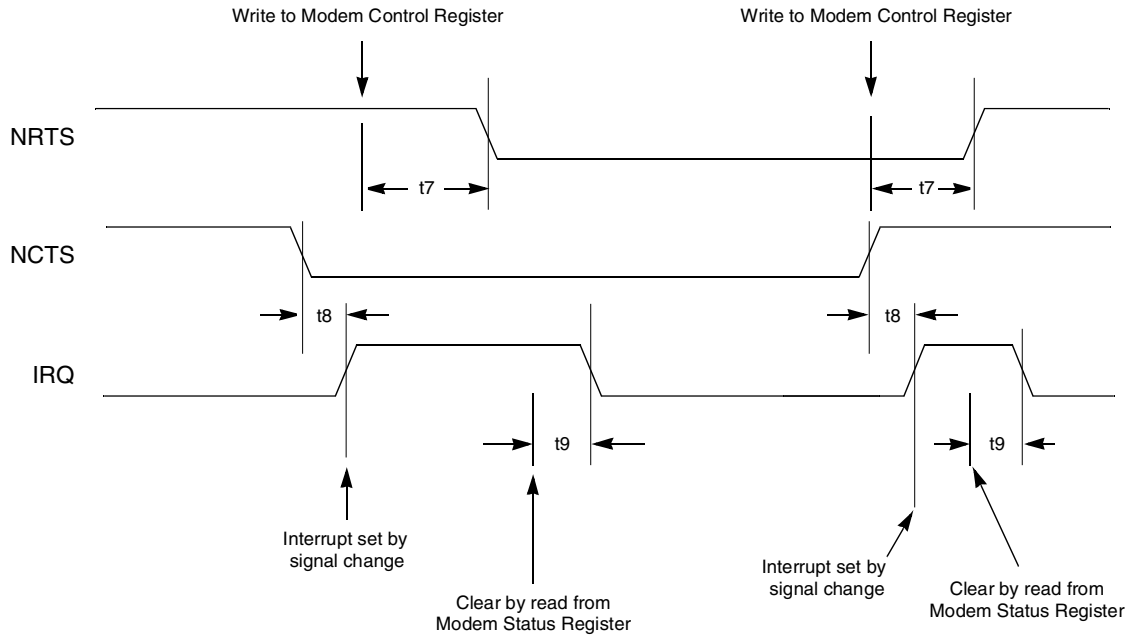


Table 18-12: UART Timing Parameters

Symbol	Parameter	Test Condition	Min	Max	Units
UART Transmit Timing					
t1	Delay time, WR THR (Tx Holding Register) to reset interrupt	CL = 75 pF		50	ns
t2	Delay time, WR THR to transmit start		8	24	baud clocks
t3	Delay time, start to reset interrupt		8	10	baud clocks
t4	Delay time, read IIR (Interrupt Identification Register) to reset interrupt	CL = 75 pF		70	ns
UART Receive Timing					
t5	Delay time, stop to receiver error interrupt or read RBR (RX Buffer Register) to LS (Line Status) interrupt			2	RCLK clocks
t6	Delay time, read RBR/LSR low to reset interrupt low	CL = 75 pF		120	ns
UART Modem Timing					
t7	Delay time, WR MCR (Modem Control Register) to output			60	ns
t8	Delay time, modem interrupt to set interrupt			35	ns
t9	Delay time, RD MSR (Modem Status Register) to reset interrupt			45	ns

AC97-Link and I²S Timing

AC97-Link Timing

Figure 18-21 shows the input and output timing waveforms for the AC-Link interface with respect to BIT_CLK. Figure 18-22 shows the cycle timing waveforms for BIT_CLK and SYNC. Table 18-12 lists the AC timing values for the parameters shown in the two figures.

Figure 18-21: AC-Link I/O Timing

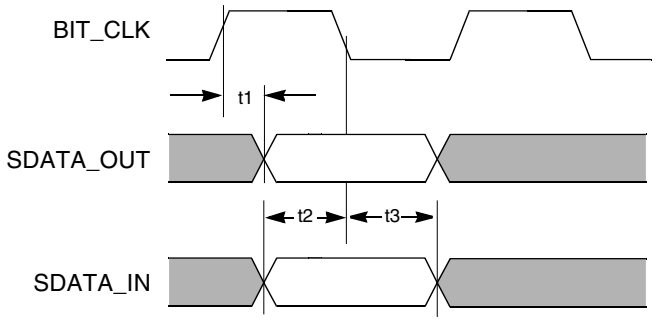


Figure 18-22: AC97-Link Cycle Timing

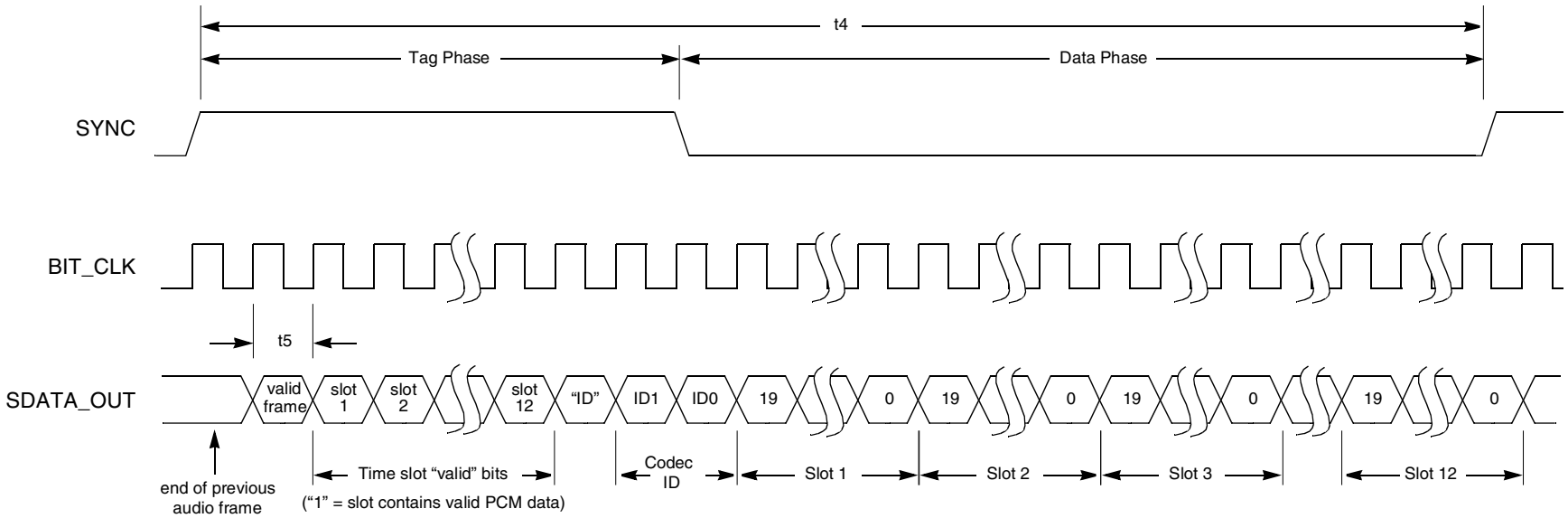


Table 18-13: AC97-Link Timing Parameters

Symbol	Parameter	Min	Typ	Max	Units
I/O Timing					
t1	Output valid delay			15	
t2	Input setup	10			
t3	Input hold	10			
Cycle Timing					
t4	SYNC Cycle Time		20.8		µs
t5	BIT_CLK Cycle Time		81.4		ns

I²S Timing

Figure 18-23 shows the timing waveforms for I²S transmit operations. Figure 18-24 shows the timing waveforms for I²S receive operations. Table 18-14 lists the AC timing values for the parameters shown in the two figures.

Figure 18-23: I²S Transmitter Timing

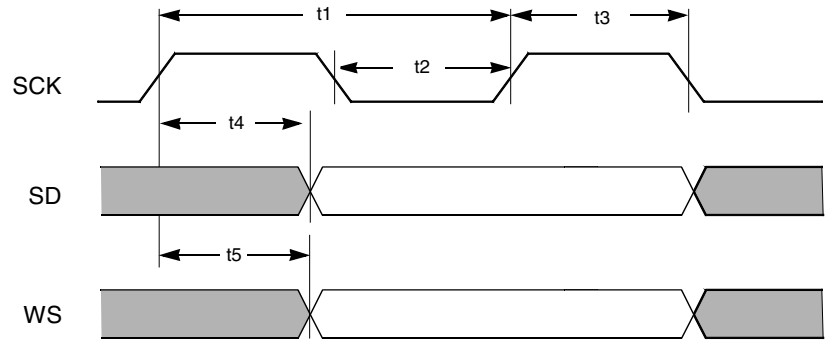


Figure 18-24: I²S Receiver Timing

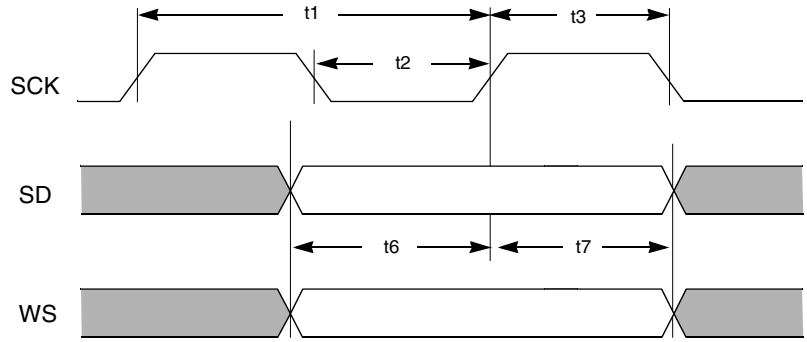


Table 18-14: I²S Timing Parameters

Sym	Parameter	Min	Typ	Max	Unit
t1	Clock period	425	472[1]	519	ns

Table 18-14: I^2S Timing Parameters

Sym	Parameter	Min	Typ	Max	Unit
t2	Clock low	0.35 * t1	165[1]		ns
t3	Clock high	0.35 * t1	165[1]		ns
t4	Transmitter hold time	0			ns
t5	Transmitter delay		377[1]	0.80 * t1	ns
t6	Receiver setup time	0.2 * t1	94[1]		ns
t7	Receiver hold time	0			ns

1. For data rate = 2.1168 MHz.

8051 μ-Controller Timing

This section shows the 8051 waveforms in 12-bit address mode with and without latching of the upper four address bits (AD[11:8]). The 12-bit address mode is enabled by bit 4 of the Mode Select register (see page 12-4). Whether or not latching occurs is determined by bit 26 of the Miscellaneous Control register (see page 2-6). Figure 18-25 and Figure 18-26 show the timing waveforms for 8051 read and write operations with latching, respectively. Figure 18-27 and Figure 18-28 show the timing waveforms for 8051 read and write operations without latching, respectively. Table 18-15 provides the AC timing values for the measurements shown in the figures.

Figure 18-25: 8051 Read Timing (Address High Latched)

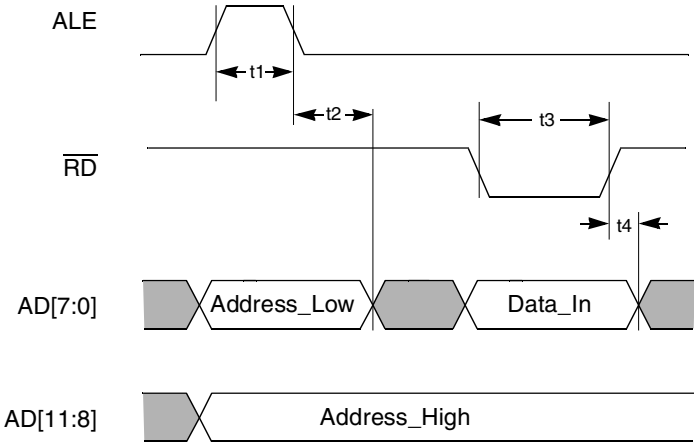


Figure 18-26: 8051 Write Timing (Address High Latched)

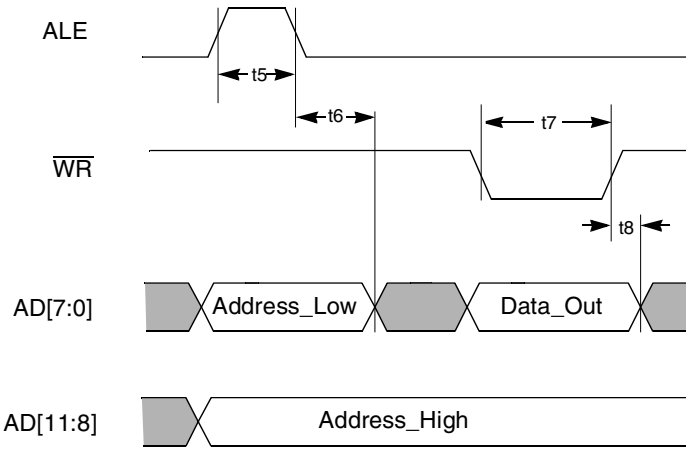


Figure 18-27: 8051 Read Timing (Address High Without Latch)

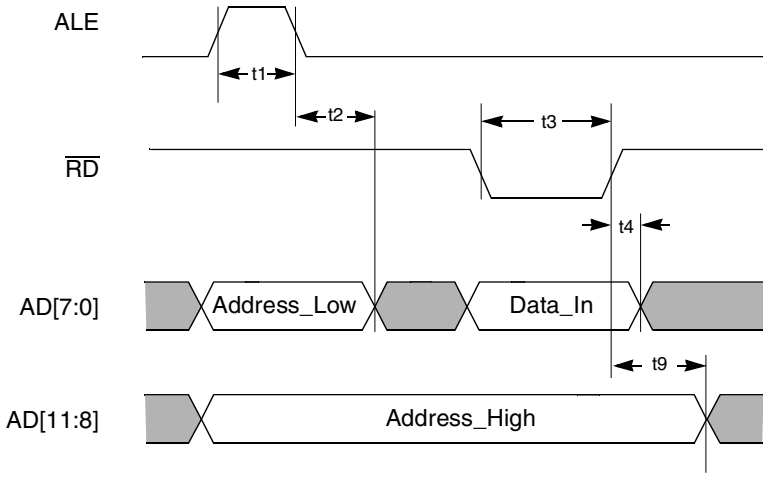


Figure 18-28: 8051 Write Timing (Address High Without Latch)

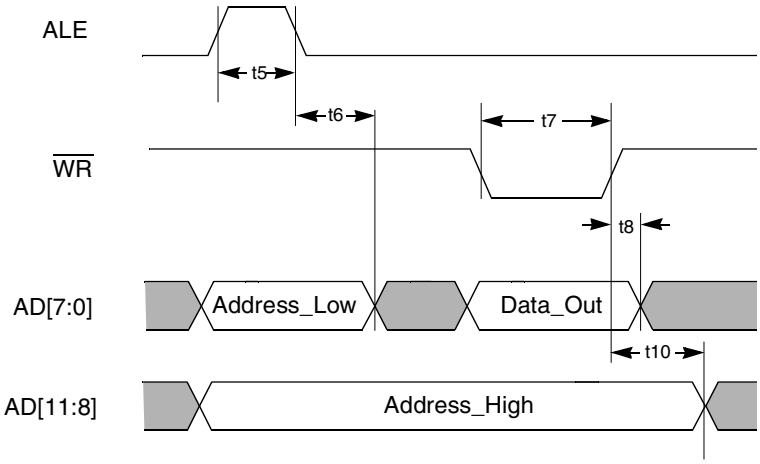


Table 18-15: 8051 Timing Parameters

Symbol	Parameter	Nominal	Units
8051 Read TIming			
t1	ALE width	1	CCLK[1] cycle
t2	Address hold after ALE	1	CCLK cycle
t3	RD width	2	CCLK cycles
t4	Read data hold after RD	0.5	CCLK cycles
8051 Write TIming			
t5	ALE width	1	CCLK cycle
t6	Address hold after ALE	1	CCLK cycle
t7	WR width	2	CCLK cycles
t8	Write data hold after WR	0.5	CCLK cycles
8051 Unlatched Address High Timing			
t9	AD[11:8] hold after RD	1	CCLK cycle
t10	AD[11:8] hold after WR	1	CCLK cycle

1. CCLK is the 8051 clock controlled by bits [1:0] of the Mode Select register (see page 12-4).

Local SDRAM Timing

Figure 18-29 shows the timing waveforms for the Local SDRAM. Table 18-16 lists the AC timing values for the parameters shown in Figure 18-29.

Figure 18-29: Local SDRAM Timing

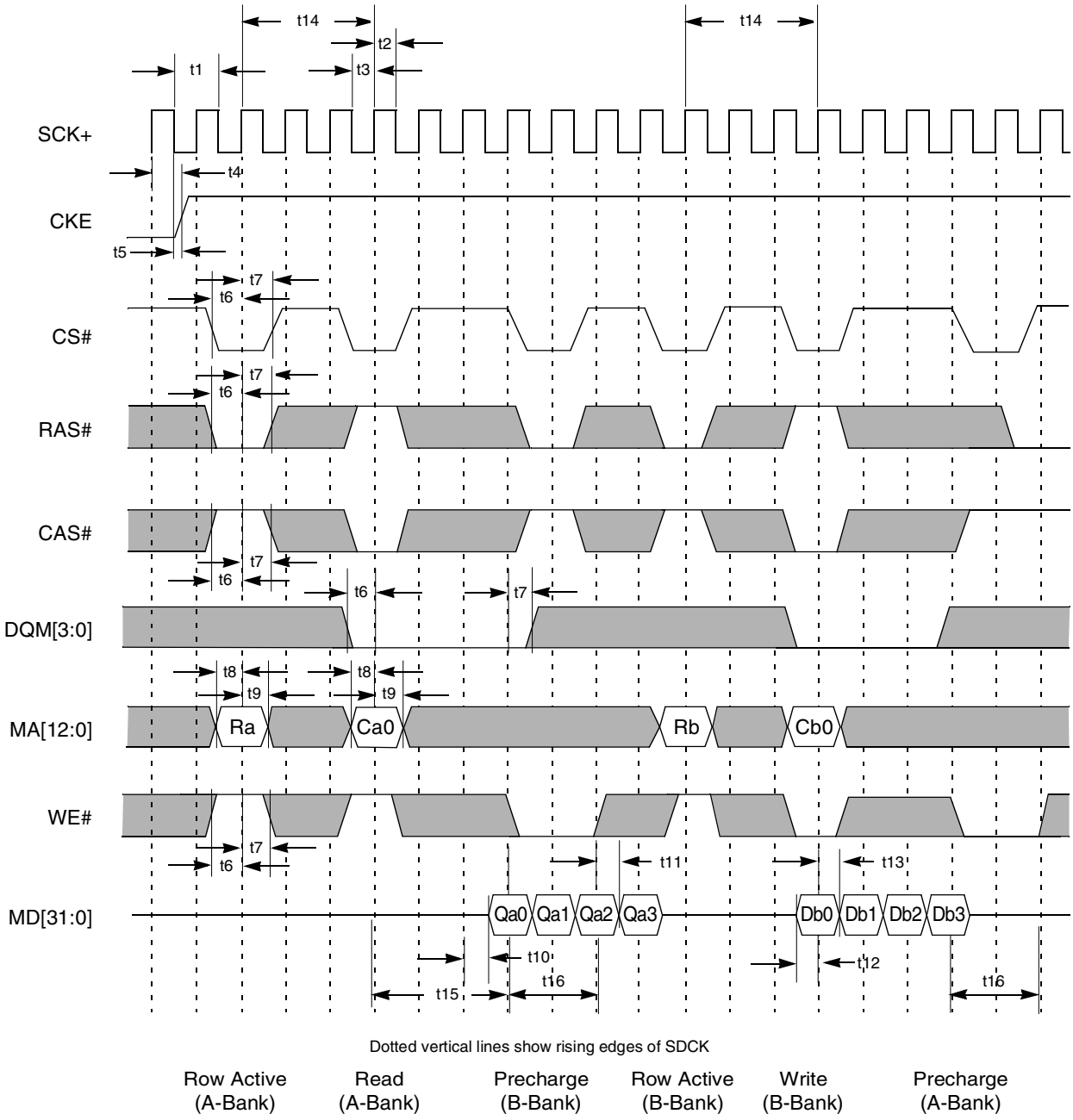


Table 18-16: Local SDRAM Timing Parameters

Symbol	Parameter	Min	Max	Units
t1	SCK+ Cycle Time	6		ns
t2	SCK+ High Time	2.5		ns
t3	SCK+ Low Time	2.5		ns
t4	CKE hold time	0.8		ns
t5	CKE setup time	1.5		ns
t6	Command setup time	1.5		ns
t7	Command hold time	0.8		ns
t8	Address/BA setup time	1.5		ns
t9	Address/BA hold time	0.8		ns
t10	Access time from SCK+		t1 - 2	ns
t11	Data Out hold time from SCK+	1.8		ns
t12	Data In setup time from SCK+	1.5		ns
t13	Data In hold time from SCK+	0.8		ns
t14	Active to READ, WRITE delay (Row Active)	3* t1		ns
t15	READ Latency	3* t1		ns
t16	Write recovery time (Precharge)	2* t1		ns