LH75400/01/10/11 System-on-Chip Preliminary User's Guide

12/8/03

SHARP

Specifications are subject to change without notice.

Suggested applications (if any) are for standard use; See Important Restrictions for limitations on special applications. See Limited Warranty for Sharp's product warranty. The Limited Warranty is in lieu, and exclusive of, all other warranties, express or implied. ALL EXPRESS AND IMPLIED WARRANTIES, INCLUDING THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR USE AND FITNESS FOR A PARTICULAR PURPOSE, ARE SPECIFICALLY EXCLUDED. In no event will Sharp be liable, or in any way responsible, for any incidental or consequential economic or property damage.

LH75400/01/10/11 System-on-Chip Preliminary User's Guide
Produced by the SHARP Microelectronics of the Americas Technical Publication Group.

Reference No. SMA02058

Content Revisions

This document contains the following changes to content, causing it to differ from previous versions.

Record of Revisions

DATE	PAGE NO.	SECTION, TABLE, OR ILLUSTRATION	SUMMARY OF CHANGES
11-15-03	1.5	1.5.1	Pull-up resistor value of 10 KΩ specified for LINREGEN.
	1-7	1.7	Text corrected to reflect connectable crystal range of 14 - 20 MHz, plus frequency yield of PLL.
	6-2	Table 6-1	Numbers in Table heads and Notes clarified.
	6-4	Table 6-4	'HR-TFT Controller' replaced with 'Advanced LCD Interface'
	7-13+	Register Bit Fields	Where Registers contain Read Only and Read/Write bits, Reserved bits now read 'Write the Reset Value'.
	9-4	9.2.2	Crystal range connectable to XTALIN and XTALOUT clarified.
	9-7	Table 9-3	Bits 6:5 corrected to R/W, Write the Reset Value.
	9-9	Table 9-10	Bits 15:0 corrected to 'Do not write any other value to these bits.'
	9-13	Tables 9-18, 19	Bits 8:3 corrected to R/W, Write the Reset Value.
	9-14	Tables 9-20, 21	Bits 15:2 Reset values corrected.
	9-15	Tables 9-22, 23	Bit 1 corrected to R/W, Write the Reset Value.
	9-18	Tables 9-30, 31	Bits 15:14 corrected to R/W, Write the Reset Value.
	9-20	Tables 9-32, 33	Bit 7 corrected to W, Write only 0.
	9-20	Table 9-32	Register address space corrected.
		Chapter 13	Rewritten to clarify operation of Advanced LCD Interface, formerly known as the HR-TFT Controller. Old register names are retained as notes. Register functions and formulae are clarified, but there are no changes to actual operation.
	13-4	13.3	Caution statement added regarding enabling LCD before initializing buffer space when REMAP = 1.
	18-12	18.5.2.1	SCR bit description clarified.
	18-16	18.5.2.5	CPSR Register clarified.
	19-4	19.2.3	UART0 and UART1 DMA capabilities clarified.
	23-13	23.3.2.4	Bit 5, Interrupt Mask Register is Reserved, R/W, Reset Value 0, Write the Reset Value.
	23-26	23.3.3	Text and formulas clarified.

Table of Contents

Chapter 11 – I/O Configuration

Chapter 12 – Direct Memory Access Controller

Chapter 13 – Color Liquid Crystal Display Controller

Chapter 16 – Watchdog Timer

Chapter 17 – Real-Time Clock

Chapter 18 – Synchronous Serial Port

List of Figures

Chapter 13 – Color Liquid Crystal Display Controller

Chapter 14 – Liquid Crystal Display Controller

Chapter 15 – Timers

Chapter 16 – Watchdog Timer

Chapter 17 – Real-Time Clock

Chapter 18 – Synchronous Serial Port

Chapter 19 – UART0 and UART1

Chapter 20 – UART2

Chapter 21 – General Purpose Input/Output

Chapter 22 – Controller Area Network

Chapter 23 – Analog-to-Digital Converter/Brownout Detector

Chapter 25 – Recommended Layout Practices

List of Tables

Chapter 9 – Reset, Clock, and Power Controller

Chapter 10 – Vectored Interrupt Controller

Chapter 11 – I/O Configuration

Chapter 12 – Direct Memory Access Controller

Chapter 13 – Color Liquid Crystal Display Controller

Chapter 14 – Liquid Crystal Display Controller

Chapter 15 – Timers

Chapter 16 – Watchdog Timer

Chapter 17 – Real-Time Clock

Chapter 18 – Synchronous Serial Port

Chapter 19 – UART0 and UART1

Chapter 20 – UART2

Chapter 21 – General Purpose Input/Output

Chapter 22 – Controller Area Network

Chapter 23 – Analog-to-Digital Converter/Brownout Detector

Chapter 24 – LCD Pin Multiplexing

Chapter 26 – Register Map

Preface

This User's Guide describes the SHARP BlueStreak LH75400/01/10/11 16/32-bit ARM-based System-on-Chip (SoC) devices. These four devices are:

- LH75400
- LH75401
- LH75410
- LH75411.

This User's Guide is the principal technical reference for these SoCs. It is intended for engineers responsible for designing, integrating, programming, and testing embedded systems based on these four SHARP SoCs. Electrical Characteristics are found in the Data Sheet.

This User's Guide assumes that the reader:

- Is familiar with the basic principles and techniques of embedded-microprocessor system engineering
- Has familiarity with programming the ARM7TDMI-S Processor
- Has access to documentation, support, and training available from ARM.

NOTE: This User's Guide does not cover the ARM instruction set. For information about ARM instruction set, and programming the ARM7TDMI-S processor, visit the ARM Web site at www.arm.com.

Supplemental Documentation

An abridged version of this User's Guide (and containing the Electrical Characteristics), is available as a Data Sheet. An even more abbreviated version is available as a single-page Product Brief. Please contact your local SHARP representative for details, or visit the SHARP Microelectronics of the Americas Web site at www.sharpsma.com.

Terms and Conventions

This User's Guide describes four devices that differ only in feature sets. In referring to these devices collectively, the term 'LH75400/01/10/11' becomes clumsy and difficult to read, so this Guide uses the terms 'SoC' and 'device' to describe this set of System-on-Chip devices.

For information about specific terms and acronyms, see the Glossary in this User's Guide.

Multiplexed Pins

These devices are manufactured in a 144-pin Low Profile Quad Flat Pack (LQFP) package. Some pins have only one function, but others are multiplexed and may carry as many as three functions. Although more than one function can be assigned to multiplexed pins, the pins support only one function at a time.

Pin Names

Package pins are named to indicate the signal(s) or functionality available at the pin. If the signal or function is active LOW, the name is prefixed with a lower-case 'n', such as nCS2. Multiplexed pins are named to indicate all available functions, such as Pin 30: PB0/nCS1.

These naming conventions help designers recognize and avoid collisions between multiplexed functions but can complicate explanatory text, so this Guide uses the name appropriate to the context. A discussion of chip selects, for example, would refer to signal nCS1, but information about PortB, bit 0 would use PB0. Readers must be aware that these are separate signals, with distinctly different functionality, which happen to be available on the same pin. Such signals are not simultaneously available at the pin.

Peripheral Devices

These SOCs are built using the ARM7TDMI-S RISC core as a base. Objects within the chip but external to the core processor and its support devices are referred to throughout this Guide as 'Peripheral Devices'.

All four SoCs include two buses:

- An Advanced High-Performance Bus (AHB)
- An Advanced Peripheral Bus (APB).

The devices shown on the APB in the block diagrams are an example of 'peripheral devices' in this document. Devices that are external to the chip are referred to as 'external devices'.

Register Names

In this User's Guide, the terms 'mask' and 'enable' may be used within register names. Registers may also be named for one function, and at the same time contain or control other functions.

Register Addresses

The SoCs are memory-mapped with programmable, internal registers that control its operation. Each internal register is located at a unique address in the memory map and the registers are generally grouped in the map by subsystem. In this Guide, the addresses for all registers are expressed as a base address. The base address indicates where in the map a group of registers begins.

Register Tables

All registers are presented in tabular format. A primary table presents each register's name, address, permissions, bit-field names, and the register's contents at reset. Subsequent tables detail the specific function(s) of all bit fields in the register and explain any important variations that may exist. See Table 1 and Table 2 for examples.

Table 1. Register Name

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	F31	F30	F29	F28	F27	F26	F25	F24	F23	F22	F21	F20	F19	F18	F17	F16
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	F15	F14	F13	F12	F11	F10	F09	F08	F07	F06	F05	F04	F03	F02	F01	F00
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	REGISTERBASE															

Table 2. Bit Definitions

BITS	FIELD NAME	DESCRIPTION
31:0	F31:F0	A description of these bits' functionality

Numeric Values

Binary values are prefixed with 0b, as in 0b00001000. Binary values may also be shown within quotation marks, as in '0'.

Hexadecimal values are expressed with UPPER CASE letters and prefixed with 0x, as in 0x0FBC.

All numeric values not specifically identified with the above prefixes as either binary or hexadecimal are decimal values.

Block Diagrams

The functional descriptions in this User's Guide include block diagrams with symbols representing logical or mathematical operations or selections, usually the result of writing a value to a register. Figure 1 shows an example of a multiplexer with three inputs and one output (the result).

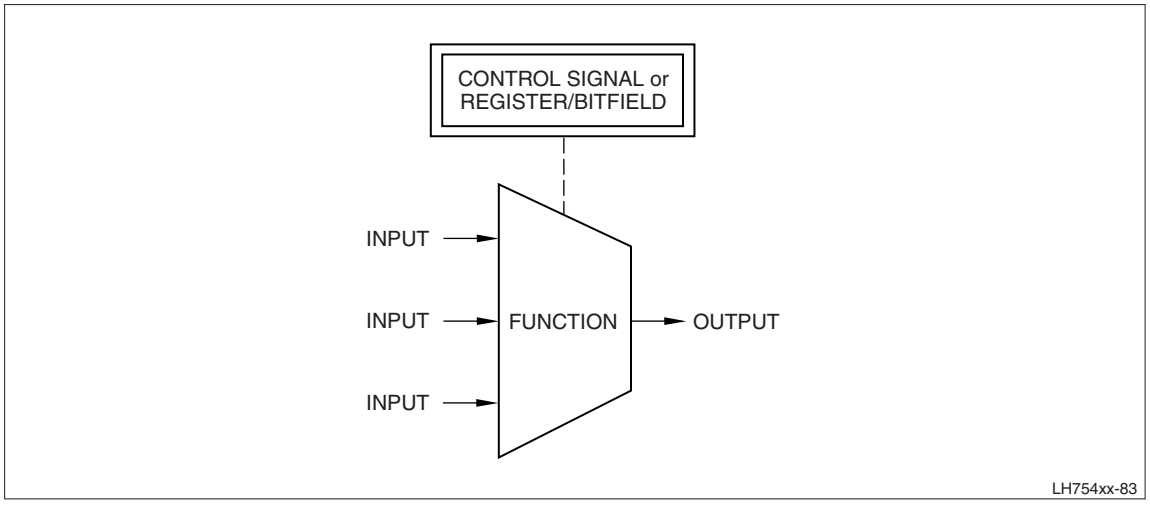


Figure 1. Multiplexer

Block diagrams can include symbols representing registers and the bit fields within them. Figure 2 shows that the BITFIELDNAME bit field in the REGISTERNAME register enables or disables the signal named OUTPUT.

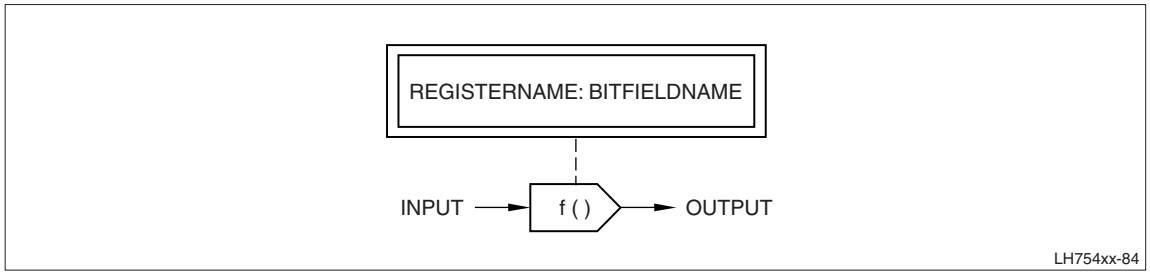


Figure 2. Register with Bit Field Named

Figure 3 is similar to Figure 2, except that Figure 3 references multiple (different) BITFIELDS in the REGISTERNAME register.

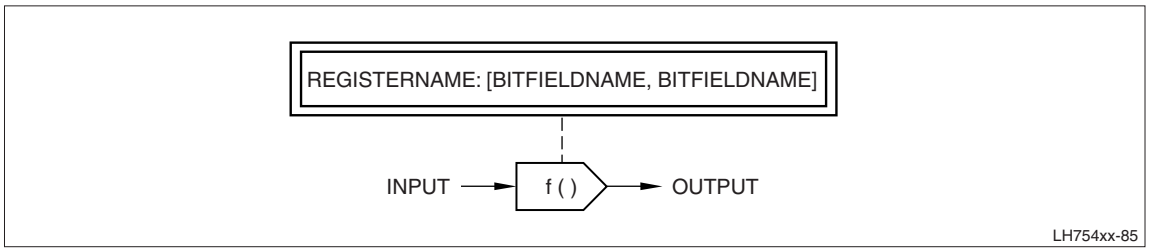


Figure 3. Register with Multiple Bit Fields Named

Not all bit fields are named. If a bit field has no name, the register is shown with numbers indicating the appropriate bit-positions, with the least-significant bit on the right, as in Figure 4. This bit-ordering matches that of the register tables, shown in Table 1.

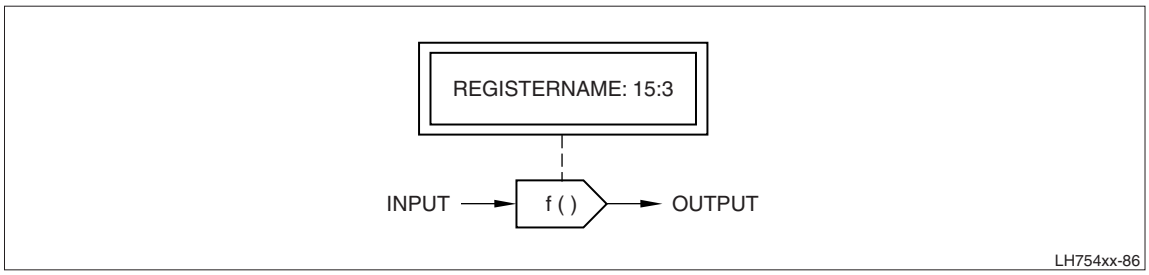


Figure 4. Register with Bit Field Named

Chapter 1
Introduction

1.1 Product Overview

LH75400/01/10/11 is a family of four 16/32-bit System-on-Chip (SoC) devices:

- LH75401 — contains the superset of features. See Chapter 2.
- LH75411 — similar to the LH75401, without the Controller Area Network (CAN) 2.0B interface. See Chapter 3.
- LH75400 — similar to the LH75401, but with a Grayscale Liquid Crystal Display (LCD) Controller only. See Chapter 4.
- LH75410 — similar to the LH75400, without the CAN 2.0B interface. See Chapter 5.

All four devices are low-cost solutions that use the ARM7TDMI-S core integrated with a wide variety of functions. All operate up to 51.6096 MHz at 3.3 V and come in a 144-pin Low-profile Quad Flat Pack (LQFP) package.

NOTE: In this User's Guide, the nonspecific term 'SoC' or 'device' refers collectively to all four SoCs. Where information pertains to certain SoCs only, those devices are clearly identified.

Table 1-1. Feature Summary

FEATURE	LH75401	LH75411	LH75400	LH75410
Static Random Access Memory Controller	X	X	X	X
Static Memory Controller	X	X	X	X
Direct Memory Access Controller	X	X	X	X
Color Liquid Crystal Display Controller	X	X		
Liquid Crystal Display Controller			X	X
UART0 and UART1	X	X	X	X
UART2	X	X	X	X
Timers	X	X	X	X
Real-Time Clock	X	X	X	X
Controller Area Network	X		X	
Analog-to-Digital Converter/Brownout Detector	X	X	X	X
Synchronous Serial Port	X	X	X	X
Watchdog Timer	X	X	X	X
Vectored Interrupt Controller	X	X	X	X
Reset, Clock and Power Controller	X	X	X	X
Operating Modes	X	X	X	X
I/O Configuration	X	X	X	X
General Purpose I/O	X	X	X	X

1.2 ARM and Thumb State

These SoCs consist of a 32-bit core processor with a 16-bit data bus that can operate in ARM Thumb mode for executing 16-bit instructions. Thumb is an extension to the ARM architecture. It contains 36 instruction formats drawn from the standard 32-bit ARM instruction set that have been re-coded into 16-bit-wide opcodes. On execution, the SoCs decompresses the 16-bit Thumb opcodes to its ARM instruction set equivalents, which are then run normally.

Unlike processors that offer a mixed instruction set, these devices support Thumb-code and ARM-code as two separate instruction sets. The fact that the two instruction sets are separate means that decoding logic is extremely simple. This, in turn:

- Keeps silicon area small
- Maintains low power and MIPS/Watt performance
- Allows designers to keep their ARM 32-bit instruction set, while benefiting from the code-size advantages of the Thumb instruction set.

The combination of the two instruction sets running on a single Thumb-aware core makes these devices effective solutions to the code-size and performance problems of 16-bit systems. And since the thumb-aware core is simply an extension of the ARM architecture, designers can:

- Compile for Thumb-code, ARM-code, or a mix of both
- Retain 32-bit RISC performance.

1.3 Bus Architecture

The SoCs employ the ARM Advanced Microcontroller Bus Architecture (AMBA) 2.0 internal bus protocol. Each device has three AHB masters on the AHB that control access to the external memory and the on-chip peripherals:

- An ARM processor to fetch instructions and transfer data
- A DMA Controller to transfer between memory and UART0, UART1, an external peripheral, or memory
- An LCD Controller to refresh an LCD panel with data from the external memory or from Tightly Coupled Memory if the frame buffer is 16 Kb or less.

The ARM7TDMI-S processor is the default bus master. Table 1-2 lists the priorities for the three AHB masters. These levels are fixed and cannot change.

Table 1-2. Bus Master Priority

PRIORITY	BUS MASTER PRIORITY
1 (highest)	CLCD Controller (LH75401 and LH75411) LCD Controller (LH75400 and LH75410)
2	DMA Controller
3 (lowest)	ARM7TDMI-S Core (default)

An APB bridge is used to provide access to the various APB peripherals:

- Analog-to-Digital Converter
- Controller Area Network (LH75401 and LH75400 only)
- Counter/Timers
- General Purpose Input/Output
- I/O Configuration
- Real Time Clock
- Reset, Clock, and Power Controller
- Synchronous Serial Port
- UARTs
- Watchdog Timer.

Generally, APB peripherals are serviced by the ARM core. This arrangement maximizes system performance by allowing the DMA Controller to transfer data while the ARM core executes from Tightly Coupled Memory (TCM).

1.4 Operating Modes

The SoCs support three operating modes:

- Normal Mode
- PLL Bypass Mode
- Embedded ICE Mode.

The operating mode that the SoC enters at Power-on Reset is determined by the state of the TEST1, TEST2, and nRESETIN signals. Table 1-3 shows the signal states that correspond to each operating mode.

The TEST1, TEST2, and nRESETIN signals are latched on the rising edge of nPOR. The device remains in that operating mode until power is removed or nPOR transitions from LOW to HIGH.

Table 1-3. Device Operating Modes

OPERATING MODE	TEST2	TEST1	nRESETIN
Reserved	0	0	0
PLL Bypass	0	0	1
Reserved	0	1	x
Reserved	1	0	0
Embedded ICE	1	0	1
Normal	1	1	x

1.4.1 Normal Mode

As its name implies, Normal Mode is the mode in which the SoC is placed for normal operation. The system clock is generated from the PLL. In this mode, the JTAG interface is active and accesses the boundary scan TAP Controller. This TAP Controller controls the boundary scan cells that exist around the periphery of the device.

The LH75400/01/10/11 device JTAG ID Code is:

```
[31:28] Version (0x0)
[27:12] Part Number (0x0754)
[11:1] Manufacturer's ID (0x030)
[0] (0x1)
```

This JTAG ID Code is equivalent to 0x0754061.

1.4.2 PLL Bypass Mode

The SoC can be configured to bypass its internal PLL and operate with an external clock source. In this mode:

- The PLL is placed in Bypass Mode
- The lock output from the PLL is forced HIGH
- XTALIN becomes the direct clock input to the RCPC, replacing the PLL output.

The system clock can be scaled down from divide by 30 to divide by 2.

During PLL Bypass Mode, the TEST2 pin must remain LOW until nPOR is asserted and the operating mode is changed.

1.4.3 Embedded ICE Mode

In Embedded ICE Mode, the JTAG port accesses the TAP Controller in the ARM7 core and the ARM7 core is placed in Debug Mode.

1.5 Power Supplies

The core logic requires a 1.8 V supply. 5 V-tolerant, 3.3 V I/Os, requiring a 3.3 V supply, are employed. These devices are designed to require a single 3.3 V supply. A 3.3 V-to-1.8 V linear regulator is integrated into the chip to be used to generate the 1.8 V needed by the core logic.

- Nine pairs of power pins (VDD and VSS) are dedicated to the 3.3 V supply.
- Two pairs of power pins are dedicated to the 1.8 V supply.
- One pair of power pins is dedicated to the analog circuitry of the A-to-D converter.
- One pair of power pins is dedicated to the PLL and crystal oscillators.

1.5.1 Linear Regulator Power

When the linear regulator is enabled, the 1.8 V power pins (VDDC) are outputs of the regulator. This allows regulator operation to be verified. An external low ESR capacitor must be tied to the regulator output for stability. If the regulator is disabled, the 1.8 V power pins are used as inputs and an external 1.8 V supply must be provided.

The linear regulator provides an ENABLE input that is tied to the LINREGEN pin. The linear regulator is enabled by holding LINREGEN HIGH, and is disabled by holding LINREGEN LOW. The external pull-up resistor on LINREGEN must be 10 kΩ or less.

Proper power-up sequencing for the SoC must be considered when employing the linear regulator. In order to ensure this takes place, nPOR must be held LOW until the linear regulator has ramped up to an acceptable operating voltage.

Table 1-4 lists the linear regulator ramp-up time.

Table 1-4. Linear Regulator Ramp-up Time

DESCRIPTION	TYP. (µs)	MAX. (µs)
Linear regulator stabilization time after power-up		200
PLL stabilization time after power-up	8.57143	10

1.5.1.1 PLL Power

The PLL requires a 1.8 V supply.

- If the linear regulator is disabled, 1.8 V ±0.18 V must be supplied to pin 85.
- If the linear regulator is enabled, the PLL power supply should come from the VDDC output(s). In this instance, connect VDDA_PLL to VDDC through the filter.

1.5.1.2 PCB Mounted Analog Power Supply Filter for PLL Usage

Ideally, an Analog Power Supply Filter — a low-pass filter with -3 dB at < 1 kHz and < -70 dB in the absorption band — should be used. However, real-life components limit the -3 dB point. A good board layout is vital to achieving good high-frequency absorption. An R-C or R-L-C filter is usually used, with the 'C' composed of multiple devices to achieve a wide spectrum of noise absorption. For DC reasons, the series resistance of this filter is limited; generally, a < 5% voltage drop across this device should be observed under worst-case conditions. High-quality series inductors should not be used without a series resistor; otherwise, a high-gain series resonator is created.

To achieve the low-frequency cut off, the design needs an electrolytic capacitor in the filter. As the filter also needs to sustain its attenuation into high frequencies, (e.g., > 100 MHz), the design needs at least one non-electrolytic capacitor in parallel. The leads of the high-frequency capacitor(s) must be kept short.

Board layout around this high-frequency capacitor, and the path to the pads, is critical. It is vital that the quiet ground and power are treated like analog signals.

The power (VDDA_PLL) path must be a single wire that runs:

- From the IC package pin to the high-frequency capacitor
- Then to the low-frequency capacitor
- Then through the series element (e.g., resistor)
- Then to board power (VDDC).

The distance from the IC pin to the high-frequency capacitor should be as short as possible.

Similarly, the ground (VSSA_PLL) path should run from the IC pin to the high-frequency capacitor, then to the low-frequency capacitor. The distance from IC pin to high-frequency capacitor should be very short.

The PLL has the DC ground connection made on chip. Therefore, the external VSSA_PLL pin must be connected to the power supply filter only, not to PCB ground.

To minimize noise, especially non-common-mode noise, power and ground traces should be run as closely and as parallel as possible, with large spacings to adjacent traces, for all applications. The leads of the high-frequency capacitor must be kept short; this includes:

- Board wires
- Vias
- Capacitor wires
- Wires within the capacitor package.

Therefore, select components carefully. The area and impedance of the power loop must be minimized, where the loop includes the high-frequency capacitor and VDDA_PLL and VSSA_PLL board traces to the IC. The board layout should have the smallest total analog power circuit, with short and adjacent wire traces. Extra connections should not be made to board power planes; the only connections should be those described above.

1.5.2 Real-World Component Selection

Throughout the attenuating frequency range, there should be no resonant nonabsorptions. This means the series element will be either a resistor or a very poor (i.e., resistive) inductor.

Using the series element with the greatest impedance possible (e.g., 100 Ω), the electrolytic used is usually the largest capacitance tantalum that fits nicely on the board (e.g., 25 μF). Similarly, the other capacitor is the highest value HF capacitor that can be found in a small package (e.g., 100 nF).

1.6 Crystal Oscillator Usage

When a chip containing a crystal oscillator is used on a board, the user must make some resistance and capacitance connections from the chip to the board.

Figure 1-1 shows an application diagram for the crystal oscillator.

Rfb is a feedback resistor, and C1 and C2 are load capacitances. To determine the appropriate values for the system, a good starting point is 1 MΩ for Rfb, 10 pF for C1, and 10 pF for C2. The values of Rfb, C1, and C2 may be further refined to meet the frequency requirements of the system.

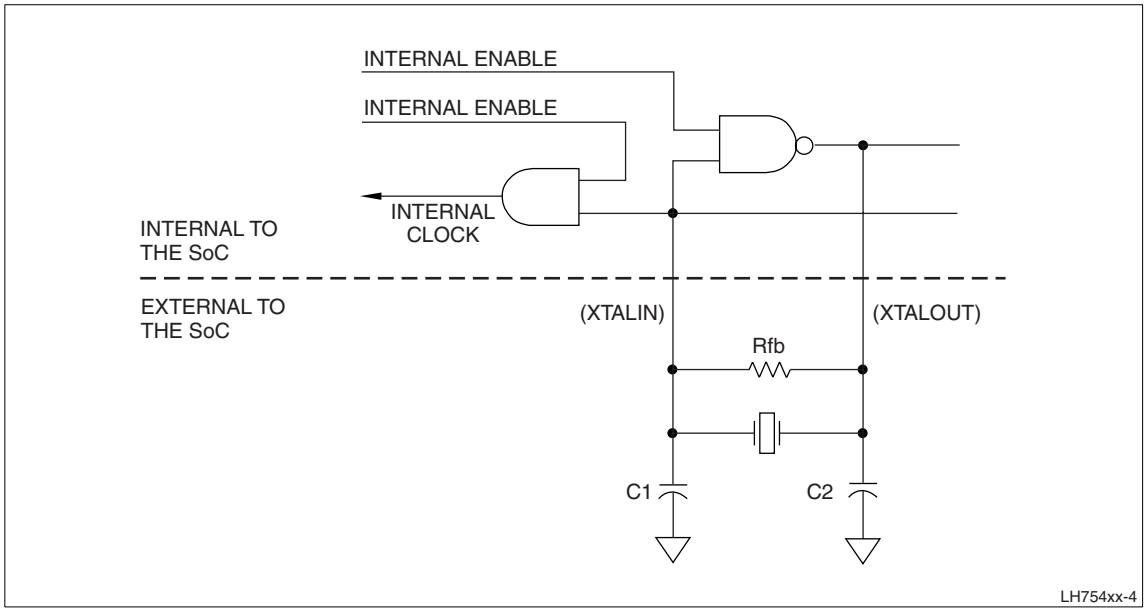


Figure 1-1. Crystal Oscillator

1.7 Clocking Strategy

The SoCs provide for two crystal oscillators.

- The first oscillator drives an internal Phased Lock Loop (PLL) and the three UARTs.

The internal PLL's input is taken from the crystal oscillator, which has a useful range of 14 MHz to 20 MHz. This input frequency is multiplied by seven by the PLL. The resulting output frequency range is 98 MHz to 140 MHz. The Core clock (HCLK) divisor is selectable; but at the minimum divisor of 2, this yields an HCLK of 70 MHz.

- The second oscillator is a 32.768 kHz oscillator that generates a 1 Hz clock for the RTC.

The frequency ranges of the PLL and crystal oscillator together make for a crystal frequency range of operation that is from 14 MHz to 20 MHz. However, since the UART clocks are driven by the crystal oscillator, an oscillator frequency of 14.7456 MHz is recommended for this design (but not required). This frequency can be divided down to the exact frequencies that a UART needs to achieve modem baud rates. This creates a PLL output frequency of approximately 103.2192 MHz. The system clock frequency can be set up to be from divide-by-30 to divide-by-2 (3.44 MHz to 51.6096 MHz using a 14.7456 MHz crystal) in decrements of two (30, 28, 26, 24…) of the PLL frequency. These devices are designed to have a maximum operating frequency of 51.6096 MHz. If UART0 and UART1 are to be used, the system clock frequency must not be set any lower than 3/5 of the frequency applied to the crystal input pin (XTALIN) for proper UART operation.

The SoCs can be configured to operate via an external clock source, bypassing the internal PLL. This is done by holding TEST2 and TEST1 LOW and nURESET HIGH while nPOR is active. These signals are latched on the rising edge of nPOR. During this mode TEST2 must remain LOW, this keeps the PLL in Bypass Mode and XTALIN becomes the direct clock input. In this mode, the system clock frequency can be set up to be from divide-by-30 to divide-by-2 in decrements of two (30, 28, 26, 24…) of the XTALIN frequency. It can have a maximum system clock frequency of 51.6096 MHz and a minimum of zero, as this is a static design. The device remains in PLL Bypass Mode until power is removed or nPOR transitions from LOW to HIGH again.

1.8 Reset Strategy

Two external signals, nPOR and nRESETIN, generate resets to the SoCs. If nPOR is asserted, all internal registers are set to their default state. It is intended to be used as a Power-On Reset only. If nRESETIN is asserted, all internal registers EXCEPT the JTAG circuitry within the devices is set to its default state. The amount of time that Power-On Reset should be held LOW (crystal stabilization time plus 200 μs) varies, depending on the crystal used.

While nPOR is asserted, nRESETIN defines the Test Mode, if any, into which the devices are placed. Once nPOR is released, nRESETIN behaves during Reset as described previously. For more details, see Chapter 9, Reset, Clock, and Power Controller.

The SoCs can generate two types of Internal Resets: System Reset and RTC Reset. A System Reset refers to an nPOR Reset, an nRESETIN Reset, a Software Reset, or a Watchdog Timer Reset. For more details about a Watchdog Timer Reset, see Chapter 9. As previously stated, there are two types of Software Resets. Either type causes a System Reset. Only one type causes an RTC Reset, which is the only way to reset the RTC. For more information about Software Resets, see Chapter 9.

The System Reset is brought out to an external pin (nRESETOUT). nRESETOUT is held asserted for eight system clock cycles following the release of the reset causing the System Reset.

Chapter 2
LH75401 SoC

2.1 LH75401 Features

- ARM7TDMI-S™ Core
- Up to 70 MHz System Clock (HCLK)*
  - Internal PLL Driven or External Clock Driven
  - Crystal Oscillator/Internal PLL Can Operate with Input Frequency Range of 14 MHz to 20 MHz.
- 32KB On-chip SRAM
  - 16KB Tightly Coupled Memory (TCM) SRAM
  - 16KB Internal SRAM.
- Clock and Power Management
  - Low Power Modes: Standby, Sleep, Stop.
- Eight Channel, 10-bit Analog-to-Digital (A/D) Converter
- Integrated Touch Screen Controller
- Four DMA Channels
- Color and Grayscale LCD Controller
  - 12-bit (4096) Direct Mode Color, up to VGA (640 × 480)
  - 8-bit (256) Direct or Palletized Color, up to SVGA (800 × 600 DPI)
  - 4-bit (16) Direct Mode Color/Grayscale, up to XGA (1,024 × 768 DPI)
  - 12-bit Video Bus
  - Supports Supertwist Nematic (STN), Thin Film Transistor (TFT), High Reflective TFT (HR-TFT), and Advanced TFT (AD-TFT) Displays.
- Controller Area Network (CAN) Controller that supports CAN version 2.0B
- Serial Interfaces
  - Two 16C550-type UARTs
  - One 82510-type UART.
- Synchronous Serial Port (SSP)
  - Motorola SPI™
  - National Semiconductor Microwire™
  - Texas Instruments SSI.
- Real-Time Clock (RTC)
- Three Counter/Timers
  - Capture/Compare/PWM Compatibility
- Watchdog Timer (WDT)
- Low-Voltage Detector

- JTAG Debug Interface and Boundary Scan
- Single 3.3 V Supply
- 5.0 V Tolerant Inputs
- 144-pin LQFP Package
- -40°C to +85°C Operating Temperature.

NOTE: *70 MHz Operation requires the use of the internal 3.3 V-to-1.8 V internal linear regulator.

2.2 LH75401 Block Diagram

Figure 2-1 shows a block diagram of the LH75401. For information about the blocks shown in this figure, see the appropriate chapters in this User's Guide.

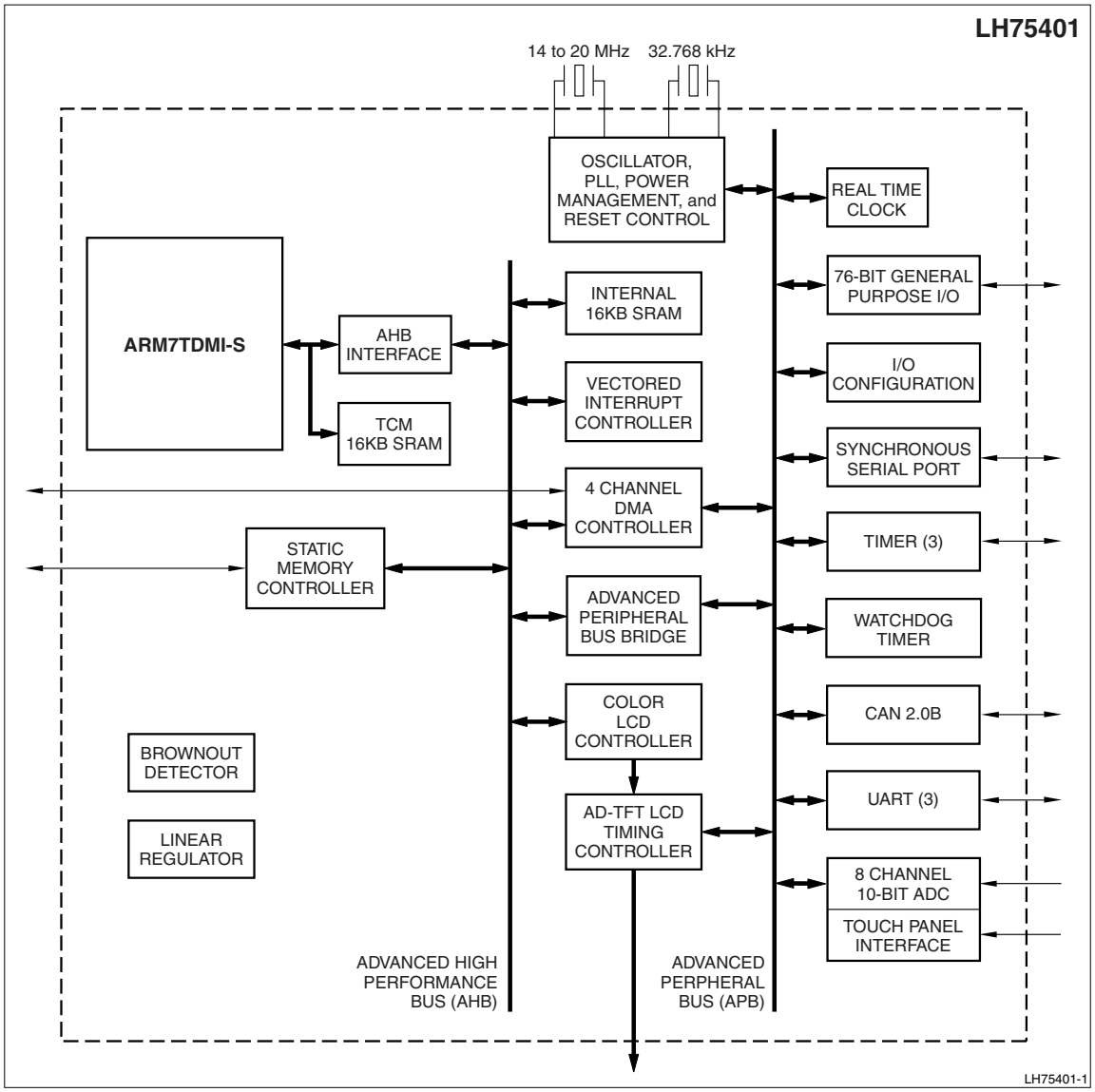


Figure 2-1. LH75401 Block Diagram

2.3 LH75401 Applications

The SHARP LH75401 is ideally suited for a variety of applications, including white-goods and industrial-control.

Typical white-goods applications include, but are not limited to:

- Air conditioners
- Washing machines
- Refrigerators
- Cooking equipment.

Typical industrial-control applications include, but are not limited to:

- Measuring instruments
- Machine-control systems
- Programmable Logic Controllers.

Figure 2-2 shows a system application example for the LH75401.

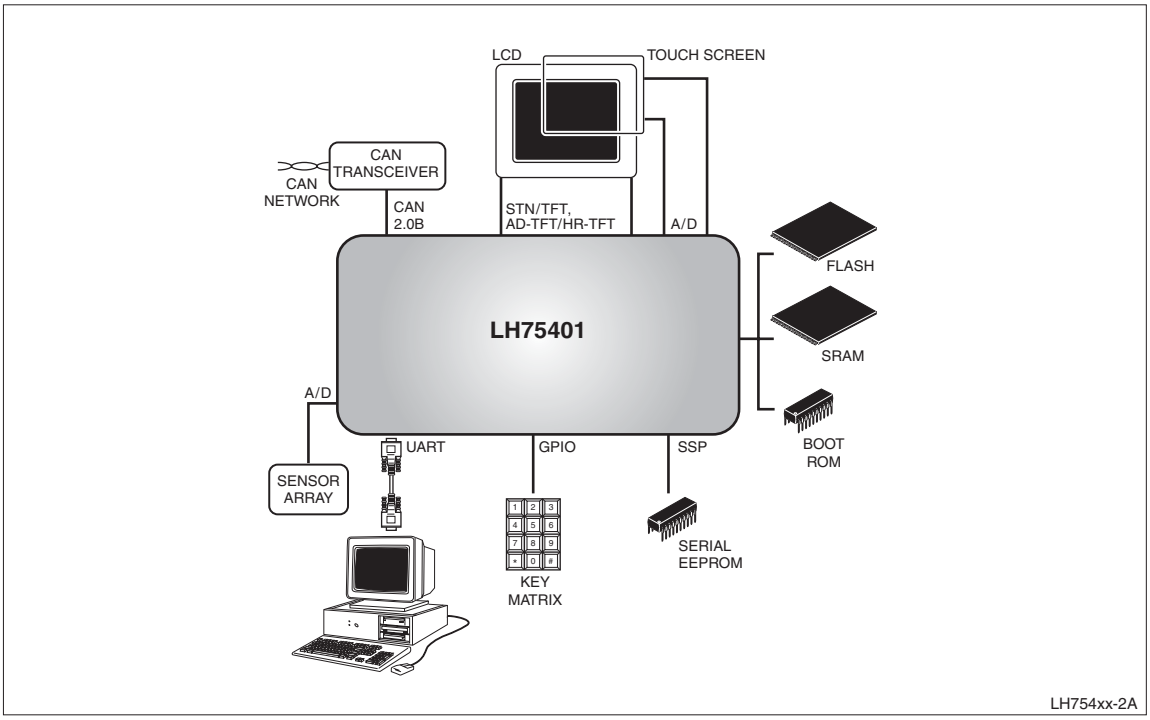


Figure 2-2. LH75401 System Application Example

2.4 LH75401 Pin Diagram

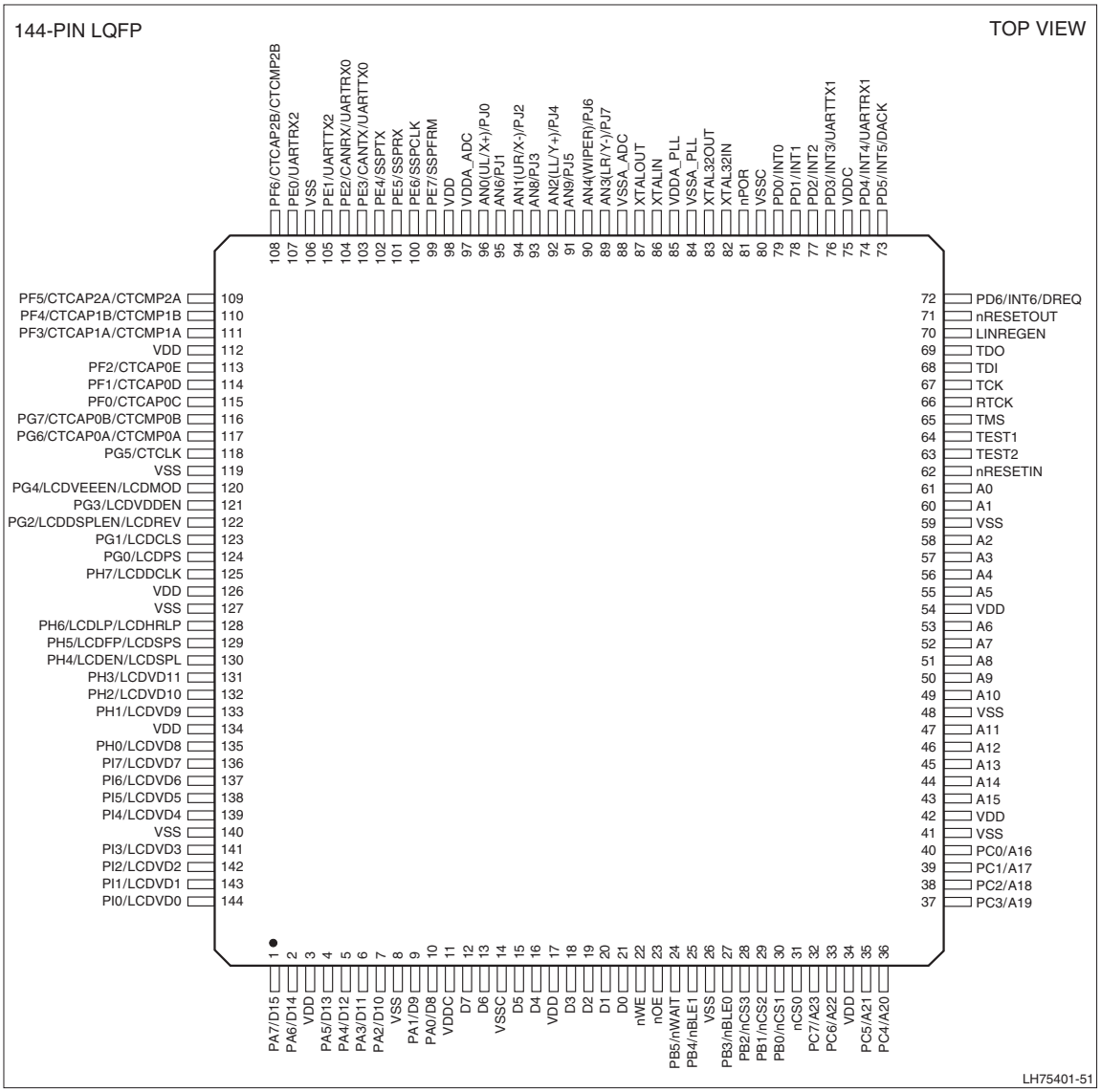


Figure 2-3. LH75401 Pin Diagram

2.5 LH75401 Numerical Pin Listing

Table 2-1. LH75401 Numerical Pin List

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
1	PA7	D15		I/O	8 mA	Bidirectional		1
2	PA6	D14		I/O	8 mA	Bidirectional		1
3	VDD			Power	None			
4	PA5	D13		I/O	8 mA	Bidirectional		1
5	PA4	D12		I/O	8 mA	Bidirectional		1
6	PA3	D11		I/O	8 mA	Bidirectional		1
7	PA2	D10		I/O	8 mA	Bidirectional		1
8	VSS			Ground	None			
9	PA1	D9		I/O	8 mA	Bidirectional		1
10	PA0	D8		I/O	8 mA	Bidirectional		1
11	VDDC			Power	None			
12	D7			I/O	8 mA	Bidirectional		
13	D6			I/O	8 mA	Bidirectional		
14	VSSC			Ground	None			
15	D5			I/O	8 mA	Bidirectional		
16	D4			I/O	8 mA	Bidirectional		
17	VDD			Power	None			
18	D3			I/O	8 mA	Bidirectional		
19	D2			I/O	8 mA	Bidirectional		
20	D1			I/O	8 mA	Bidirectional		
21	D0			I/O	8 mA	Bidirectional		
22	nWE				8 mA	Output		3
23	nOE				8 mA	Output		3
24	PB5	nWAIT			8 mA	Bidirectional	Pull-up	1, 3
25	PB4	nBLE1			8 mA	Bidirectional	Pull-up	1, 3
26	VSS			Ground	None			
27	PB3	nBLE0			8 mA	Bidirectional	Pull-up	1, 3
28	PB2	nCS3			8 mA	Bidirectional	Pull-up	1, 3
29	PB1	nCS2			8 mA	Bidirectional	Pull-up	1, 3
30	PB0	nCS1			8 mA	Bidirectional	Pull-up	1, 3
31	nCS0				8 mA	Output		3
32	PC7	A23			8 mA	Bidirectional	Pull-down	1
33	PC6	A22			8 mA	Bidirectional	Pull-down	1
34	VDD			Power	None			
35	PC5	A21			8 mA	Bidirectional	Pull-down	1
36	PC4	A20			8 mA	Bidirectional	Pull-down	1
37	PC3	A19			8 mA	Bidirectional	Pull-down	1
38	PC2	A18			8 mA	Bidirectional	Pull-down	1
39	PC1	A17			8 mA	Bidirectional	Pull-down	1

Table 2-1. LH75401 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
40	PC0	A16			8 mA	Bidirectional	Pull-down	1
41	VSS			Ground	None			
42	VDD			Power	None			
43	A15				8 mA	Output		
44	A14				8 mA	Output		
45	A13				8 mA	Output		
46	A12				8 mA	Output		
47	A11				8 mA	Output		
48	VSS			Ground	None			
49	A10				8 mA	Output		
50	A9				8 mA	Output		
51	A8				8 mA	Output		
52	A7				8 mA	Output		
53	A6				8 mA	Output		
54	VDD			Power	None			
55	A5				8 mA	Output		
56	A4				8 mA	Output		
57	A3				8 mA	Output		
58	A2				8 mA	Output		
59	VSS			Ground	None			
60	A1				8 mA	Output		
61	A0				8 mA	Output		
62	nRESETIN				None	Input	Pull-up	2, 3
63	TEST2				None	Input	Pull-up	2
64	TEST1				None	Input	Pull-up	2
65	TMS				None	Input	Pull-up	2
66	RTCK				8 mA	Output		
67	TCK				None	Input		
68	TDI				None	Input	Pull-up	2
69	TDO				4 mA	Output		
70	LINREGEN				None	Input		
71	nRESETOUT				8 mA	Output		3
72	PD6	INT6	DREQ		6 mA	Bidirectional	Pull-down	1
73	PD5	INT5	DACK		6 mA	Bidirectional		1, 2
74	PD4	INT4	UARTRX1		8 mA	Bidirectional	Pull-up	1
75	VDDC			Power	None			
76	PD3	INT3	UARTTX1		8 mA	Bidirectional	Pull-up	1
77	PD2	INT2			2 mA	Bidirectional	Pull-up	1
78	PD1	INT1			6 mA	Bidirectional		1, 2
79	PD0	INT0			2 mA	Bidirectional		1
80	VSSC			Ground	None			

Table 2-1. LH75401 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
81	nPOR				None	Input	Pull-up	2, 3
82	XTAL32IN				None	Input		
83	XTAL32OUT				None	Output		
84	VSSA_PLL			Ground	None			
85	VDDA_PLL			Power	None			
86	XTALIN				None	Input		
87	XTALOUT				None	Output		
88	VSSA_ADC			Ground	None			
89	AN3 (LR/Y-)	PJ7			None	Input		
90	AN4 (Wiper)	PJ6			None	Input		
91	AN9	PJ5			None	Input		
92	AN2 (LL/Y+)	PJ4			None	Input		
93	AN8	PJ3			None	Input		
94	AN1 (UR/X-)	PJ2			None	Input		
95	AN6	PJ1			None	Input		
96	AN0 (UL/X+)	PJ0			None	Input		
97	VDDA_ADC			Power	None			
98	VDD			Power	None			
99	PE7	SSPFRM			4 mA	Bidirectional	Pull-up	1
100	PE6	SSPCLK			4 mA	Bidirectional	Pull-down	1
101	PE5	SSPRX			4 mA	Bidirectional	Pull-up	1
102	PE4	SSPTX			4 mA	Bidirectional	Pull-down	1
103	PE3	CANTX	UARTTX0		8 mA	Bidirectional	Pull-up	1
104	PE2	CANRX	UARTRX0		2 mA	Bidirectional	Pull-up	1
105	PE1	UARTTX2			4 mA	Bidirectional	Pull-up	1
106	VSS			Ground	None			
107	PE0	UARTRX2			4 mA	Bidirectional	Pull-up	1
108	PF6	CTCAP2B	CTCMP2B		4 mA	Bidirectional		2
109	PF5	CTCAP2A	CTCMP2A		4 mA	Bidirectional		
110	PF4	CTCAP1B	CACMP1B		4 mA	Bidirectional		2
111	PF3	CTCAP1A	CTCMP1A		4 mA	Bidirectional		
112	VDD			Power	None			
113	PF2	CTCAP0E			4 mA	Bidirectional		2
114	PF1	CTCAP0D			4 mA	Bidirectional		
115	PF0	CTCAP0C			4 mA	Bidirectional		2
116	PG7	CTCAP0B	CTCMP0B		4 mA	Bidirectional		
117	PG6	CTCAP0A	CTCMP0A		4 mA	Bidirectional		2
118	PG5	CTCLK			4 mA	Bidirectional		
119	VSS			Ground	None			
120	PG4	LCDVEEEN	LCDMOD		8 mA	Bidirectional		
121	PG3	LCDVDDEN			8 mA	Bidirectional		

Table 2-1. LH75401 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
122	PG2	LCDDSPLEN	LCDREV		8 mA	Bidirectional		
123	PG1	LCDCLS			8 mA	Bidirectional		
124	PG0	LCDPS			8 mA	Bidirectional		
125	PH7	LCDDCLK			8 mA	Bidirectional		
126	VDD			Power	None			
127	VSS			Ground	None			
128	PH6	LCDLP	LCDHRLP		8 mA	Bidirectional		
129	PH5	LCDFP	LCDSPS		8 mA	Bidirectional		
130	PH4	LCDEN	LCDSPL		8 mA	Bidirectional		
131	PH3	LCDVD11			8 mA	Bidirectional		
132	PH2	LCDVD10			8 mA	Bidirectional		
133	PH1	LCDVD9			8 mA	Bidirectional		
134	VDD			Power	None			
135	PH0	LCDVD8			8 mA	Bidirectional		
136	PI7	LCDVD7			8 mA	Bidirectional		
137	PI6	LCDVD6			8 mA	Bidirectional		
138	PI5	LCDVD5			8 mA	Bidirectional		
139	PI4	LCDVD4			8 mA	Bidirectional		
140	VSS			Ground	None			
141	PI3	LCDVD3			8 mA	Bidirectional		
142	PI2	LCDVD2			8 mA	Bidirectional		
143	PI1	LCDVD1			8 mA	Bidirectional		
144	PI0	LCDVD0			8 mA	Bidirectional		

NOTES:

1. Signal is selectable as pull-up, pull-down, or no pull-up/pull-down via the I/O Configuration peripheral.
2. CMOS Schmitt trigger input.
3. Signals preceded with 'n' are active LOW.

2.6 LH75401 Signal Descriptions

Table 2-2. LH75401 Signal Descriptions

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
MEMORY INTERFACE (MI)				
1 2 4 5 6 7 9 10 12 13 15 16 18 19 20 21	D[15:0]	Input/Output	Data Input/Output Signals	1
22	nWE	Output	Static Memory Controller Write Enable	2
23	nOE	Output	Static Memory Controller Output Enable	2
24	nWAIT	Input	Static Memory Controller External Wait Control	1, 2
25	nBLE1	Output	Static Memory Controller Byte Lane Strobe	1, 2
27	nBLE0	Output	Static Memory Controller Byte Lane Strobe	1, 2
28	nCS3	Output	Static Memory Controller Chip Select	1, 2
29	nCS2	Output	Static Memory Controller Chip Select	1, 2
30	nCS1	Output	Static Memory Controller Chip Select	1, 2
31	nCS0	Output	Static Memory Controller Chip Select	2
32 33 35 36 37 38 39 40 43 44 45 46 47 49 50 51 52 53 55 56 57 58 60 61	A[23:0]	Output	Address Signals	1
DMA CONTROLLER (DMAC)				
72	DREQ	Input	DMA Request	1
73	DACK	Output	DMA Acknowledge	1

Table 2-2. LH75401 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
COLOR LCD CONTROLLER (CLCDC)				
120	LCDMOD	Output	HR-TFT Signal Used by the Row Driver (HR-TFT only)	1
120	LCDVEEEN	Output	Analog Supply Enable (AC Bias SIgnal)	1
121	LCDVDDEN	Output	Digital Supply Enable	1
122	LCDDSPLEN	Output	LCD Panel Power Enable	1
122	LCDREV	Output	HR-TFT Reverse Signal (HR-TFT only)	1
123	LCDCLS	Output	HR-TFT Clock to the Row Drivers (HR-TFT only)	1
124	LCDPS	Output	HT-TFT Power Save (HR-TFT only)	1
125	LCDDCLK	Output	LCD Panel Clock	1
128	LCDLP	Output	Line Synchronization Pulse (STN), Horizontal Synchronization Pulse (TFT)	1
128	LCDHRLP	Output	HR-TFT Latch Pulse (HR-TFT only)	1
129	LCDFP	Output	Frame Pulse (STN), Vertical Synchronization Pulse (TFT)	1
129	LCDSPS	Output	HR-TFT Signal that Resets the Row Driver Counter (HR-TFT only)	1
130	LCDEN	Output	LCD Data Enable	1
130	LCDSPL	Output	HR-TFT Start Pulse Left (HR-TFT only)	1
131 132 133 135 136 137 138 139 141 142 143 144	LCDVD[11:0]	Output	LCD Panel Data bus	1
SYNCHRONOUS SERIAL PORT (SSP)				
99	SSPFRM	Input	SSP Serial Frame	1
100	SSPCLK	Input	SSP Clock	1
101	SSPRX	Input	SSP RXD	1
102	SSPTX	Output	SSP TXD	1
UART0 (U0)				
103	UARTTX0	Output	UART0 Transmitted Serial Data Output	1
104	UARTRX0	Input	UART0 Received Serial Data Input	1
UART1 (U1)				
74	UARTRX1	Input	UART1 Received Serial Data Input	1
76	UARTTX1	Output	UART1 Transmitted Serial Data Output	1
UART2 (U2)				
105	UARTTX2	Output	UART2 Transmitted Serial Data Output	1
107	UARTRX2	Input	UART2 Received Serial Data Input	1
CONTROLLER AREA NETWORK (CAN)				
103	CANTX	Output	CAN Transmitted Serial Data Output	1
104	CANRX	Input	CAN Received Serial Data Input	1

Table 2-2. LH75401 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
ANALOG-TO-DIGITAL CONVERTER (ADC)				
89 90 91 92 93 94 95 96	AN3 (LR/Y-) AN4 (Wiper) AN9 AN2 (LL/Y+) AN8 AN1 (UR/X-) AN6 AN0 (UL/X+)	Input	ADC Inputs	1
TIMER 0				
117 116 115 114 113	CTCAP0[A:E]	Input	Timer 0 Capture Inputs	1
117 116	CTCMP0[A:B]	Output	Timer 0 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 1				
111 110	CTCAP1[A:B]	Input	Timer 1 Capture Inputs	1
111 110	CTCMP1[A:B]	Output	Timer 1 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 2				
109 108	CTCAP2[A:B]	Input	Timer 2 Capture Inputs	1
109 108	CTCMP2[A:B]	Input	Timer 2 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
GENERAL PURPOSE INPUT/OUTPUT (GPIO)				
1 2 4 5 6 7 9 10	PA7 PA6 PA5 PA4 PA3 PA2 PA1 PA0	Input/Output	General Purpose I/O Signals - Port A	1
24 25 27 28 29 30	PB5 PB4 PB3 PB2 PB1 PB0	Input/Output	General Purpose I/O Signals - Port B	1
32 33 35 36 37 38 39 40	PC7 PC6 PC5 PC4 PC3 PC2 PC1 PC0	Input/Output	General Purpose I/O Signals - Port C	1

Table 2-2. LH75401 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
72 73 74 76 77 78 79	PD6 PD5 PD4 PD3 PD2 PD1 PD0	Input/Output	General Purpose I/O Signals - Port D	1
89 90 91 92 93 94 95 96	PJ7 PJ6 PJ5 PJ4 PJ3 PJ2 PJ1 PJ0	Input	General Purpose I/O Signals - Port J	1
99 100 101 102 103 104 105 107	PE7 PE6 PE5 PE4 PE3 PE2 PE1 PE0	Input/Output	General Purpose I/O Signals - Port E	1
108 109 110 111 113 114 115	PF6 PF5 PF4 PF3 PF2 PF1 PF0	Input/Output	General Purpose I/O Signals - Port F	1
116 117 118 120 121 122 123 124	PG7 PG6 PG5 PG4 PG3 PG2 PG1 PG0	Input/Output	General Purpose I/O Signals - Port G	1
125 128 129 130 131 132 133 135	PH7 PH6 PH5 PH4 PH3 PH2 PH1 PH0	Input/Output	General Purpose I/O Signals - Port H	1
136 137 138 139 141 142 143 144	PI7 PI6 PI5 PI4 PI3 PI2 PI1 PI0	Input/Output	General Purpose I/O Signals - Port I	1
RESET, CLOCK, AND POWER CONTROLLER (RCPC)				
62	nRESETIN	Input	User Reset Input	2
71	nRESETOUT	Output	System Reset Output	2
72	INT6	Input	External Interrupt Input 6	1

Table 2-2. LH75401 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
73	INT5	Input	External Interrupt Input 5	1
74	INT4	Input	External Interrupt Input 4	1
76	INT3	Input	External Interrupt Input 3	1
77	INT2	Input	External Interrupt Input 2	1
78	INT1	Input	External Interrupt Input 1	1
79	INT0	Input	External Interrupt Input 0	1
81	nPOR	Input	Power-on Reset Input	2
82	XTAL32IN	Input	32.768 kHz Crystal Clock Input	
83	XTAL32OUT	Output	32.768 kHz Crystal Clock Output	
86	XTALIN	Input	Crystal Clock Input	
87	XTALOUT	Output	Crystal Clock Output	
TEST INTERFACE				
63	TEST2	Input	Test Mode Pin 2	
64	TEST1	Input	Test Mode Pin 1	
65	TMS	Input	JTAG Test Mode Select Input	
66	RTCK	Output	Returned JTAG Test Clock Output	
67	TCK	Input	JTAG Test Clock Input	
68	TDI	Input	JTAG Test Serial Data Input	
69	TDO	Output	JTAG Test Data Serial Output	
POWER AND GROUND (GND)				
3 17 34 42 54 98 112 126 134	VDD	Power	I/O Ring VDD	
8 26 41 48 59 106 119 127 140	VSS	Power	I/O Ring VSS	
11 75	VDDC	Power	Core VDD supply (Output if Linear Regulator Enabled, Otherwise Input)	
14 80	VSSC	Power	Core VSS	
70	LINREGEN	Input	Linear Regulator Enable	
84	VSSA_PLL	Power	PLL Analog VSS	
85	VDDA_PLL	Power	PLL Analog VDD Supply	
88	VSSA_ADC	Power	A-to-D converter Analog VSS	
97	VDDA_ADC	Power	A-to-D converter Analog VDD Supply	

NOTES:

1. These pin numbers have multiplexed functions.
2. Signals preceded with 'n' are active LOW.

Chapter 3
LH75411 SoC

3.1 LH75411 Features

- ARM7TDMI-S™ Core
- Up to 70 MHz System Clock (HCLK)
  - Internal PLL Driven or External Clock Driven
  - Crystal Oscillator/Internal PLL Can Operate with Input Frequency Range of 14 MHz to 20 MHz.
- 32KB On-chip SRAM
  - 16KB Tightly Coupled Memory (TCM) SRAM
  - 16KB Internal SRAM.
- Clock and Power Management
  - Low Power Modes: Standby, Sleep, Stop.
- Eight Channel, 10-bit A/D Converter
- Integrated Touch Screen Controller
- Four DMA Channels
- Color and Grayscale LCD Controller
  - 12-bit (4096) Direct Mode Color, up to VGA (640 × 480 Dots per Inch)
  - 8-bit (256) Direct or Palletized Color, up to SVGA (800 x 600 DPI)
  - 4-bit (16) Direct Mode Color/Grayscale, up to XGA (1024 x 768 DPI)
  - 12-bit Video Bus
  - Supports Supertwist Nematic (STN), Thin Film Transistor (TFT), High Reflective TFT (HR-TFT), and Advanced TFT (AD-TFT) Displays.
- Serial interfaces
  - Two 16C550-type UARTs
  - One 82510-type UART.
- Synchronous Serial Port (SSP)
  - Motorola SPI™
  - National Semiconductor Microwire™
  - Texas Instruments SSI.
- Real-Time Clock (RTC)
- Three Counter/Timers
  - Capture/Compare/PWM Compatibility
  - Watchdog Timer (WDT).
- Low-Voltage Detector
- JTAG Debug Interface and Boundary Scan

- Single 3.3 V Supply
- 5.0 V Tolerant Inputs
- 144-pin LQFP Package
- -40°C to +85°C Operating Temperature.

3.2 LH75411 Block Diagram

Figure 3-1 shows a block diagram of the LH75411. For information about the blocks shown in this figure, see the appropriate Chapters in this User's Guide.

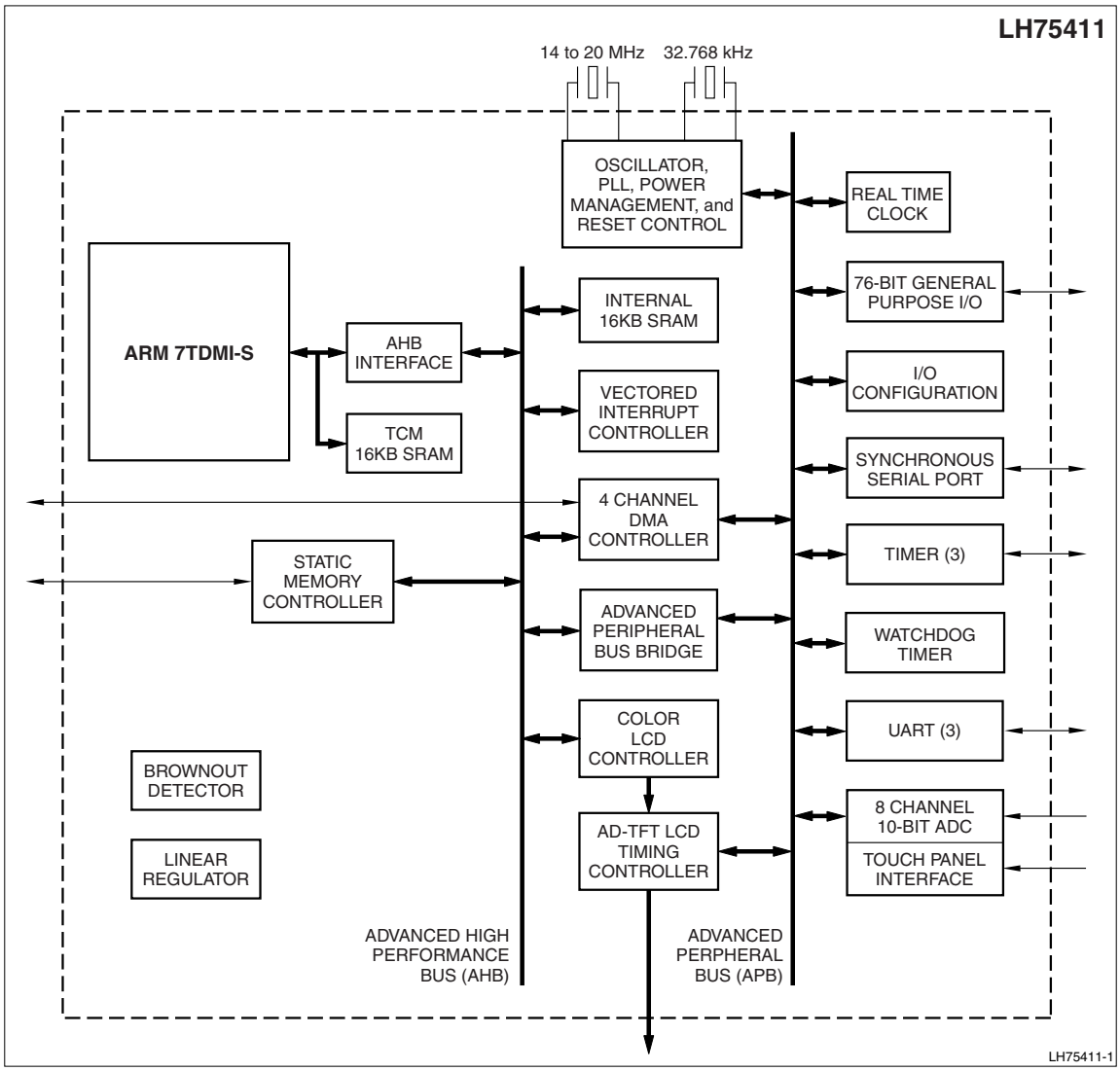


Figure 3-1. LH75411 Block Diagram

3.3 LH75411 Applications

The SHARP LH75411 is ideally suited for a variety of applications, including white-goods and industrial-control.

Typical white-goods applications include, but are not limited to:

- Air conditioners
- Washing machines
- Refrigerators
- Cooking equipment.

Typical industrial-control applications include, but are not limited to:

- Measuring instruments
- Machine-control systems
- Program-logic controllers.

Figure 3-2 shows a system application example for the LH75411.

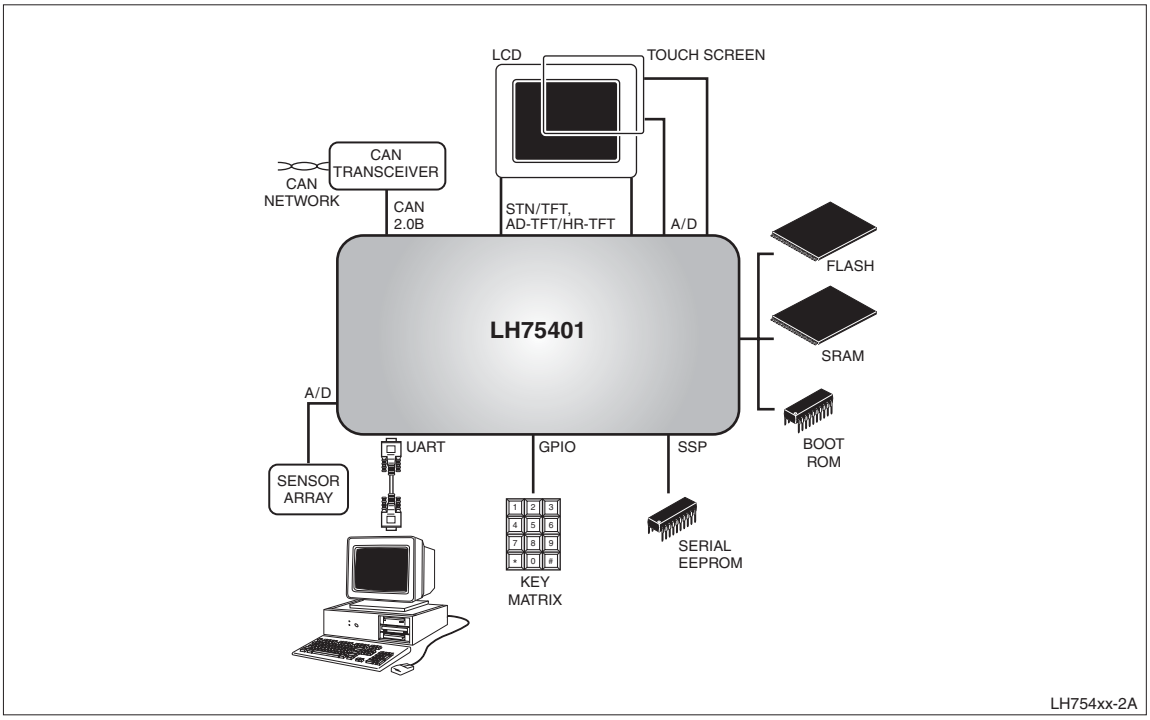


Figure 3-2. LH75411 System Application Example

3.4 LH75411 Pin Diagram

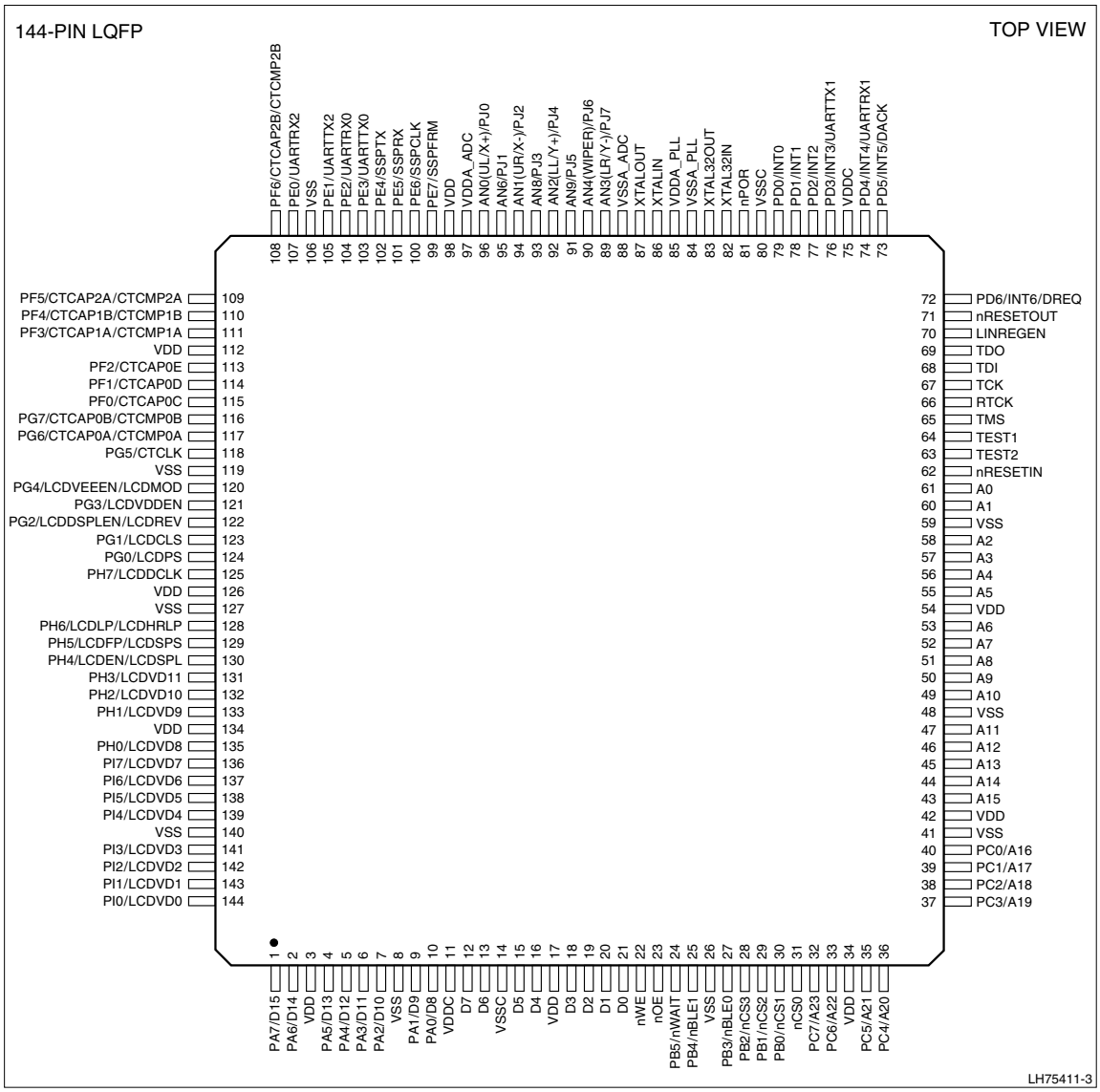


Figure 3-3. LH75411 Pin Diagram

3.5 LH75411 Numerical Pin Listing

Table 3-1. LH75411 Numerical Pin List

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
1	PA7	D15		I/O	8 mA	Bidirectional		1
2	PA6	D14		I/O	8 mA	Bidirectional		1
3	VDD			Power	None			
4	PA5	D13		I/O	8 mA	Bidirectional		1
5	PA4	D12		I/O	8 mA	Bidirectional		1
6	PA3	D11		I/O	8 mA	Bidirectional		1
7	PA2	D10		I/O	8 mA	Bidirectional		1
8	VSS			Ground	None			
9	PA1	D9		I/O	8 mA	Bidirectional		1
10	PA0	D8		I/O	8 mA	Bidirectional		1
11	VDDC			Power	None			
12	D7			I/O	8 mA	Bidirectional		
13	D6			I/O	8 mA	Bidirectional		
14	VSSC			Ground	None			
15	D5			I/O	8 mA	Bidirectional		
16	D4			I/O	8 mA	Bidirectional		
17	VDD			Power	None			
18	D3			I/O	8 mA	Bidirectional		
19	D2			I/O	8 mA	Bidirectional		
20	D1			I/O	8 mA	Bidirectional		
21	D0			I/O	8 mA	Bidirectional		
22	nWE				8 mA	Output		3
23	nOE				8 mA	Output		3
24	PB5	nWAIT			8 mA	Bidirectional	Pull-up	1, 3
25	PB4	nBLE1			8 mA	Bidirectional	Pull-up	1, 3
26	VSS			Ground	None			
27	PB3	nBLE0			8 mA	Bidirectional	Pull-up	1, 3
28	PB2	nCS3			8 mA	Bidirectional	Pull-up	1, 3
29	PB1	nCS2			8 mA	Bidirectional	Pull-up	1, 3
30	PB0	nCS1			8 mA	Bidirectional	Pull-up	1, 3
31	nCS0				8 mA	Output		3
32	PC7	A23			8 mA	Bidirectional	Pull-down	1
33	PC6	A22			8 mA	Bidirectional	Pull-down	1
34	VDD			Power	None			
35	PC5	A21			8 mA	Bidirectional	Pull-down	1
36	PC4	A20			8 mA	Bidirectional	Pull-down	1
37	PC3	A19			8 mA	Bidirectional	Pull-down	1
38	PC2	A18			8 mA	Bidirectional	Pull-down	1
39	PC1	A17			8 mA	Bidirectional	Pull-down	1

Table 3-1. LH75411 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
40	PC0	A16			8 mA	Bidirectional	Pull-down	1
41	VSS			Ground	None			
42	VDD			Power	None			
43	A15				8 mA	Output		
44	A14				8 mA	Output		
45	A13				8 mA	Output		
46	A12				8 mA	Output		
47	A11				8 mA	Output		
48	VSS			Ground	None			
49	A10				8 mA	Output		
50	A9				8 mA	Output		
51	A8				8 mA	Output		
52	A7				8 mA	Output		
53	A6				8 mA	Output		
54	VDD			Power	None			
55	A5				8 mA	Output		
56	A4				8 mA	Output		
57	A3				8 mA	Output		
58	A2				8 mA	Output		
59	VSS			Ground	None			
60	A1				8 mA	Output		
61	A0				8 mA	Output		
62	nRESETIN				None	Input	Pull-up	2, 3
63	TEST2				None	Input	Pull-up	2
64	TEST1				None	Input	Pull-up	2
65	TMS				None	Input	Pull-up	2
66	RTCK				8 mA	Output		
67	TCK				None	Input		
68	TDI				None	Input	Pull-up	2
69	TDO				4 mA	Output		
70	LINREGEN				None	Input		
71	nRESETOUT				8 mA	Output		3
72	PD6	INT6	DREQ		6 mA	Bidirectional	Pull-down	1
73	PD5	INT5	DACK		6 mA	Bidirectional		1, 2
74	PD4	INT4	UARTRX1		8 mA	Bidirectional	Pull-up	1
75	VDDC			Power	None			
76	PD3	INT3	UARTTX1		8 mA	Bidirectional	Pull-up	1
77	PD2	INT2			2 mA	Bidirectional	Pull-up	1
78	PD1	INT1			6 mA	Bidirectional		1, 2
79	PD0	INT0			2 mA	Bidirectional		1
80	VSSC			Ground	None			

Table 3-1. LH75411 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
81	nPOR				None	Input	Pull-up	2, 3
82	XTAL32IN				None	Input		
83	XTAL32OUT				None	Output		
84	VSSA_PLL			Ground	None			
85	VDDA_PLL			Power	None			
86	XTALIN				None	Input		
87	XTALOUT				None	Output		
88	VSSA_ADC			Ground	None			
89	AN3 (LR/Y-)	PJ7			None	Input		
90	AN4 (Wiper)	PJ6			None	Input		
91	AN9	PJ5			None	Input		
92	AN2 (LL/Y+)	PJ4			None	Input		
93	AN8	PJ3			None	Input		
94	AN1 (UR/X-)	PJ2			None	Input		
95	AN6	PJ1			None	Input		
96	AN0 (UL/X+)	PJ0			None	Input		
97	VDDA_ADC			Power	None			
98	VDD			Power	None			
99	PE7	SSPFRM			4 mA	Bidirectional	Pull-up	1
100	PE6	SSPCLK			4 mA	Bidirectional	Pull-down	1
101	PE5	SSPRX			4 mA	Bidirectional	Pull-up	1
102	PE4	SSPTX			4 mA	Bidirectional	Pull-down	1
103	PE3	UARTTX0			8 mA	Bidirectional	Pull-up	1
104	PE2	UARTRX0			2 mA	Bidirectional	Pull-up	1
105	PE1	UARTTX2			4 mA	Bidirectional	Pull-up	1
106	VSS			Ground	None			
107	PE0	UARTRX2			4 mA	Bidirectional	Pull-up	1
108	PF6	CTCAP2B	CTCMP2B		4 mA	Bidirectional		2
109	PF5	CTCAP2A	CTCMP2A		4 mA	Bidirectional		
110	PF4	CTCAP1B	CACMP1B		4 mA	Bidirectional		2
111	PF3	CTCAP1A	CTCMP1A		4 mA	Bidirectional		
112	VDD			Power	None			
113	PF2	CTCAP0E			4 mA	Bidirectional		2
114	PF1	CTCAP0D			4 mA	Bidirectional		
115	PF0	CTCAP0C			4 mA	Bidirectional		2
116	PG7	CTCAP0B	CTCMP0B		4 mA	Bidirectional		
117	PG6	CTCAP0A	CTCMP0A		4 mA	Bidirectional		2
118	PG5	CTCLK			4 mA	Bidirectional		
119	VSS			Ground	None			
120	PG4	LCDVEEEN	LCDMOD		8 mA	Bidirectional		
121	PG3	LCDVDDEN			8 mA	Bidirectional		

Table 3-1. LH75411 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
122	PG2	LCDDSPLEN	LCDREV		8 mA	Bidirectional		
123	PG1	LCDCLS			8 mA	Bidirectional		
124	PG0	LCDPS			8 mA	Bidirectional		
125	PH7	LCDDCLK			8 mA	Bidirectional		
126	VDD			Power	None			
127	VSS			Ground	None			
128	PH6	LCDLP	LCDHRLP		8 mA	Bidirectional		
129	PH5	LCDFP	LCDSPS		8 mA	Bidirectional		
130	PH4	LCDEN	LCDSPL		8 mA	Bidirectional		
131	PH3	LCDVD11			8 mA	Bidirectional		
132	PH2	LCDVD10			8 mA	Bidirectional		
133	PH1	LCDVD9			8 mA	Bidirectional		
134	VDD			Power	None			
135	PH0	LCDVD8			8 mA	Bidirectional		
136	PI7	LCDVD7			8 mA	Bidirectional		
137	PI6	LCDVD6			8 mA	Bidirectional		
138	PI5	LCDVD5			8 mA	Bidirectional		
139	PI4	LCDVD4			8 mA	Bidirectional		
140	VSS			Ground	None			
141	PI3	LCDVD3			8 mA	Bidirectional		
142	PI2	LCDVD2			8 mA	Bidirectional		
143	PI1	LCDVD1			8 mA	Bidirectional		
144	PI0	LCDVD0			8 mA	Bidirectional		

NOTES:

1. Signal is selectable as pull-up, pull-down, or no pull-up/pull-down via the I/O Configuration peripheral.
2. CMOS Schmitt trigger input.
3. Signals preceded with 'n' are active LOW.

3.6 LH75411 Signal Descriptions

Table 3-2. LH75411 Signal Descriptions

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
MEMORY INTERFACE (MI)				
1 2 4 5 6 7 9 10 12 13 15 16 18 19 20 21	D[15:0]	Input/Output	Data Input/Output Signals	1
22	nWE	Output	Static Memory Controller Write Enable	2
23	nOE	Output	Static Memory Controller Output Enable	2
24	nWAIT	Input	Static Memory Controller External Wait Control	1, 2
25	nBLE1	Output	Static Memory Controller Byte Lane Strobe	1, 2
27	nBLE0	Output	Static Memory Controller Byte Lane Strobe	1, 2
28	nCS3	Output	Static Memory Controller Chip Select	1, 2
29	nCS2	Output	Static Memory Controller Chip Select	1, 2
30	nCS1	Output	Static Memory Controller Chip Select	1, 2
31	nCS0	Output	Static Memory Controller Chip Select	2
32 33 35 36 37 38 39 40 43 44 45 46 47 49 50 51 52 53 55 56 57 58 60 61	A[23:0]	Output	Address Signals	1
DMA CONTROLLER (DMAC)				
72	DREQ	Input	DMA Request	1
73	DACK	Output	DMA Acknowledge	1

Table 3-2. LH75411 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
COLOR LCD CONTROLLER (CLCDC)				
120	LCDMOD	Output	Row Driver Signal (AD-TFT, HR-TFT only)	1
120	LCDVEEEN	Output	Analog Supply Enable (AC Bias SIgnal)	1
121	LCDVDDEN	Output	Digital Supply Enable	1
122	LCDDSPLEN	Output	LCD Panel Power Enable	1
122	LCDREV	Output	Reverse Signal (AD-TFT, HR-TFT only)	1
123	LCDCLS	Output	Row Drivers Clock (AD-TFT, HR-TFT only)	1
124	LCDPS	Output	Power Save (AD-TFT, HR-TFT only)	1
125	LCDDCLK	Output	LCD Panel Clock	1
128	LCDLP	Output	Line Synchronization Pulse (STN), Horizontal Synchronization Pulse (TFT)	1
128	LCDHRLP	Output	Latch Pulse (AD-TFT, HR-TFT only)	1
129	LCDFP	Output	Frame Pulse (STN), Vertical Synchronization Pulse (TFT)	1
129	LCDSPS	Output	Row Driver Counter Reset Signal (AD-TFT, HR-TFT only)	1
130	LCDEN	Output	LCD Data Enable	1
130	LCDSPL	Output	Start Pulse Left (AD-TFT, HR-TFT only)	1
131 132 133 135 136 137 138 139 141 142 143 144	LCDVD[11:0]	Output	LCD Panel Data bus	1
SYNCHRONOUS SERIAL PORT (SSP)				
99	SSPFRM	Input	SSP Serial Frame	1
100	SSPCLK	Input	SSP Clock	1
101	SSPRX	Input	SSP RXD	1
102	SSPTX	Output	SSP TXD	1
UART0 (U0)				
104	UARTRX0	Input	UART0 Received Serial Data Input	1
103	UARTTX0	Output	UART0 Transmitted Serial Data Output	1
UART1 (U1)				
74	UARTRX1	Input	UART1 Received Serial Data Input	1
76	UARTTX1	Output	UART1 Transmitted Serial Data Output	1
UART2 (U2)				
105	UARTTX2	Output	UART2 Transmitted Serial Data Output	1
107	UARTRX2	Input	UART2 Received Serial Data Input	1
ANALOG-TO-DIGITAL CONVERTER (ADC)				
89 90 91 92 93 94 95 96	AN3 (LR/Y-) AN4 (Wiper) AN9 AN2 (LL/Y+) AN8 AN1 (UR/X-) AN6 AN0 (UL/X+)	Input	ADC Inputs	1

Table 3-2. LH75411 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
TIMER 0				
117 116 115 114 113	CTCAP0[A:E]	Input	Timer 0 Capture Inputs	1
117 116	CTCMP0[A:B]	Output	Timer 0 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 1				
111 110	CTCAP1[A:B]	Input	Timer 1 Capture Inputs	1
111 110	CTCMP1[A:B]	Output	Timer 1 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 2				
109 108	CTCAP2[A:B]	Input	Timer 2 Capture Inputs	1
109 108	CTCMP2[A:B]	Input	Timer 2 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
GENERAL PURPOSE INPUT/OUTPUT (GPIO)				
1 2 4 5 6 7 9 10	PA7 PA6 PA5 PA4 PA3 PA2 PA1 PA0	Input/Output	General Purpose I/O Signals - Port A	1
24 25 27 28 29 30	PB5 PB4 PB3 PB2 PB1 PB0	Input/Output	General Purpose I/O Signals - Port B	1
32 33 35 36 37 38 39 40	PC7 PC6 PC5 PC4 PC3 PC2 PC1 PC0	Input/Output	General Purpose I/O Signals - Port C	1
72 73 74 76 77 78 79	PD6 PD5 PD4 PD3 PD2 PD1 PD0	Input/Output	General Purpose I/O Signals - Port D	1

Table 3-2. LH75411 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
89 90 91 92 93 94 95 96	PJ7 PJ6 PJ5 PJ4 PJ3 PJ2 PJ1 PJ0	Input	General Purpose I/O Signals - Port J	1
99 100 101 102 103 104 105 107	PE7 PE6 PE5 PE4 PE3 PE2 PE1 PE0	Input/Output	General Purpose I/O Signals - Port E	1
108 109 110 111 113 114 115	PF6 PF5 PF4 PF3 PF2 PF1 PF0	Input/Output	General Purpose I/O Signals - Port F	1
116 117 118 120 121 122 123 124	PG7 PG6 PG5 PG4 PG3 PG2 PG1 PG0	Input/Output	General Purpose I/O Signals - Port G	1
125 128 129 130 131 132 133 135	PH7 PH6 PH5 PH4 PH3 PH2 PH1 PH0	Input/Output	General Purpose I/O Signals - Port H	1
136 137 138 139 141 142 143 144	PI7 PI6 PI5 PI4 PI3 PI2 PI1 PI0	Input/Output	General Purpose I/O Signals - Port I	1
RESET, CLOCK, AND POWER CONTROLLER (RCPC)				
62	nRESETIN	Input	User Reset Input	2
71	nRESETOUT	Output	System Reset Output	2
72	INT6	Input	External Interrupt Input 6	1
73	INT5	Input	External Interrupt Input 5	1
74	INT4	Input	External Interrupt Input 4	1
76	INT3	Input	External Interrupt Input 3	1
77	INT2	Input	External Interrupt Input 2	1
78	INT1	Input	External Interrupt Input 1	1

Table 3-2. LH75411 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
79	INT0	Input	External Interrupt Input 0	1
81	nPOR	Input	Power-on Reset Input	2
82	XTAL32IN	Input	32.768 kHz Crystal Clock Input	
83	XTAL32OUT	Output	32.768 kHz Crystal Clock Output	
86	XTALIN	Input	Crystal Clock Input	
87	XTALOUT	Output	Crystal Clock Output	
TEST INTERFACE				
63	TEST2	Input	Test Mode Pin 2	
64	TEST1	Input	Test Mode Pin 1	
65	TMS	Input	JTAG Test Mode Select Input	
66	RTCK	Output	Returned JTAG Test Clock Output	
67	TCK	Input	JTAG Test Clock Input	
68	TDI	Input	JTAG Test Serial Data Input	
69	TDO	Output	JTAG Test Data Serial Output	
POWER AND GROUND (GND)				
3 17 34 42 54 98 112 126 134	VDD	Power	I/O Ring VDD	
8 26 41 48 59 106 119 127 140	VSS	Power	I/O Ring VSS	
11 75	VDDC	Power	Core VDD supply (Output if Linear Regulator Enabled, Otherwise Input)	
14 80	VSSC	Power	Core VSS	
70	LINREGEN	Input	Linear Regulator Enable	
84	VSSA_PLL	Power	PLL Analog VSS	
85	VDDA_PLL	Power	PLL Analog VDD Supply	
88	VSSA_ADC	Power	A-to-D converter Analog VSS	
97	VDDA_ADC	Power	A-to-D converter Analog VDD Supply	

NOTES:
1. These pin numbers have multiplexed functions.
2. Signals preceded with 'n' are active LOW.

Chapter 4
LH75400 SoC

4.1 LH75400 Features

- ARM7TDMI-S™ Core
- Up to 70 MHz System Clock (HCLK)
  - Internal PLL Driven or External Clock Driven
  - Crystal Oscillator/Internal PLL Can Operate with Input Frequency Range of 14 MHz to 20 MHz.
- 32KB On-chip SRAM
  - 16KB Tightly Coupled Memory (TCM) SRAM
  - 16KB Internal SRAM.
- Clock and Power Management
  - Low Power Modes: Standby, Sleep, Stop.
- Eight Channel, 10-bit A/D Converter
- Integrated Touch Screen Controller
- Four DMA Channels
- Grayscale LCD Controller
  - 4-bit (16 Level) Grayscale, up to XGA (1024 × 768)
  - 8-bit Video Bus
  - Supports STN Displays.
- CAN Controller that supports CAN version 2.0B
- Serial interfaces
  - Two 16C550-type UARTs
  - One 82510-type UART.
- Synchronous Serial Port (SSP)
  - Motorola SPI™
  - National Semiconductor Microwire™
  - Texas Instruments SSI.
- Real-Time Clock (RTC)
- Three Counter/Timers
  - Capture/Compare/PWM Compatibility
  - Watchdog Timer (WDT).
- Low-Voltage Detector
- JTAG Debug Interface and Boundary Scan
- Single 3.3 V Supply
- 5.0 V Tolerant Inputs
- 144-pin LQFP Package
- -40°C to +85°C Operating Temperature.

4.2 LH75400 Block Diagram

Figure 4-1 shows a block diagram of the LH75400. For information about the blocks shown in this figure, see the appropriate Chapters in this User's Guide.

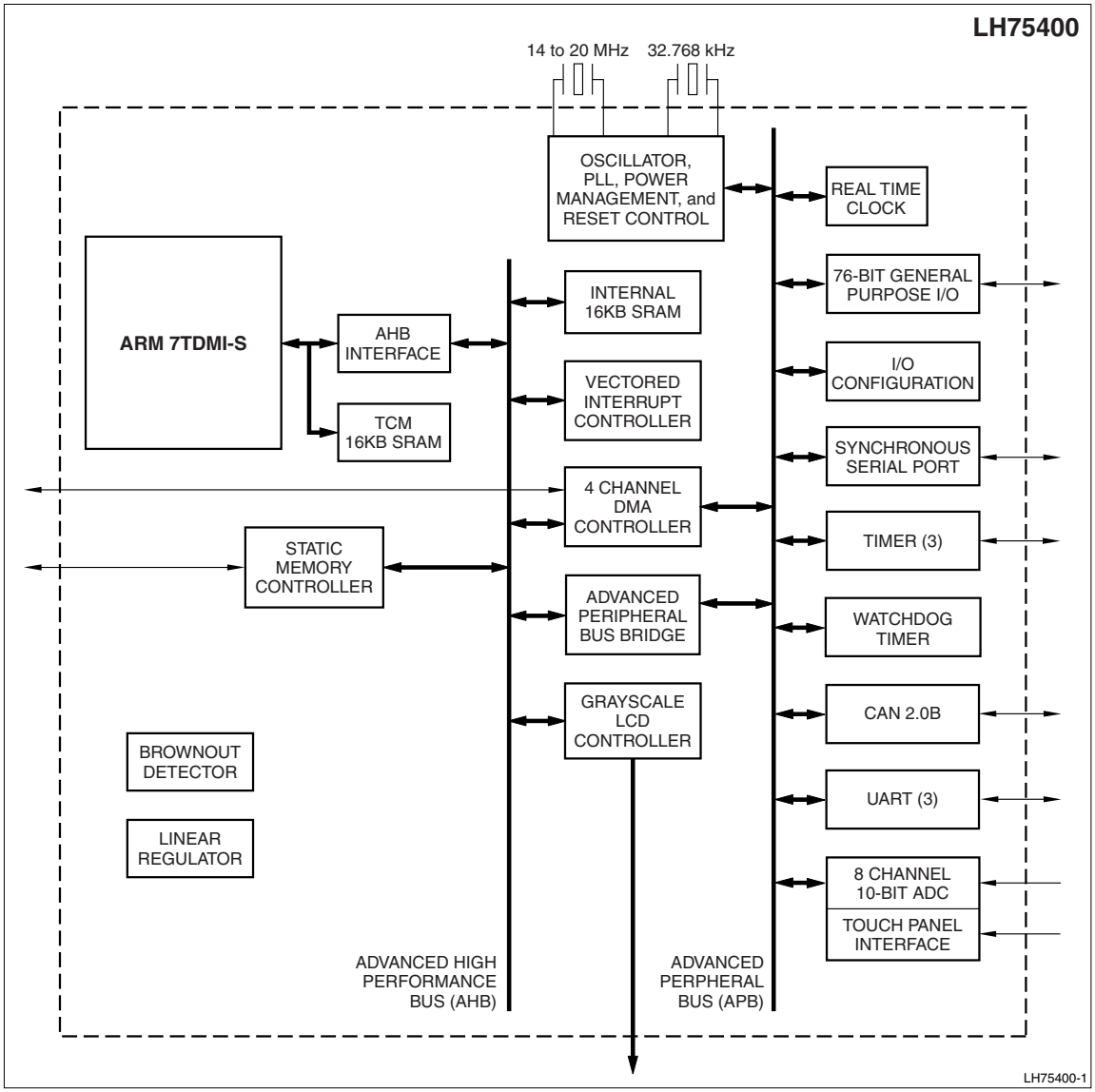


Figure 4-1. LH75400 Block Diagram

4.3 LH75400 Applications

The SHARP LH75400 is ideally suited for a variety of applications, including white-goods and industrial-control.

Typical white-goods applications include, but are not limited to:

- Air conditioners
- Washing machines
- Refrigerators
- Cooking equipment.

Typical industrial-control applications include, but are not limited to:

- Measuring instruments
- Machine-control systems
- Program-logic controllers.

Figure 4-2 shows a system application example for the LH75400.

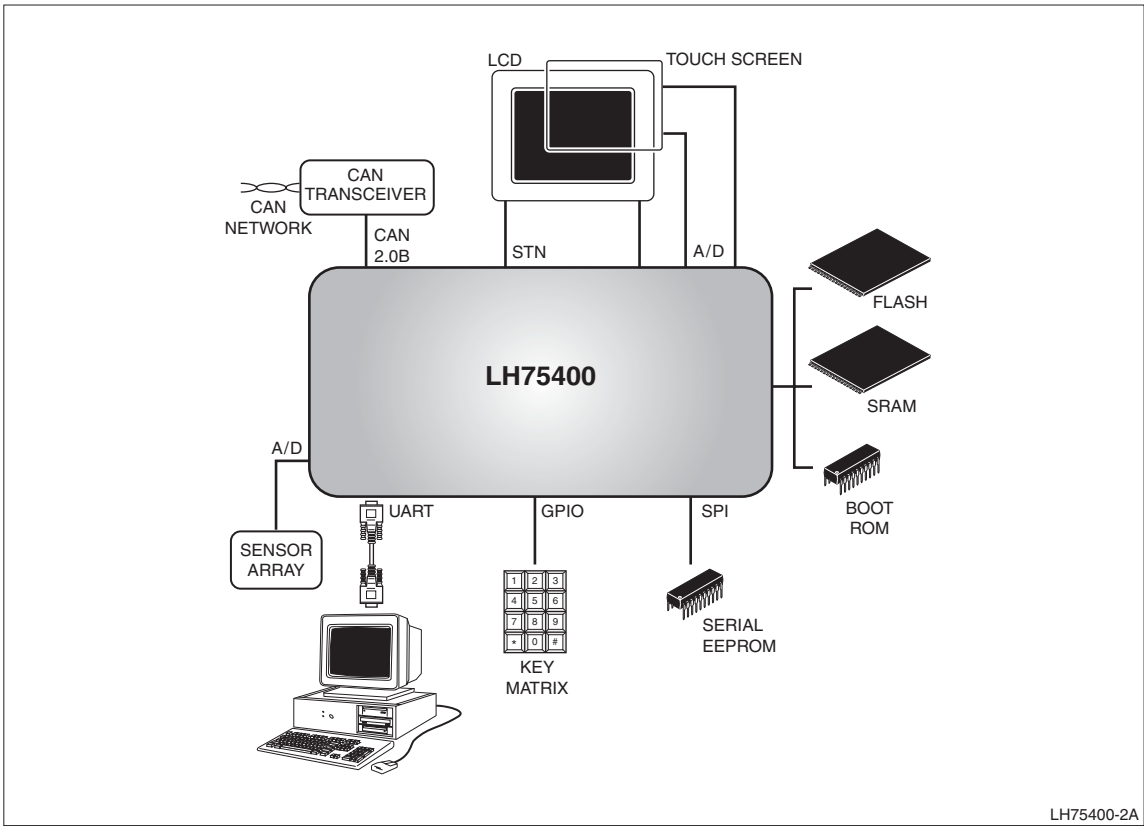


Figure 4-2. LH75400 System Application Example

4.4 LH75400 Pin Diagram

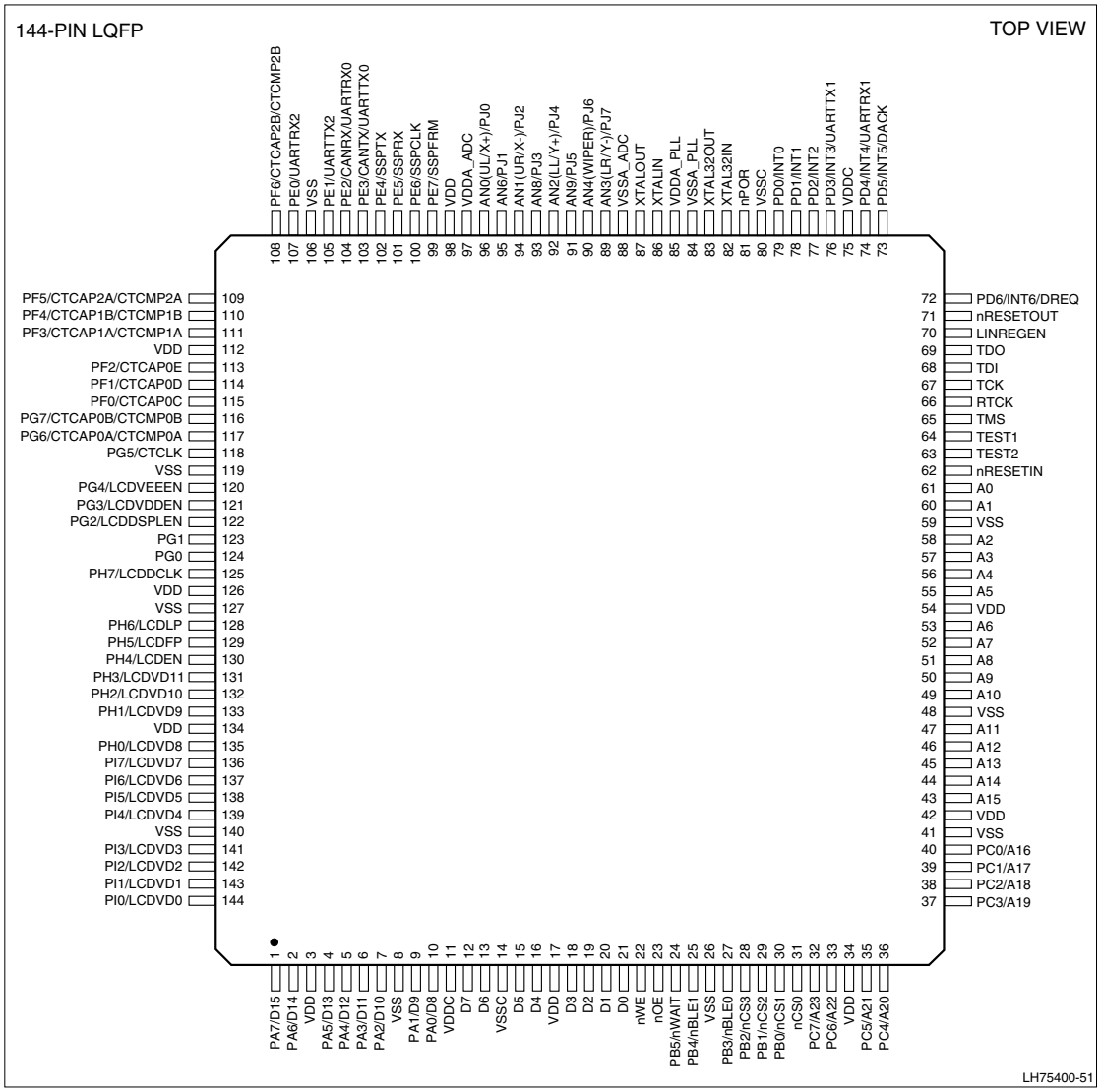


Figure 4-3. LH75400 Pin Diagram

4.5 LH75400 Numerical Pin Listing

Table 4-1. LH75400 Numerical Pin List

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
1	PA7	D15		I/O	8 mA	Bidirectional		1
2	PA6	D14		I/O	8 mA	Bidirectional		1
3	VDD			Power	None			
4	PA5	D13		I/O	8 mA	Bidirectional		1
5	PA4	D12		I/O	8 mA	Bidirectional		1
6	PA3	D11		I/O	8 mA	Bidirectional		1
7	PA2	D10		I/O	8 mA	Bidirectional		1
8	VSS			Ground	None			
9	PA1	D9		I/O	8 mA	Bidirectional		1
10	PA0	D8		I/O	8 mA	Bidirectional		1
11	VDDC			Power	None			
12	D7			I/O	8 mA	Bidirectional		
13	D6			I/O	8 mA	Bidirectional		
14	VSSC			Ground	None			
15	D5			I/O	8 mA	Bidirectional		
16	D4			I/O	8 mA	Bidirectional		
17	VDD			Power	None			
18	D3			I/O	8 mA	Bidirectional		
19	D2			I/O	8 mA	Bidirectional		
20	D1			I/O	8 mA	Bidirectional		
21	D0			I/O	8 mA	Bidirectional		
22	nWE				8 mA	Output		3
23	nOE				8 mA	Output		3
24	PB5	nWAIT			8 mA	Bidirectional	Pull-up	1, 3
25	PB4	nBLE1			8 mA	Bidirectional	Pull-up	1, 3
26	VSS			Ground	None			
27	PB3	nBLE0			8 mA	Bidirectional	Pull-up	1, 3
28	PB2	nCS3			8 mA	Bidirectional	Pull-up	1, 3
29	PB1	nCS2			8 mA	Bidirectional	Pull-up	1, 3
30	PB0	nCS1			8 mA	Bidirectional	Pull-up	1, 3
31	nCS0				8 mA	Output		3
32	PC7	A23			8 mA	Bidirectional	Pull-down	1
33	PC6	A22			8 mA	Bidirectional	Pull-down	1
34	VDD			Power	None			
35	PC5	A21			8 mA	Bidirectional	Pull-down	1
36	PC4	A20			8 mA	Bidirectional	Pull-down	1
37	PC3	A19			8 mA	Bidirectional	Pull-down	1
38	PC2	A18			8 mA	Bidirectional	Pull-down	1
39	PC1	A17			8 mA	Bidirectional	Pull-down	1

Table 4-1. LH75400 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
40	PC0	A16			8 mA	Bidirectional	Pull-down	1
41	VSS			Ground	None			
42	VDD			Power	None			
43	A15				8 mA	Output		
44	A14				8 mA	Output		
45	A13				8 mA	Output		
46	A12				8 mA	Output		
47	A11				8 mA	Output		
48	VSS			Ground	None			
49	A10				8 mA	Output		
50	A9				8 mA	Output		
51	A8				8 mA	Output		
52	A7				8 mA	Output		
53	A6				8 mA	Output		
54	VDD			Power	None			
55	A5				8 mA	Output		
56	A4				8 mA	Output		
57	A3				8 mA	Output		
58	A2				8 mA	Output		
59	VSS			Ground	None			
60	A1				8 mA	Output		
61	A0				8 mA	Output		
62	nRESETIN				None	Input	Pull-up	2, 3
63	TEST2				None	Input	Pull-up	2
64	TEST1				None	Input	Pull-up	2
65	TMS				None	Input	Pull-up	2
66	RTCK				8 mA	Output		
67	TCK				None	Input		
68	TDI				None	Input	Pull-up	2
69	TDO				4 mA	Output		
70	LINREGEN				None	Input		
71	nRESETOUT				8 mA	Output		3
72	PD6	INT6	DREQ		6 mA	Bidirectional	Pull-down	1
73	PD5	INT5	DACK		6 mA	Bidirectional		1, 2
74	PD4	INT4	UARTRX1		8 mA	Bidirectional	Pull-up	1
75	VDDC			Power	None			
76	PD3	INT3	UARTTX1		8 mA	Bidirectional	Pull-up	1
77	PD2	INT2			2 mA	Bidirectional	Pull-up	1
78	PD1	INT1			6 mA	Bidirectional		1, 2
79	PD0	INT0			2 mA	Bidirectional		1
80	VSSC			Ground	None			

Table 4-1. LH75400 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
81	nPOR				None	Input	Pull-up	2, 3
82	XTAL32IN				None	Input		
83	XTAL32OUT				None	Output		
84	VSSA_PLL			Ground	None			
85	VDDA_PLL			Power	None			
86	XTALIN				None	Input		
87	XTALOUT				None	Output		
88	VSSA_ADC			Ground	None			
89	AN3 (LR/Y-)	PJ7			None	Input		
90	AN4 (Wiper)	PJ6			None	Input		
91	AN9	PJ5			None	Input		
92	AN2 (LL/Y+)	PJ4			None	Input		
93	AN8	PJ3			None	Input		
94	AN1 (UR/X-)	PJ2			None	Input		
95	AN6	PJ1			None	Input		
96	AN0 (UL/X+)	PJ0			None	Input		
97	VDDA_ADC			Power	None			
98	VDD			Power	None			
99	PE7	SSPFRM			4 mA	Bidirectional	Pull-up	1
100	PE6	SSPCLK			4 mA	Bidirectional	Pull-down	1
101	PE5	SSPRX			4 mA	Bidirectional	Pull-up	1
102	PE4	SSPTX			4 mA	Bidirectional	Pull-down	1
103	PE3	CANTX	UARTTX0		8 mA	Bidirectional	Pull-up	1
104	PE2	CANRX	UARTRX0		2 mA	Bidirectional	Pull-up	1
105	PE1	UARTTX2			4 mA	Bidirectional	Pull-up	1
106	VSS			Ground	None			
107	PE0	UARTRX2			4 mA	Bidirectional	Pull-up	1
108	PF6	CTCAP2B	CTCMP2B		4 mA	Bidirectional		2
109	PF5	CTCAP2A	CTCMP2A		4 mA	Bidirectional		
110	PF4	CTCAP1B	CACMP1B		4 mA	Bidirectional		2
111	PF3	CTCAP1A	CTCMP1A		4 mA	Bidirectional		
112	VDD			Power	None			
113	PF2	CTCAP0E			4 mA	Bidirectional		2
114	PF1	CTCAP0D			4 mA	Bidirectional		
115	PF0	CTCAP0C			4 mA	Bidirectional		2
116	PG7	CTCAP0B	CTCMP0B		4 mA	Bidirectional		
117	PG6	CTCAP0A	CTCMP0A		4 mA	Bidirectional		2
118	PG5	CTCLK			4 mA	Bidirectional		
119	VSS			Ground	None			
120	PG4	LCDVEEEN			8 mA	Bidirectional		
121	PG3	LCDVDDEN			8 mA	Bidirectional		

Table 4-1. LH75400 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
122	PG2	LCDDSPLEN			8 mA	Bidirectional		
123	PG1				8 mA	Bidirectional		
124	PG0				8 mA	Bidirectional		
125	PH7	LCDDCLK			8 mA	Bidirectional		
126	VDD			Power	None			
127	VSS			Ground	None			
128	PH6	LCDLP			8 mA	Bidirectional		
129	PH5	LCDFP			8 mA	Bidirectional		
130	PH4	LCDEN			8 mA	Bidirectional		
131	PH3	LCDVD11			8 mA	Bidirectional		
132	PH2	LCDVD10			8 mA	Bidirectional		
133	PH1	LCDVD9			8 mA	Bidirectional		
134	VDD			Power	None			
135	PH0	LCDVD8			8 mA	Bidirectional		
136	PI7	LCDVD7			8 mA	Bidirectional		
137	PI6	LCDVD6			8 mA	Bidirectional		
138	PI5	LCDVD5			8 mA	Bidirectional		
139	PI4	LCDVD4			8 mA	Bidirectional		
140	VSS			Ground	None			
141	PI3	LCDVD3			8 mA	Bidirectional		
142	PI2	LCDVD2			8 mA	Bidirectional		
143	PI1	LCDVD1			8 mA	Bidirectional		
144	PI0	LCDVD0			8 mA	Bidirectional		

NOTES:

1. Signal is selectable as pull-up, pull-down, or no pull-up/pull-down via the I/O Configuration peripheral.
2. CMOS Schmitt trigger input.
3. Signals preceded with 'n' are active LOW.

4.6 LH75400 Signal Descriptions

Table 4-2. LH75400 Signal Descriptions

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
MEMORY INTERFACE (MI)				
1 2 4 5 6 7 9 10 12 13 15 16 18 19 20 21	D[15:0]	Input/Output	Data Input/Output Signals	1
22	nWE	Output	Static Memory Controller Write Enable	2
23	nOE	Output	Static Memory Controller Output Enable	2
24	nWAIT	Input	Static Memory Controller External Wait Control	1, 2
25	nBLE1	Output	Static Memory Controller Byte Lane Strobe	1, 2
27	nBLE0	Output	Static Memory Controller Byte Lane Strobe	1, 2
28	nCS3	Output	Static Memory Controller Chip Select	1, 2
29	nCS2	Output	Static Memory Controller Chip Select	1, 2
30	nCS1	Output	Static Memory Controller Chip Select	1, 2
31	nCS0	Output	Static Memory Controller Chip Select	2
32 33 35 36 37 38 39 40 43 44 45 46 47 49 50 51 52 53 55 56 57 58 60 61	A[23:0]	Output	Address Signals	1
DMA CONTROLLER (DMAC)				
72	DREQ	Input	DMA Request	1
73	DACK	Output	DMA Acknowledge	1

Table 4-2. LH75400 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
LCD CONTROLLER (LCDC)				
120	LCDVEEEN	Output	Analog Supply Enable (AC Bias SIgnal)	1
121	LCDVDDEN	Output	Digital Supply Enable	1
122	LCDDSPLEN	Output	LCD Panel Power Enable	1
125	LCDDCLK	Output	LCD Panel Clock	1
128	LCDLP	Output	Line Synchronization Pulse (STN), Horizontal Synchronization Pulse (TFT)	1
129	LCDFP	Output	Frame Pulse (STN), Vertical Synchronization Pulse (TFT)	1
130	LCDEN	Output	LCD Data Enable	1
131 132 133 135 136 137 138 139 141 142 143 144	LCDVD[11:0]	Output	LCD Panel Data bus	1
SYNCHRONOUS SERIAL PORT (SSP)				
99	SSPFRM	Input	SSP Serial Frame	1
100	SSPCLK	Input	SSP Clock	1
101	SSPRX	Input	SSP RXD	1
102	SSPTX	Output	SSP TXD	1
UART0 (U0)				
103	UARTTX0	Output	UART0 Transmitted Serial Data Output	1
104	UARTRX0	Input	UART0 Received Serial Data Input	1
UART1 (U1)				
74	UARTRX1	Input	UART1 Received Serial Data Input	1
76	UARTTX1	Output	UART1 Transmitted Serial Data Output	1
UART2 (U2)				
105	UARTTX2	Output	UART2 Transmitted Serial Data Output	1
107	UARTRX2	Input	UART2 Received Serial Data Input	1
ANALOG-TO-DIGITAL CONVERTER (ADC)				
89 90 91 92 93 94 95 96	AN3 (LR/Y-) AN4 (Wiper) AN9 AN2 (LL/Y+) AN8 AN1 (UR/X-) AN6 AN0 (UL/X+)	Input	ADC Inputs	1
CONTROLLER AREA NETWORK (CAN)				
103	CANTX	Output	CAN Transmitted Serial Data Output	1
104	CANRX	Input	CAN Received Serial Data Input	1

Table 4-2. LH75400 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
TIMER 0				
117 116 115 114 113	CTCAP0[A:E]	Input	Timer 0 Capture Inputs	1
117 116	CTCMP0[A:B]	Output	Timer 0 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 1				
111 110	CTCAP1[A:B]	Input	Timer 1 Capture Inputs	1
111 110	CTCMP1[A:B]	Output	Timer 1 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 2				
109 108	CTCAP2[A:B]	Input	Timer 2 Capture Inputs	1
109 108	CTCMP2[A:B]	Input	Timer 2 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
GENERAL PURPOSE INPUT/OUTPUT (GPIO)				
1 2 4 5 6 7 9 10	PA7 PA6 PA5 PA4 PA3 PA2 PA1 PA0	Input/Output	General Purpose I/O Signals - Port A	1
24 25 27 28 29 30	PB5 PB4 PB3 PB2 PB1 PB0	Input/Output	General Purpose I/O Signals - Port B	1
32 33 35 36 37 38 39 40	PC7 PC6 PC5 PC4 PC3 PC2 PC1 PC0	Input/Output	General Purpose I/O Signals - Port C	1
72 73 74 76 77 78 79	PD6 PD5 PD4 PD3 PD2 PD1 PD0	Input/Output	General Purpose I/O Signals - Port D	1

Table 4-2. LH75400 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
89 90 91 92 93 94 95 96	PJ7 PJ6 PJ5 PJ4 PJ3 PJ2 PJ1 PJ0	Input	General Purpose I/O Signals - Port J	1
99 100 101 102 103 104 105 107	PE7 PE6 PE5 PE4 PE3 PE2 PE1 PE0	Input/Output	General Purpose I/O Signals - Port E	1
108 109 110 111 113 114 115	PF6 PF5 PF4 PF3 PF2 PF1 PF0	Input/Output	General Purpose I/O Signals - Port F	1
116 117 118 120 121 122 123 124	PG7 PG6 PG5 PG4 PG3 PG2 PG1 PG0	Input/Output	General Purpose I/O Signals - Port G	1
125 128 129 130 131 132 133 135	PH7 PH6 PH5 PH4 PH3 PH2 PH1 PH0	Input/Output	General Purpose I/O Signals - Port H	1
136 137 138 139 141 142 143 144	PI7 PI6 PI5 PI4 PI3 PI2 PI1 PI0	Input/Output	General Purpose I/O Signals - Port I	1
RESET, CLOCK, AND POWER CONTROLLER (RCPC)				
62	nRESETIN	Input	User Reset Input	2
71	nRESETOUT	Output	System Reset Output	2
72	INT6	Input	External Interrupt Input 6	1
73	INT5	Input	External Interrupt Input 5	1
74	INT4	Input	External Interrupt Input 4	1
76	INT3	Input	External Interrupt Input 3	1
77	INT2	Input	External Interrupt Input 2	1
78	INT1	Input	External Interrupt Input 1	1

Table 4-2. LH75400 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
79	INT0	Input	External Interrupt Input 0	1
81	nPOR	Input	Power-on Reset Input	2
82	XTAL32IN	Input	32.768 kHz Crystal Clock Input	
83	XTAL32OUT	Output	32.768 kHz Crystal Clock Output	
86	XTALIN	Input	Crystal Clock Input	
87	XTALOUT	Output	Crystal Clock Output	
TEST INTERFACE				
63	TEST2	Input	Test Mode Pin 2	
64	TEST1	Input	Test Mode Pin 1	
65	TMS	Input	JTAG Test Mode Select Input	
66	RTCK	Output	Returned JTAG Test Clock Output	
67	TCK	Input	JTAG Test Clock Input	
68	TDI	Input	JTAG Test Serial Data Input	
69	TDO	Output	JTAG Test Data Serial Output	
POWER AND GROUND (GND)				
3 17 34 42 54 98 112 126 134	VDD	Power	I/O Ring VDD	
8 26 41 48 59 106 119 127 140	VSS	Power	I/O Ring VSS	
11 75	VDDC	Power	Core VDD supply (Output if Linear Regulator Enabled, Otherwise Input)	
14 80	VSSC	Power	Core VSS	
70	LINREGEN	Input	Linear Regulator Enable	
84	VSSA_PLL	Power	PLL Analog VSS	
85	VDDA_PLL	Power	PLL Analog VDD Supply	
88	VSSA_ADC	Power	A-to-D converter Analog VSS	
97	VDDA_ADC	Power	A-to-D converter Analog VDD Supply	

NOTES:
1. These pin numbers have multiplexed functions.
2. Signals preceded with 'n' are active LOW.

Chapter 5
LH75410 SoC

5.1 LH75410 Features

- ARM7TDMI-S™ Core
- Up to 70 MHz System Clock (HCLK)
  - Internal PLL Driven or External Clock Driven
  - Crystal Oscillator/Internal PLL Can Operate with Input Frequency Range of 14 MHz to 20 MHz.
- 32KB On-chip SRAM
  - 16KB Tightly Coupled Memory (TCM) SRAM
  - 16KB Internal SRAM.
- Clock and Power Management
  - Low Power Modes: Standby, Sleep, Stop.
- Eight Channel, 10-bit A/D Converter
- Integrated Touch Screen Controller
- Four DMA Channels
- Grayscale LCD Controller
  - 4-bit (16 Level) Grayscale, up to XGA (1024 × 768)
  - 8-bit Video Bus
  - Supports STN Displays.
- Serial interfaces
  - Two 16C550-type UARTs
  - One 82510-type UART.
- Synchronous Serial Port (SSP)
  - Motorola SPI™
  - National Semiconductor Microwire™
  - Texas Instruments SSI.
- Real-Time Clock (RTC)
- Three Counter/Timers
  - Capture/Compare/PWM Compatibility
  - Watchdog Timer (WDT).
- Low-Voltage Detector
- JTAG Debug Interface and Boundary Scan
- Single 3.3 V Supply
- 5.0 V Tolerant Inputs
- 144-pin LQFP Package
- -40°C to +85°C Operating Temperature.

5.2 LH75410 Block Diagram

Figure 5-1 shows a block diagram of the LH75410. For information about the blocks shown in this figure, see the appropriate Chapters in this User's Guide.

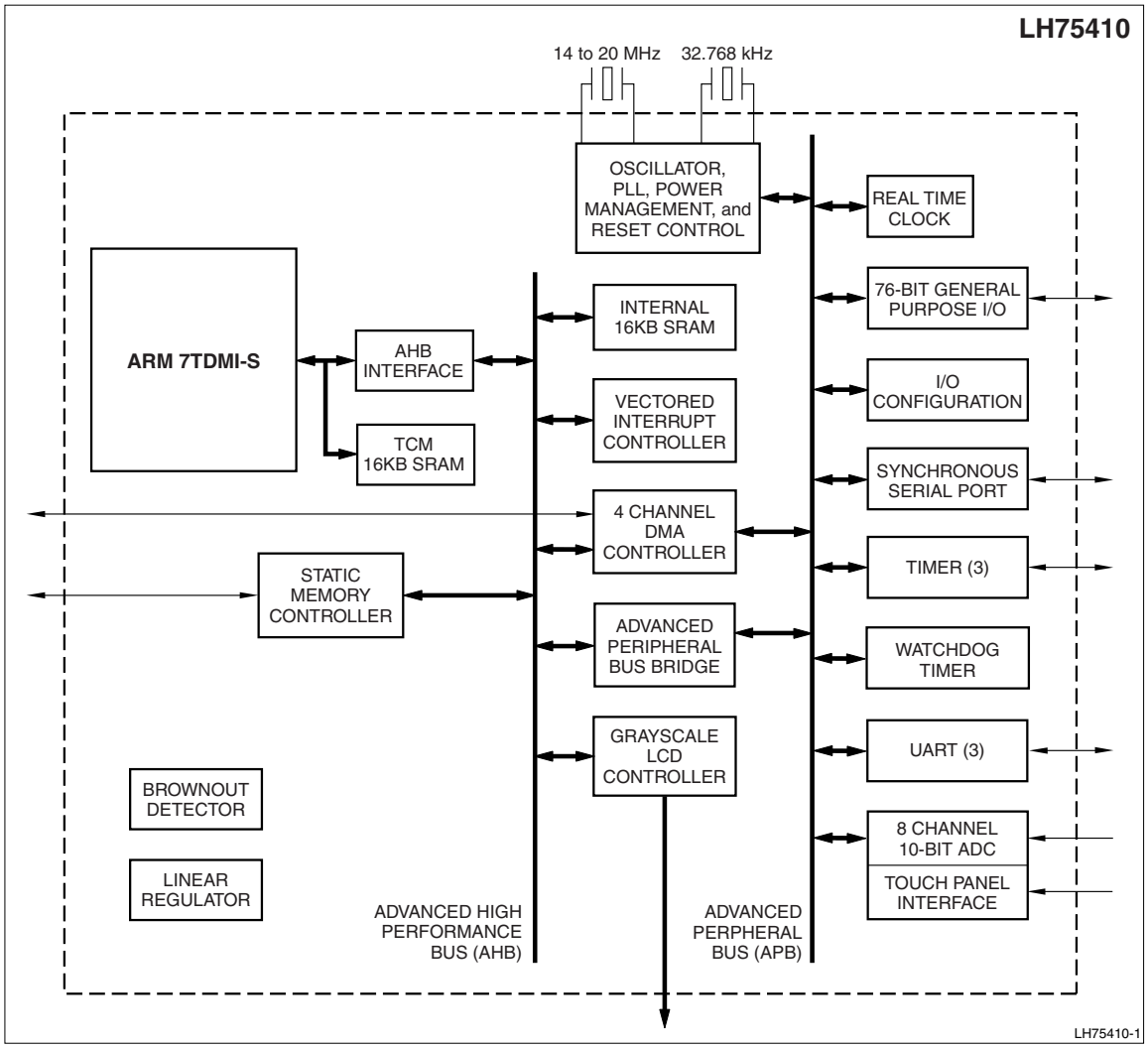


Figure 5-1. LH75410 Block Diagram

5.3 LH75410 Applications

The SHARP LH75410 is ideally suited for a variety of applications, including white-goods and industrial-control.

Typical white-goods applications include, but are not limited to:

- Air conditioners
- Washing machines
- Refrigerators
- Cooking equipment.

Typical industrial-control applications include, but are not limited to:

- Measuring instruments
- Machine-control systems
- Program-logic controllers.

Figure 5-2 shows a system application example for the LH75410.

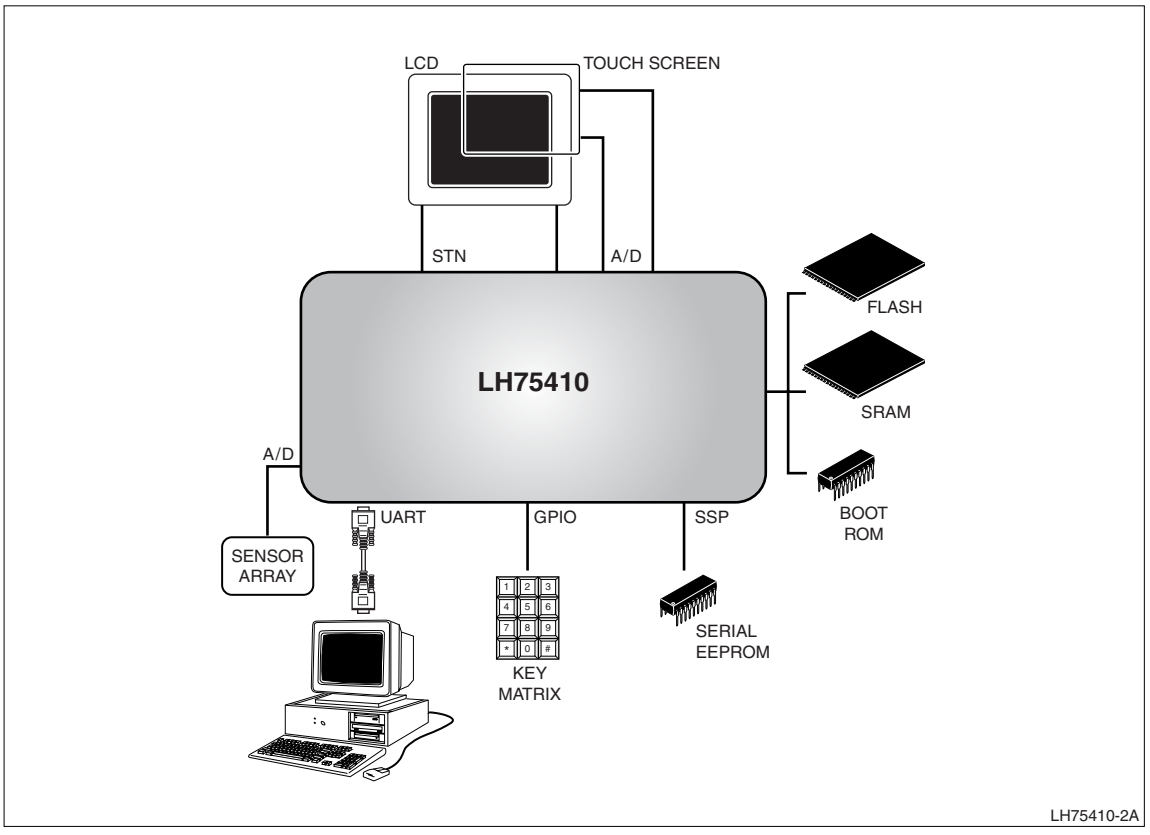


Figure 5-2. LH75410 System Application Example

5.4 LH75410 Pin Diagram

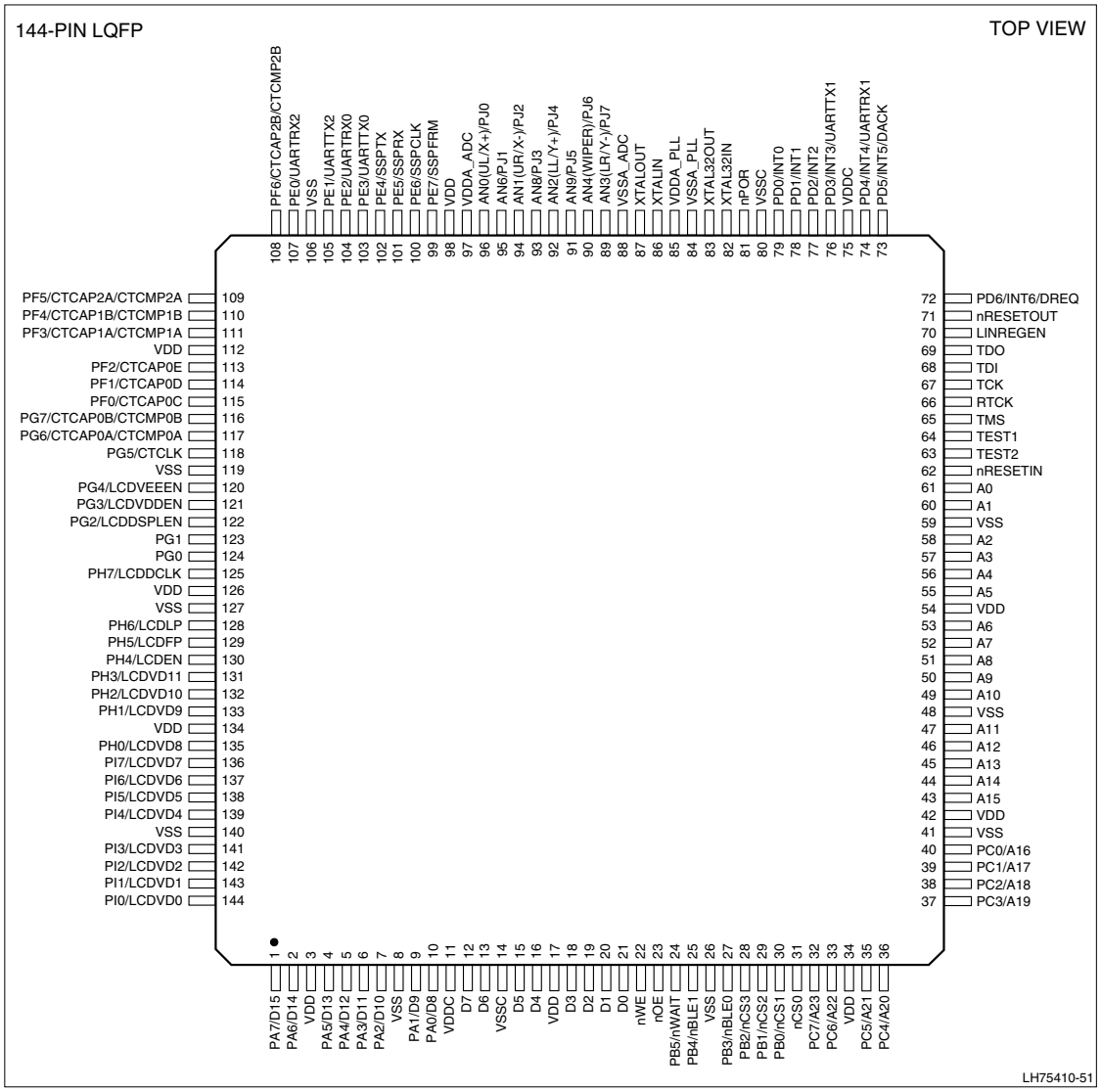


Figure 5-3. LH75410 Pin Diagram

5.5 LH75410 Numerical Pin Listing

Table 5-1. LH75410 Numerical Pin List

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
1	PA7	D15		I/O	8 mA	Bidirectional		1
2	PA6	D14		I/O	8 mA	Bidirectional		1
3	VDD			Power	None			
4	PA5	D13		I/O	8 mA	Bidirectional		1
5	PA4	D12		I/O	8 mA	Bidirectional		1
6	PA3	D11		I/O	8 mA	Bidirectional		1
7	PA2	D10		I/O	8 mA	Bidirectional		1
8	VSS			Ground	None			
9	PA1	D9		I/O	8 mA	Bidirectional		1
10	PA0	D8		I/O	8 mA	Bidirectional		1
11	VDDC			Power	None			
12	D7			I/O	8 mA	Bidirectional		
13	D6			I/O	8 mA	Bidirectional		
14	VSSC			Ground	None			
15	D5			I/O	8 mA	Bidirectional		
16	D4			I/O	8 mA	Bidirectional		
17	VDD			Power	None			
18	D3			I/O	8 mA	Bidirectional		
19	D2			I/O	8 mA	Bidirectional		
20	D1			I/O	8 mA	Bidirectional		
21	D0			I/O	8 mA	Bidirectional		
22	nWE				8 mA	Output		3
23	nOE				8 mA	Output		3
24	PB5	nWAIT			8 mA	Bidirectional	Pull-up	1
25	PB4	nBLE1			8 mA	Bidirectional	Pull-up	1
26	VSS			Ground	None			
27	PB3	nBLE0			8 mA	Bidirectional	Pull-up	1
28	PB2	nCS3			8 mA	Bidirectional	Pull-up	1
29	PB1	nCS2			8 mA	Bidirectional	Pull-up	1
30	PB0	nCS1			8 mA	Bidirectional	Pull-up	1
31	nCS0				8 mA	Output		3
32	PC7	A23			8 mA	Bidirectional	Pull-down	1
33	PC6	A22			8 mA	Bidirectional	Pull-down	1
34	VDD			Power	None			
35	PC5	A21			8 mA	Bidirectional	Pull-down	1
36	PC4	A20			8 mA	Bidirectional	Pull-down	1
37	PC3	A19			8 mA	Bidirectional	Pull-down	1
38	PC2	A18			8 mA	Bidirectional	Pull-down	1
39	PC1	A17			8 mA	Bidirectional	Pull-down	1

Table 5-1. LH75410 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
40	PC0	A16			8 mA	Bidirectional	Pull-down	1
41	VSS			Ground	None			
42	VDD			Power	None			
43	A15				8 mA	Output		
44	A14				8 mA	Output		
45	A13				8 mA	Output		
46	A12				8 mA	Output		
47	A11				8 mA	Output		
48	VSS			Ground	None			
49	A10				8 mA	Output		
50	A9				8 mA	Output		
51	A8				8 mA	Output		
52	A7				8 mA	Output		
53	A6				8 mA	Output		
54	VDD			Power	None			
55	A5				8 mA	Output		
56	A4				8 mA	Output		
57	A3				8 mA	Output		
58	A2				8 mA	Output		
59	VSS			Ground	None			
60	A1				8 mA	Output		
61	A0				8 mA	Output		
62	nRESETIN				None	Input	Pull-up	2, 3
63	TEST2				None	Input	Pull-up	2
64	TEST1				None	Input	Pull-up	2
65	TMS				None	Input	Pull-up	2
66	RTCK				8 mA	Output		
67	TCK				None	Input		
68	TDI				None	Input	Pull-up	2
69	TDO				4 mA	Output		
70	LINREGEN				None	Input		
71	nRESETOUT				8 mA	Output		3
72	PD6	INT6	DREQ		6 mA	Bidirectional	Pull-down	1
73	PD5	INT5	DACK		6 mA	Bidirectional		1, 2
74	PD4	INT4	UARTRX1		8 mA	Bidirectional	Pull-up	1
75	VDDC			Power	None			
76	PD3	INT3	UARTTX1		8 mA	Bidirectional	Pull-up	1
77	PD2	INT2			2 mA	Bidirectional	Pull-up	1
78	PD1	INT1			6 mA	Bidirectional		1, 2
79	PD0	INT0			2 mA	Bidirectional		1
80	VSSC			Ground	None			

Table 5-1. LH75410 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
81	nPOR				None	Input	Pull-up	2, 3
82	XTAL32IN				None	Input		
83	XTAL32OUT				None	Output		
84	VSSA_PLL			Ground	None			
85	VDDA_PLL			Power	None			
86	XTALIN				None	Input		
87	XTALOUT				None	Output		
88	VSSA_ADC			Ground	None			
89	AN3 (LR/Y-)	PJ7			None	Input		
90	AN4 (Wiper)	PJ6			None	Input		
91	AN9	PJ5			None	Input		
92	AN2 (LL/Y+)	PJ4			None	Input		
93	AN8	PJ3			None	Input		
94	AN1 (UR/X-)	PJ2			None	Input		
95	AN6	PJ1			None	Input		
96	AN0 (UL/X+)	PJ0			None	Input		
97	VDDA_ADC			Power	None			
98	VDD			Power	None			
99	PE7	SSPFRM			4 mA	Bidirectional	Pull-up	1
100	PE6	SSPCLK			4 mA	Bidirectional	Pull-down	1
101	PE5	SSPRX			4 mA	Bidirectional	Pull-up	1
102	PE4	SSPTX			4 mA	Bidirectional	Pull-down	1
103	PE3	UARTTX0			8 mA	Bidirectional	Pull-up	1
104	PE2	UARTRX0			2 mA	Bidirectional	Pull-up	1
105	PE1	UARTTX2			4 mA	Bidirectional	Pull-up	1
106	VSS			Ground	None			
107	PE0	UARTRX2			4 mA	Bidirectional	Pull-up	1
108	PF6	CTCAP2B	CTCMP2B		4 mA	Bidirectional		2
109	PF5	CTCAP2A	CTCMP2A		4 mA	Bidirectional		
110	PF4	CTCAP1B	CACMP1B		4 mA	Bidirectional		2
111	PF3	CTCAP1A	CTCMP1A		4 mA	Bidirectional		
112	VDD			Power	None			
113	PF2	CTCAP0E			4 mA	Bidirectional		2
114	PF1	CTCAP0D			4 mA	Bidirectional		
115	PF0	CTCAP0C			4 mA	Bidirectional		2
116	PG7	CTCAP0B	CTCMP0B		4 mA	Bidirectional		
117	PG6	CTCAP0A	CTCMP0A		4 mA	Bidirectional		2
118	PG5	CTCLK			4 mA	Bidirectional		
119	VSS			Ground	None			
120	PG4	LCDVEEEN			8 mA	Bidirectional		
121	PG3	LCDVDDEN			8 mA	Bidirectional		

Table 5-1. LH75410 Numerical Pin List (Cont'd)

PIN NO.	FUNCTION AT RESET	FUNCTION 2	FUNCTION 3	FUNCTION TYPE	OUTPUT DRIVE	BUFFER TYPE	PULL-UP/PULL-DOWN AT RESET	NOTES
122	PG2	LCDDSPLEN			8 mA	Bidirectional		
123	PG1				8 mA	Bidirectional		
124	PG0				8 mA	Bidirectional		
125	PH7	LCDDCLK			8 mA	Bidirectional		
126	VDD			Power	None			
127	VSS			Ground	None			
128	PH6	LCDLP			8 mA	Bidirectional		
129	PH5	LCDFP			8 mA	Bidirectional		
130	PH4	LCDEN			8 mA	Bidirectional		
131	PH3	LCDVD11			8 mA	Bidirectional		
132	PH2	LCDVD10			8 mA	Bidirectional		
133	PH1	LCDVD9			8 mA	Bidirectional		
134	VDD			Power	None			
135	PH0	LCDVD8			8 mA	Bidirectional		
136	PI7	LCDVD7			8 mA	Bidirectional		
137	PI6	LCDVD6			8 mA	Bidirectional		
138	PI5	LCDVD5			8 mA	Bidirectional		
139	PI4	LCDVD4			8 mA	Bidirectional		
140	VSS			Ground	None			
141	PI3	LCDVD3			8 mA	Bidirectional		
142	PI2	LCDVD2			8 mA	Bidirectional		
143	PI1	LCDVD1			8 mA	Bidirectional		
144	PI0	LCDVD0			8 mA	Bidirectional		

NOTES:

1. Signal is selectable as pull-up, pull-down, or no pull-up/pull-down via the I/O Configuration peripheral.
2. CMOS Schmitt trigger input.
3. Signals preceded with 'n' are active LOW.

5.6 LH75410 Signal Descriptions

Table 5-2. LH75410 Signal Descriptions

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
MEMORY INTERFACE (MI)				
1 2 4 5 6 7 9 10 12 13 15 16 18 19 20 21	D[15:0]	Input/Output	Data Input/Output Signals	1
22	nWE	Output	Static Memory Controller Write Enable	2
23	nOE	Output	Static Memory Controller Output Enable	2
24	nWAIT	Input	Static Memory Controller External Wait Control	1, 2
25	nBLE1	Output	Static Memory Controller Byte Lane Strobe	1, 2
27	nBLE0	Output	Static Memory Controller Byte Lane Strobe	1, 2
28	nCS3	Output	Static Memory Controller Chip Select	1, 2
29	nCS2	Output	Static Memory Controller Chip Select	1, 2
30	nCS1	Output	Static Memory Controller Chip Select	1, 2
31	nCS0	Output	Static Memory Controller Chip Select	2
32 33 35 36 37 38 39 40 43 44 45 46 47 49 50 51 52 53 55 56 57 58 60 61	A[23:0]	Output	Address Signals	1
DMA CONTROLLER (DMAC)				
72	DREQ	Input	DMA Request	1
73	DACK	Output	DMA Acknowledge	1

Table 5-2. LH75410 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
LCD CONTROLLER (LCDC)				
120	LCDVEEEN	Output	Analog Supply Enable (AC Bias SIgnal)	1
121	LCDVDDEN	Output	Digital Supply Enable	1
122	LCDDSPLEN	Output	LCD Panel Power Enable	1
125	LCDDCLK	Output	LCD Panel Clock	1
128	LCDLP	Output	Line Synchronization Pulse (STN), Horizontal Synchronization Pulse (TFT)	1
129	LCDFP	Output	Frame Pulse (STN), Vertical Synchronization Pulse (TFT)	1
130	LCDEN	Output	LCD Data Enable	1
131 132 133 135 136 137 138 139 141 142 143 144	LCDVD[11:0]	Output	LCD Panel Data bus	1
SYNCHRONOUS SERIAL PORT (SSP)				
99	SSPFRM	Input	SSP Serial Frame	1
100	SSPCLK	Input	SSP Clock	1
101	SSPRX	Input	SSP RXD	1
102	SSPTX	Output	SSP TXD	1
UART0 (U0)				
103	UARTTX0	Output	UART0 Transmitted Serial Data Output	1
104	UARTRX0	Input	UART0 Received Serial Data Input	1
UART1 (U1)				
74	UARTRX1	Input	UART1 Received Serial Data Input	1
76	UARTTX1	Output	UART1 Transmitted Serial Data Output	1
UART2 (U2)				
105	UARTTX2	Output	UART2 Transmitted Serial Data Output	1
107	UARTRX2	Input	UART2 Received Serial Data Input	1
ANALOG-TO-DIGITAL CONVERTER (ADC)				
89 90 91 92 93 94 95 96	AN3 (LR/Y-) AN4 (Wiper) AN9 AN2 (LL/Y+) AN8 AN1 (UR/X-) AN6 AN0 (UL/X+)	Input	ADC Inputs	1
TIMER 0				
117 116 115 114 113	CTCAP0[A:E]	Input	Timer 0 Capture Inputs	1
117 116	CTCMP0[A:B]	Output	Timer 0 Compare Outputs	1

Table 5-2. LH75410 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
118	CTCLK	Input	Common External Clock	1
TIMER 1				
111 110	CTCAP1[A:B]	Input	Timer 1 Capture Inputs	1
111 110	CTCMP1[A:B]	Output	Timer 1 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
TIMER 2				
109 108	CTCAP2[A:B]	Input	Timer 2 Capture Inputs	1
109 108	CTCMP2[A:B]	Input	Timer 2 Compare Outputs	1
118	CTCLK	Input	Common External Clock	1
GENERAL PURPOSE INPUT/OUTPUT (GPIO)				
1 2 4 5 6 7 9 10	PA7 PA6 PA5 PA4 PA3 PA2 PA1 PA0	Input/Output	General Purpose I/O Signals - Port A	1
24 25 27 28 29 30	PB5 PB4 PB3 PB2 PB1 PB0	Input/Output	General Purpose I/O Signals - Port B	1
32 33 35 36 37 38 39 40	PC7 PC6 PC5 PC4 PC3 PC2 PC1 PC0	Input/Output	General Purpose I/O Signals - Port C	1
72 73 74 76 77 78 79	PD6 PD5 PD4 PD3 PD2 PD1 PD0	Input/Output	General Purpose I/O Signals - Port D	1
89 90 91 92 93 94 95 96	PJ7 PJ6 PJ5 PJ4 PJ3 PJ2 PJ1 PJ0	Input	General Purpose I/O Signals - Port J	1

Table 5-2. LH75410 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
99 100 101 102 103 104 105 107	PE7 PE6 PE5 PE4 PE3 PE2 PE1 PE0	Input/Output	General Purpose I/O Signals - Port E	1
108 109 110 111 113 114 115	PF6 PF5 PF4 PF3 PF2 PF1 PF0	Input/Output	General Purpose I/O Signals - Port F	1
116 117 118 120 121 122 123 124	PG7 PG6 PG5 PG4 PG3 PG2 PG1 PG0	Input/Output	General Purpose I/O Signals - Port G	1
125 128 129 130 131 132 133 135	PH7 PH6 PH5 PH4 PH3 PH2 PH1 PH0	Input/Output	General Purpose I/O Signals - Port H	1
136 137 138 139 141 142 143 144	PI7 PI6 PI5 PI4 PI3 PI2 PI1 PI0	Input/Output	General Purpose I/O Signals - Port I	1
RESET, CLOCK, AND POWER CONTROLLER (RCPC)				
62	nRESETIN	Input	User Reset Input	2
71	nRESETOUT	Output	System Reset Output	2
72	INT6	Input	External Interrupt Input 6	1
73	INT5	Input	External Interrupt Input 5	1
74	INT4	Input	External Interrupt Input 4	1
76	INT3	Input	External Interrupt Input 3	1
77	INT2	Input	External Interrupt Input 2	1
78	INT1	Input	External Interrupt Input 1	1
79	INT0	Input	External Interrupt Input 0	1
81	nPOR	Input	Power-on Reset Input	2
82	XTAL32IN	Input	32.768 kHz Crystal Clock Input	
83	XTAL32OUT	Output	32.768 kHz Crystal Clock Output	
86	XTALIN	Input	Crystal Clock Input	
87	XTALOUT	Output	Crystal Clock Output	

Table 5-2. LH75410 Signal Descriptions (Cont'd)

PIN NO.	SIGNAL NAME	TYPE	DESCRIPTION	NOTES
TEST INTERFACE				
63	TEST2	Input	Test Mode Pin 2	
64	TEST1	Input	Test Mode Pin 1	
65	TMS	Input	JTAG Test Mode Select Input	
66	RTCK	Output	Returned JTAG Test Clock Output	
67	TCK	Input	JTAG Test Clock Input	
68	TDI	Input	JTAG Test Serial Data Input	
69	TDO	Output	JTAG Test Data Serial Output	
POWER AND GROUND (GND)				
3 17 34 42 54 98 112 126 134	VDD	Power	I/O Ring VDD	
8 26 41 48 59 106 119 127 140	VSS	Power	I/O Ring VSS	
11 75	VDDC	Power	Core VDD supply (Output if Linear Regulator Enabled, Otherwise Input)	
14 80	VSSC	Power	Core VSS	
70	LINREGEN	Input	Linear Regulator Enable	
84	VSSA_PLL	Power	PLL Analog VSS	
85	VDDA_PLL	Power	PLL Analog VDD Supply	
88	VSSA_ADC	Power	A-to-D converter Analog VSS	
97	VDDA_ADC	Power	A-to-D converter Analog VDD Supply	

NOTES:
1. These pins have multiplexed functions.
2. Signals preceded with 'n' are active LOW.

Chapter 6
Memory Interface Architecture

The SHARP BlueStreak LH75400/01/10/11 devices provide the following data-path management resources on chip:

- AHB and APB data buses
- 16KB of zero-wait-state Tightly Coupled Memory (TCM) SRAM accessible via processor only
- 16KB of internal SRAM accessible via processor, DMA, and LCD Controller
- A Static Memory Controller with a 24-bit address, 16-bit data interface, and four chip selects (the Static Memory Controller controls access to external memory)
- A 4-stream general purpose DMA Controller.

All system resources are memory-mapped. These include external resources, such as:

- Read Only Memory (ROM)
- Programmable ROM (PROM)
- SRAM
- External peripherals.

These also include internal resources, such as:

- System configuration registers
- Peripheral configuration registers
- TCM and internal SRAM.

The first partitioning of memory space is its subdivision into eight ‘segments’. Each segment spans 512MB. The start address of each segment is fixed and determined by the three highest order bits of the 32-bit AHB address. These segments define the type of resource being addressed. For example:

- One segment can contain only external devices connected to the External Bus Interface.
- Another segment can contain only the internal SRAM connected to the AHB.
- Another segment is reserved for accessing the system configuration registers themselves, as well as many of the peripheral control registers.
- Another segment contains the TCM SRAM connected to the ARM7 local bus. This memory is only accessible via the processor and any attempt by the DMA Controller or LCD Controller to access this segment causes a bus error to occur. See Table 6-1.

This memory map partition has three views, based on the setting of the REMAP bits in the Reset, Clock, and Power Controller. See Chapter 9.

Table 6-1. Memory Mapping

ADDRESS	REMAP = 0x0 (DEFAULT)	REMAP = 0x1	REMAP = 0x2
0x00000000	External Memory	TCM SRAM	Internal SRAM
0x20000000	Reserved	Reserved	Reserved
0x40000000	External Memory	External Memory	External Memory
0x60000000	Internal SRAM	Internal SRAM	Internal SRAM
0x80000000	TCM SRAM	TCM SRAM	TCM SRAM
0xA0000000	Reserved	Reserved	Reserved
0xC0000000	Reserved	Reserved	Reserved
0xE0000000 - 0xFFFBFFFF	Reserved	Reserved	Reserved

NOTES:

1. REMAP is initialized to '0x0' upon System Reset. Setting REMAP to 0x1 on start-up can prevent applications displayed on the LCD from working; (see the LCD chapter).
2. Right after System Reset or when REMAP = 0x0, external memory is mapped to lower memory (0x00000000 - 0x1FFFFFFF), which means the same physical memory can be accessed from two locations: 0x00000000 and 0x40000000.
3. Programming REMAP to '0x1' will map TCM SRAM to lower memory, which means the same physical memory can be accessed from two locations: 0x00000000 and 0x60000000.
4. Programming REMAP to '0x2' will map internal SRAM to lower memory, which means the same physical memory can be accessed from two locations: 0x00000000 and 0x80000000, but only by the ARM processor.
5. Programming REMAP to '0x3' is reserved and will cause any access to lower memory to result in a memory abort.
6. Do not specify TCM as either the frame buffer or as the source or destination for DMA transfers. Setting REMAP = 0x1 causes TCM to be aliased to the address range 0x00000000-0x00003FFF. Exercise care when setting DMA transfer or frame buffer addresses to that address range. Note that the UPBASE and LPBASE Registers must also point to a memory location accessible on the AHB before the CLCDC is enabled; otherwise, the CLCDC locks up. In particular, if REMAP = 0x1, the default value of the UPBASE and LPBASE of 0x00000000 causes the CLCDC to try to access TCM. If TCM is mapped to 0 (REMAP = 0x1), initialize UPBASE and LPBASE before enabling the CLCDC.

The second partitioning of memory space is the dividing of the segments into sections. The external memory segment is divided into eight 64MB sections, of which the first four are used, each having a chip select associated with it. Access to any of the last four sections does not result in an external bus access and does not cause a memory abort. The peripheral register segment is divided into 4KB peripheral sections, 21 of which are assigned to the peripherals. See Table 6-2 through Table 6-4. In Table 6-2, 'XX' represents a valid REMAP value ('0x0', '0x1', or '0x2').

Table 6-2. External Memory Section Mapping

START ADDRESS		DEVICE
REMAP = XX	REMAP = 0x0	
0x40000000	0x00000000	Chip Select 0
0x44000000	0x04000000	Chip Select 1
0x48000000	0x08000000	Chip Select 2
0x4C000000	0x0C000000	Chip Select 3
0x50000000	0x10000000	Invalid Access
0x54000000	0x14000000	Invalid Access
0x58000000	0x18000000	Invalid Access
0x5C000000	0x1C000000	Invalid Access

Table 6-3. Primary AHB Peripheral Register Mapping

ADDRESS RANGE	DEVICE
0xFFFC0000 - 0xFFFEFFFF	APB Bridge
0xFFFF0000 - 0xFFFF0FFF	Reserved
0xFFFF1000 - 0xFFFF1FFF	Ext. Static Mem Controller
0xFFFF2000 - 0xFFFF3FFF	Reserved
0xFFFF4000 - 0xFFFF4FFF	Color LCD Controller (LH75401 and LH75411) LCD Controller (LH75400 and LH75410)
0xFFFF5000 - 0xFFFFEFFF	Reserved
0xFFFFF000 - 0XFFFFFFFF	Interrupt Controller

Table 6-4. APB Peripheral Register Mapping

ADDRESS RANGE	DEVICE
0xFFFC0000 - 0xFFFC0FFF	UART0 (16550)
0xFFFC1000 - 0xFFFC1FFF	UART1 (16550)
0xFFFC2000 - 0xFFFC2FFF	UART2 (82510)
0xFFFC3000 - 0xFFFC3FFF	Analog-to-Digital Converter
0xFFFC4000 - 0xFFFC4FFF	Timer Module
0xFFFC5000 - 0xFFFC5FFF	CAN (LH75401 and LH75400) Reserved (LH75411 and LH75410)
0xFFFC6000 - 0xFFFC6FFF	Synchronous Serial Port
0xFFFC7000 - 0xFFFDAFFF	Reserved
0xFFFDB000 - 0xFFFDBFFF	GPIO4
0xFFFDC000 - 0xFFFDCFFF	GPIO3
0xFFFDD000 - 0xFFFDDFFF	GPIO2
0xFFFDE000 - 0xFFFDEFFF	GPIO1
0xFFFDF000 - 0xFFFDFFFF	GPIO0
0xFFFE0000 - 0xFFFE0FFF	Real Time Clock
0xFFFE1000 - 0xFFFE1FFF	DMA Controller
0xFFFE2000 - 0xFFFE2FFF	Reset Clock and Power Controller
0xFFFE3000 - 0xFFFE3FFF	Watchdog Timer
0xFFFE4000 - 0xFFFE4FFF	Advanced LCD Interface
0xFFFE5000 - 0xFFFE5FFF	I/O Configuration Peripheral
0xFFFE6000 - 0xFFFEFFFF	Reserved

Chapter 7
Static Memory Controller

The Static Memory Controller (SMC) is an AMBA AHB slave peripheral. The SMC interfaces the SoC to external memory devices.

The SMC supports four banks of external memory. Each bank has a maximum size of 16MB. The ARM system supports 32 bits of address space. The base address of SMC-controlled memory space is set by bits [31:28] of the address. These bits do not pass to the SMC. Instead, the AHB arbiter uses them to identify that a system bus cycle is for the SMC.

During an external memory cycle, the bank being accessed is selected by the assertion of the appropriate nCSx signal. The nCSx signal being asserted can be ascertained by decoding bits [27:26]. For example, a value of '0' for these bits causes nCS0 to be asserted, a value of '3' causes nCS3 to be asserted, and so on. Bits [23:0] are passed as the address to the memory bank.

Table 7-1. Address Bus Organization

31:28	27:26	25:24	23:0
Base address for memory bank	Chip select address space for four memory banks	Unused	16MB memory bank address space

7.1 SMC Features

The SMC is programmed through the AHB. Each memory bank has its own Configuration Register, SMCBCR[3:0]. This register allows each bank to be configured independently to:

- Support memory-mapped devices including Random Access Memory (RAM), ROM, Flash, and burst ROM
- Vary the external bus width (8 or 16 bits wide)
- Vary the external device width (8 or 16 bits wide)
- Asynchronous Burst Mode read access to Burst Mode ROM devices
- Vary the number of wait states from 1 to 32 (independently for read and write accesses)
- Vary the number of wait states in the first access (from 1 to 32), then each subsequent access in a burst read access from a Burst Mode ROM (from 0 to 31)
- Wait states may be extended indefinitely by an external hardware pin (nWAIT)
- Vary the bus turnaround cycles (1 to 16) allowed between a read operation and a write operation
- Place the bank under write protection.

7.2 SMC Theory of Operation

After power-up, the SoC provides a single-chip select (nCS0) and 16 address bits (A[15:0]). The SoC can be reprogrammed to exchange GPIO for up to three more chip selects (nCS3 to nCS1) and up to eight more address bits (A[23:16]). The data port defaults to 16 bits (boot ×16) after Reset. However, it can be set to 8 bits after reset by setting the boot × 8 option. This can be expanded by reducing the number of GPIO available. See Chapter 21 for more details.

From an external perspective, the external memory bus cycle starts with the assertion of nCSx and ends when nCSx is de-asserted. The following sections describe the internal operations of the SMC.

7.2.1 SMC Write Process

The SMC has the same write control for all devices connected to it. Figure 7-1 shows a write bus cycle. The following steps describe the write process.

1. The SMC detects that an AHB bus cycle contains a transaction for it.
2. The AHB bus transaction information informs the SMC that it is a write transaction. The AHB bus transaction also contains the address of the transaction (all AHB bus transactions are memory-mapped).

   The SMC does not rely on asynchronously controlled timing of signals to ensure correct operation of the external memories. Rather, timing relationships are ensured by timing the interface signals with the System Clock signal. This approach makes the timing characteristics of the interface dependent on the clock rates used.
3. The SMC immediately places address bits [23:0] on the external A[23:0] address bus. At the same time, it receives the write data from the AHB. (Being a pipelined bus, the AHB presents the data on the clock after the transaction is started and the address is presented.)
4. On the next clock, the SMC asserts the appropriate nCSx signal, nBLEx signal (byte lane enable), and the Write Enable signal (nWEN). It also presents the data to be written to the SRAM on D[15:0]**.**

   The SMC supports 1 to 32 wait states. It also supports an nWAIT input that can be used by an external device to vary the wait time.
5. The system clock edge following the one during which the nCSx signal is asserted counts as the end of the first wait state. If there are N wait states programmed for this bank of memory, the SMC will hold this state for either N system clocks or until nWAIT is sampled as being inactive on an system clock edge — whichever happens last.
6. The nCSx signal and the nBLEx (byte lane enables) are removed during the following system clock cycle.
7. The A[23:0], D[15:0], and nCSx signals are removed on the following clock. The system AHB is not held during the write. Up to 4 writes can be buffered in the SMC for future execution.

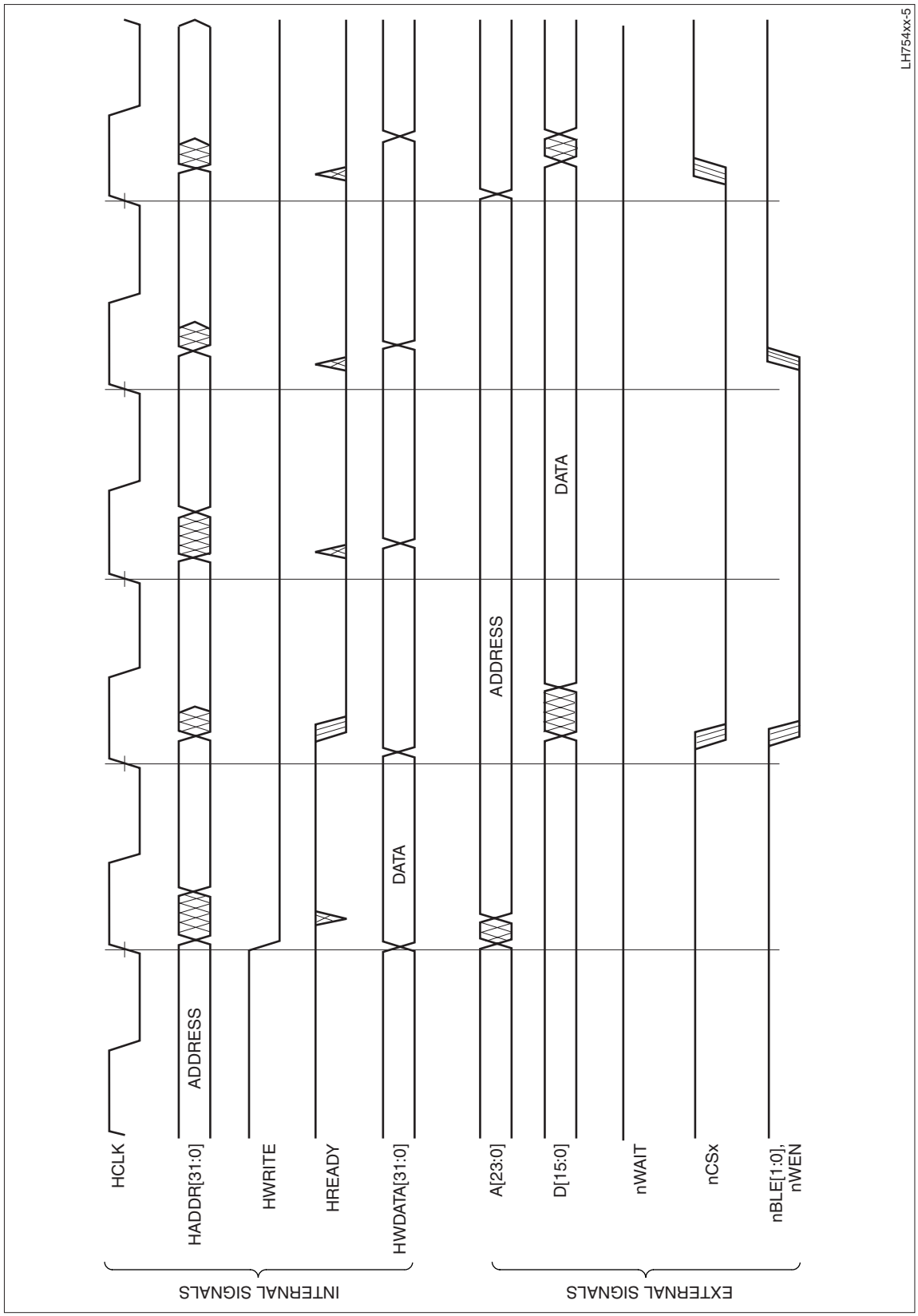


Figure 7-1. SMC Write Access

There are a number of variations to the write process:

- The nCSx signal will be de-asserted one cycle earlier if the next cycle is to be a write (see Figure 7-2).
- If the external bus is not as wide as the data in the transaction, the SMC buffers the data and issues further external write cycles, modifying the address as needed to complete the transaction.
- If the given memory bank is write-protected, the SMC does not try to perform the write; rather, it returns an error status on the AHB to indicate to the AHB bus master that initiated the transaction that the transaction failed. ROM banks are normally write-protected; however, any type of memory bank can be write-protected.

7.2.2 SMC Read Process

The SMC read process, shown in Figure 7-3, is similar to the SMC write process, except that no data is presented. The nCSx signal is asserted on the same cycle and the address is placed on A[23:0]. Each SMC bank has a programmable number of wait states.

The SMC also supports an nWAIT input that an external device can use to vary the wait time. If there are N ($1 < N < 32$) wait states, the SMC holds this state for either N clock cycles or until nWAIT is sampled as being inactive — whichever happens last. The SMC captures the data on the following edge of the system clock following the assertion of nCSx. The nCSx signal and the address are removed one cycle later. If the external memory interface is not as wide as the AHB transfer request, the SMC issues successive external read bus cycles and buffers the data before it is presented to the AHB. The AHB remains unavailable for any other purpose until the read request has been completed.

7.2.3 SMC Burst Mode Read Process

The SMC supports a Burst Mode, shown in Figure 7-4. This mode supports sequential access burst reads of up to four consecutive locations in 8- or 16-bit memories. This feature supports Burst Mode ROM devices and increases the bandwidth by using a reduced (configurable) access time for three sequential reads following a quad-location boundary read. The number of wait states for the initial read, and for subsequent reads, are separately programmable.

Quad-location boundaries occur when HADDR[1:0] = 0b00 for byte-wide memories or when HADDR[1] = 0b0 for halfword memories (refer to Table 7-2).

Accesses must be made on appropriate word boundaries. A bus error occurs:

- If a 16-bit access is made on an odd address, or
- If a 32-bit access is made on a 2-word boundary (A[1:0] = 0b10).

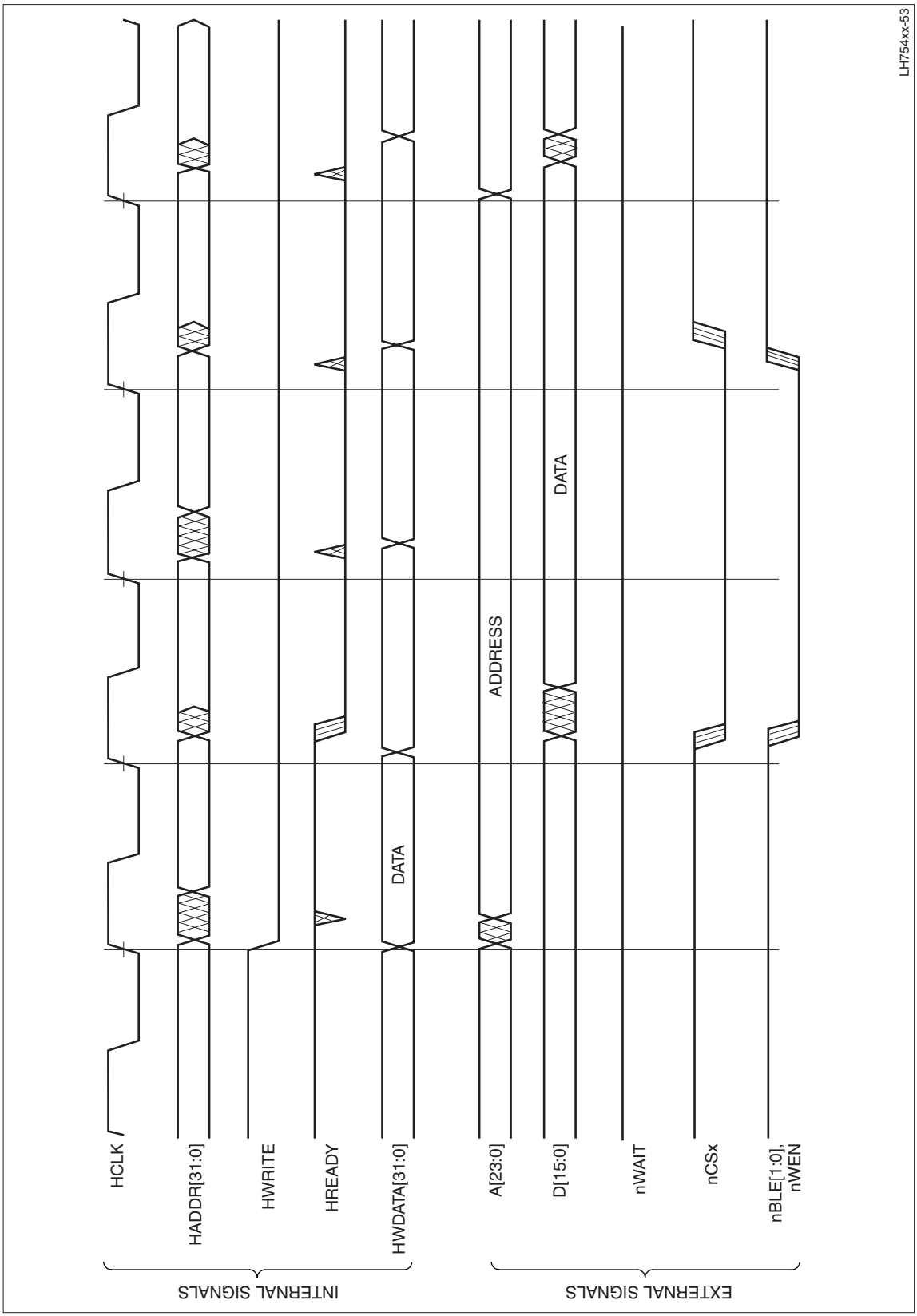


Figure 7-2. SMC Write, nCSx De-asserted Early

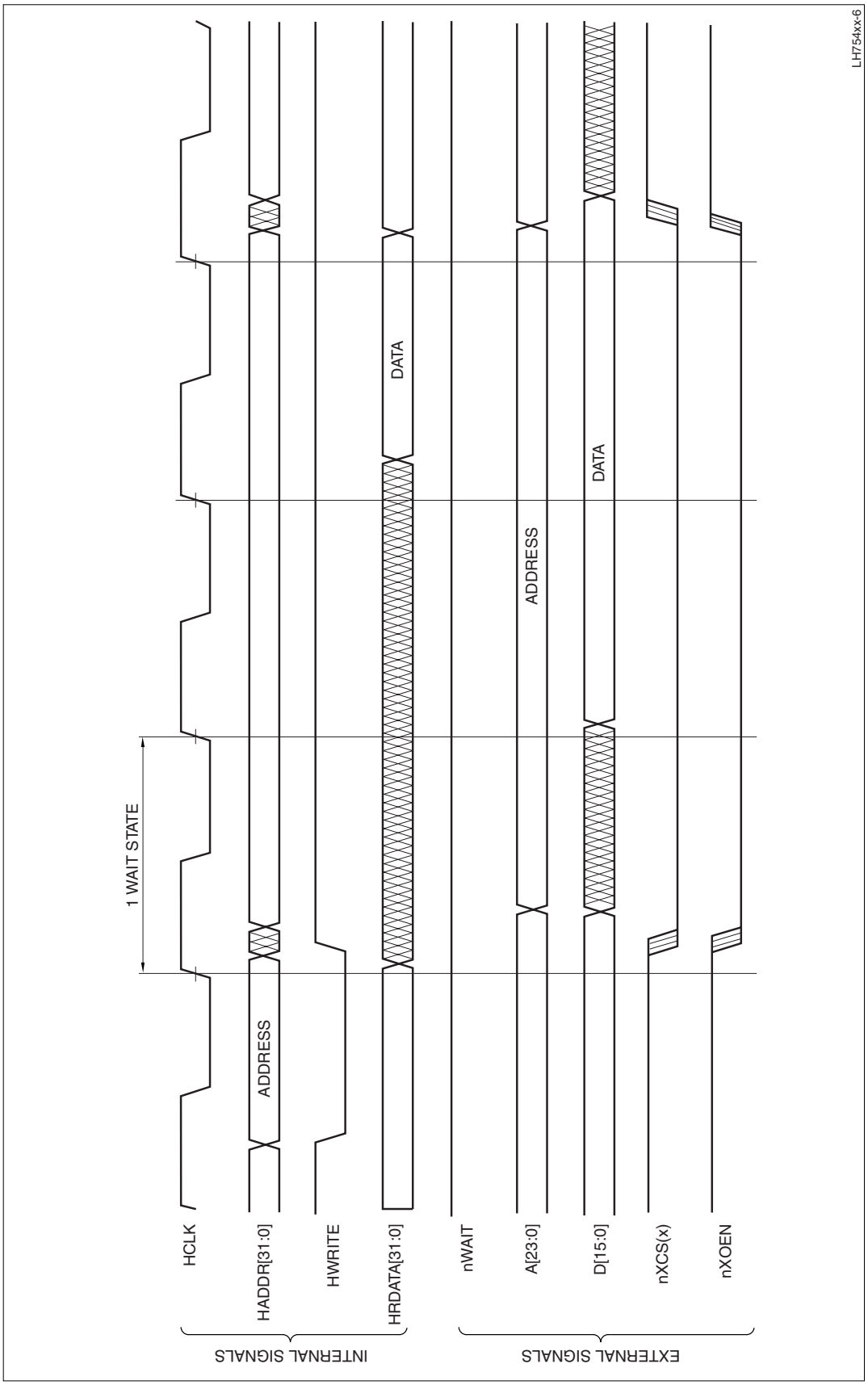


Figure 7-3. SMC Read Access

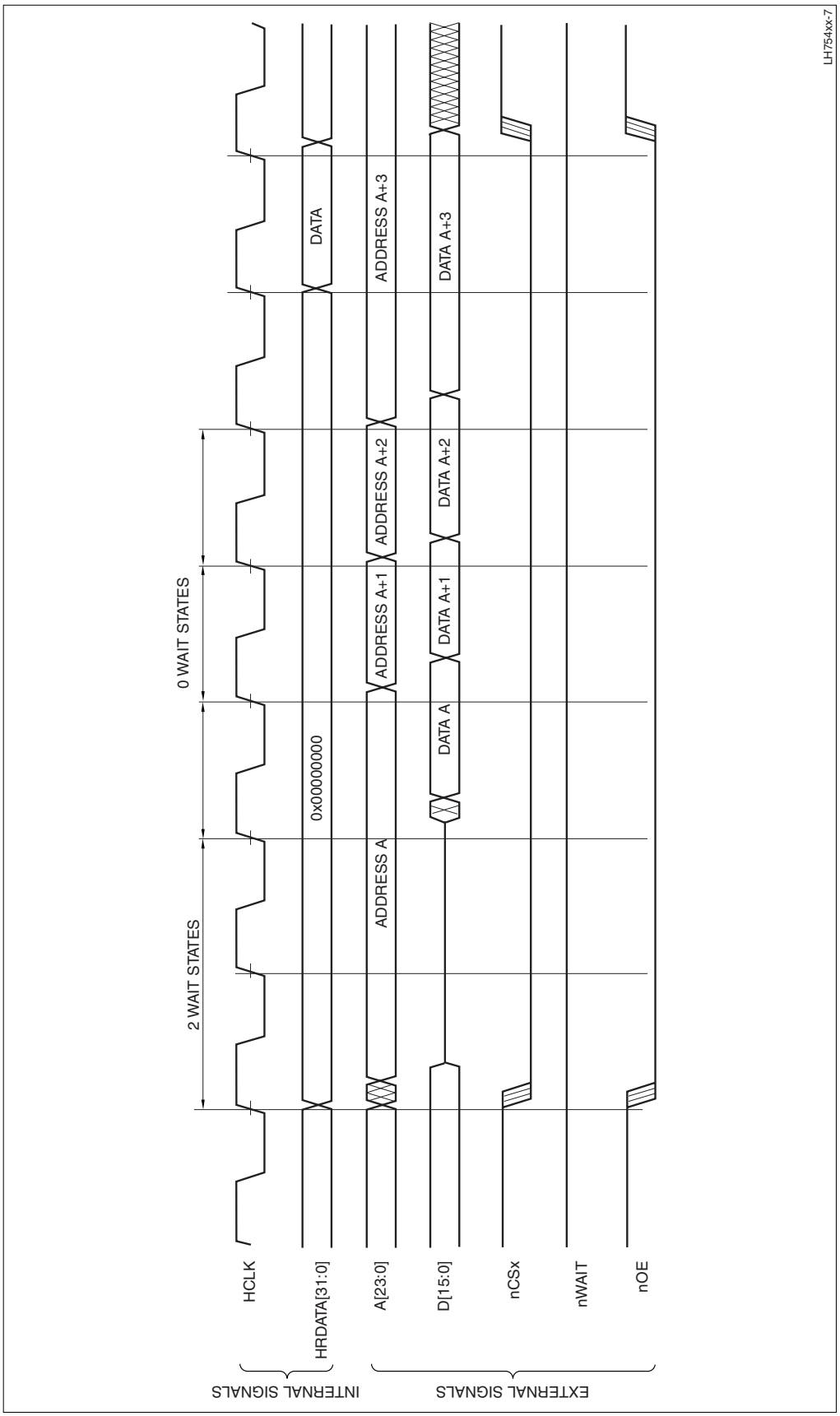


Figure 7-4. SMC Burst Read Access

7.2.4 External Memory Bus Cycle

As previously mentioned, the external memory bus cycle commences with the assertion of nCS(x) and ends when it is de-asserted. In addition to a programmable number of wait states for a read or write, the SoC supports a programmable number of wait states (from 1 to 32) for enabling the external bus to 'turn around' from a read to a write. This value can be programmed from 1 and 32. Table 7-2 lists the turnaround cycles for which the register values can be used.

Table 7-2. SMC Bus Turnaround Usage

EXTERNAL BUS TRANSFER	TURNAROUND CYCLE
Read from a memory bank, then write to the same memory bank	Generated
Read from a memory bank, then write to a different memory bank	Generated
Write to a memory bank, then write to the same memory bank	Not generated
Write to a memory bank, then write to a different memory bank	Generated
Write to a memory bank, then read from the same memory bank	Not generated
Write to a memory bank, then read from a different memory bank	Generated

NOTE: All SMC operations assume a Little Endian memory operation.

The Byte Lane Enable signals are used to control instances when:

- Data transfers are smaller than the width of the memory devices being used.
- Memory is configured to be wider than the memory devices that it is made from.

The Byte Lane Enable signals can be programmed to be either all active or all inactive during reads. The SMC performs the mapping to ensure that each byte read is at the correct location in the system bus. During writes, the nBLE[1:0] signals:

- Ensure that only the external device in a memory bank for which the data is intended will perform the write
- Direct the device to 'steer' the data to the correct portion of its memory, if the external device is wider than the data.

As this shows, the SMC considers many factors when generating the nBLE[1:0] signals. These include:

- AMBA transfer width
- External memory bank data bus width
- External memory bank type, being byte, halfword, or word
- The decoded HADDR[2:0] value for write accesses only.

7.2.5 External Bus Read/Write Operations

Table 7-3 and Table 7-4 show how the SMC places data from the external bus onto the AHB based on the specified AHB control signals. Table 7-5 and Table 7-6 show how the SMC places data onto the external data bus and exercises the nBLE[1:0] signals based on the AHB control signals. During reads, the nBLE[1:0] signals are both LOW or HIGH, depending on the SMC configuration.

Table 7-3. 8-bit External Bus Read

ACCESS: READ, 8-BIT EXTERNAL BUS				EXTERNAL DATA MAPPING ONTO AHB DATA BUS			
INTERNAL TRANSFER WIDTH	**HSIZE[1:0]**	**HADDR[1:0]**	**A[1:0]**	**31:24**	**23:16**	**15:8**	**7:0**
Word (4 transfers)	10 10 10 10	xx xx xx xx	11 10 01 00	7:0 — — —	— 7:0 — —	— — 7:0 —	— — — 7:0
Halfword (2 transfers)	01	1x	11 10	7:0 —	— 7:0	— —	— —
Halfword (2 transfers)	01	0x	01 00	— —	— —	7.0 —	—
Byte	00	11	11	7:0	—	—	—
Byte	00	10	10	—	7:0	—	—
Byte	00	01	01	—	—	7:0	—
Byte	00	00	00	—	—	—	7:0

Table 7-4. 16-bit External Bus Read

ACCESS: READ, 8-BIT EXTERNAL BUS				EXTERNAL DATA MAPPING ONTO AHB DATA BUS			
INTERNAL TRANSFER WIDTH	**HSIZE[1:0]**	**HADDR[1:0]**	**A[1:0]**	**31:24**	**23:16**	**15:8**	**7:0**
Word (2 transfers)	10 10	xx xx	1x 0x	15:8 —	7:0 —	— 15:8	— 7:0
Halfword	01	1x	1x	15:8	7:0	—	—
Halfword	01	0x	0x	—	—	15:8	7:0
Byte	00	11	1x	15:8	—	—	—
Byte	00	10	1x	—	7:0	—	—
Byte	00	01	0x	—	—	15:8	—
Byte	00	00	0x	—	—	—	7:0

Table 7-5. 8-bit External Bus Write

ACCESS: WRITE, 8-BIT EXTERNAL BUS					SYSTEM DATA MAPPING ONTO EXTERNAL DATA BUS	
INTERNAL TRANSFER WIDTH	**HSIZE[1:0]**	**HADDR[1:0]**	**A[1:0]**	**nBLE[1:0]**	**15:8**	**7:0**
Word (4 transfers)	10 10 10 10	xx xx xx xx	11 10 01 00	10 10 10 10	— — — —	31:24 23:16 15:8 7:0
Halfword (2 transfers)	01	1x	11 10	10 10	— —	31:24 23:16
Halfword (2 transfers)	01	0x	01 00	10 10	— —	15:8 7:0
Byte	00	11	11	10	—	31:24
Byte	00	10	10	10	—	23:16
Byte	00	01	01	10	—	15:8
Byte	00	00	00	10	—	7:0

Table 7-6. 16-bit External Bus Write

ACCESS: WRITE, 16-BIT EXTERNAL BUS					SYSTEM DATA MAPPING ONTO EXTERNAL DATA BUS	
INTERNAL TRANSFER WIDTH	**HSIZE[1:0]**	**HADDR[1:0]**	**A[1:0]**	**nBLE[1:0]**	**15:8**	**7:0**
Word (2 transfers)	10 10	xx xx	1x 0x	00 00	31:24 15:8	23:16 7:0
Halfword	01	1x	1x	00	31:24	23:16
Halfword	01	0x	0x	00	15:8	7:0
Byte	00	11	1x	01	31:24	—
Byte	00	10	1x	10	—	23:16
Byte	00	01	0x	01	15:8	—
Byte	00	00	0x	10	—	7:0

7.2.6 SMC Memory Connection Diagram

Figure 7-5 shows connections for a typical memory system with different data width memory devices.

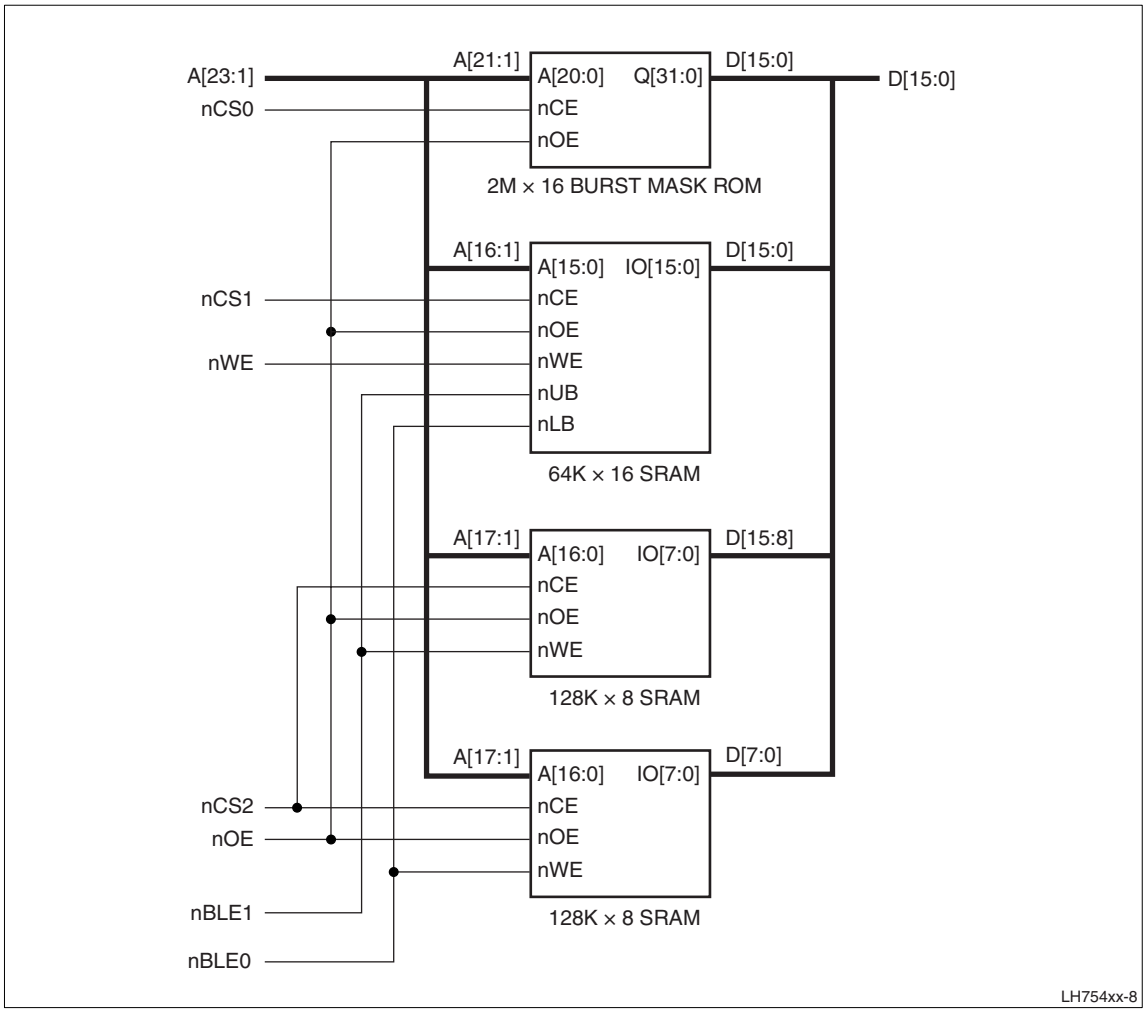


Figure 7-5. Typical Memory Connection Diagram

7.3 SMC Programmer's Model

The base address for the SMC external memory (SMC MemBase) is:

SMC MemBase Address: 0x40000000 (also 0x00000000 if REMAP is '0x0')

SMC memory banks have fixed address offsets from the base address.

Table 7-7. SMC Memory Bank Address Space

ADDRESS	DESCRIPTION
SMC MemBase + 0x00000000	SMC Memory Bank 0
SMC MemBase + 0x04000000	SMC Memory Bank 1
SMC MemBase + 0x08000000	SMC Memory Bank 2
SMC MemBase + 0x0C000000	SMC Memory Bank 3

7.3.1 SMC Register Summary

The base address for the SMC Control and Status Registers (SMC RegBase) is:

SMC RegBase: 0xFFFF1000

The SMC register banks have fixed offsets from this address.

Table 7-8. SMC Register Summary

NAME	ADDRESS OFFSET	TYPE	WIDTH	RESET VALUE	DESCRIPTION
BCR0	SMC RegBase + 0x00	RW	32	0x1000FFEF (16-bit) or 0x0000FBEF (8-bit)	Configuration Register for Memory Bank 0
BCR1	SMC RegBase + 0x04	RW	32	0x1000FFEF	Configuration Register for Memory Bank 1
BCR2	SMC RegBase + 0x08	RW	32	0x1000FFEF	Configuration Register for Memory Bank 2
BCR3	SMC RegBase + 0x0C	RW	32	0x1000FFEF	Configuration Register for Memory Bank 3

NOTE: The reset value of the first SMC base register depends on bus width. If PD2/INT2 is pulled HIGH on Reset, Bank 0 defaults to a 16-bit memory width. If PD2/INT2 is pulled LOW on Reset, Bank 0 defaults to an 8-bit memory width.

7.3.2 SMC Register Definitions

7.3.2.1 Configuration Register for Memory Bank 0

Register BCR0 has a reset value of either 0x1000FFEF (for 16-bit Mode) or 0x0000FBEF (for 8-bit Mode).

Table 7-9. BCR0 Register (16-bit Mode)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///		MW		BM	WP	WPERR	BUSERR	///							
RESET	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	WST2					RBLE	WST1					///	IDCY			
RESET	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW
ADDR	0xFFFF1000 + 0x00															

Table 7-10. BCR0 Register (8-bit Mode)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///		MW		BM	WP	WPERR	BUSERR	///							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	WST2					RBLE	WST1					///	IDCY			
RESET	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW
ADDR	0xFFFF1000 + 0x00															

Table 7-11. BCR0 Register Definitions

BITS	NAME	DESCRIPTION
31:30	///	**Reserved** Write the reset value.
29:28	MW	**Memory Width** 00 = 8-bit 01 = 16-bit 10 = Reserved 11 = Reserved
27	BM	**Burst Mode** 0 = Non-burst devices (Default) 1 = Burst ROM
26	WP	**Write Protect** 0 = SRAM, not write protected (Default) 1 = ROM, burst ROM and write-protected SRAM
25	WPERR	**Write Protect Error Status Flag** 0 = No error (Default) 1 = Write-protect error Writing a 1 to this bit clears the write-protect error-status flag.
24	BUSERR	**Bus Transfer Error Status Flag** 0 = No error (Default) 1 = Bus-transfer error Writing a 1 to this bit clears the bus-transfer error-status flag.
23:16	///	**Reserved** Write the reset value.
15:11	WST2	**Wait State2** For SRAM: WST2 is the write access time burst access time for burst ROM. The wait state time is (WST2 + 1) × tHCLK. For Burst ROM: WST2 is the burst access time. This wait state time is (WST2) × tHCLK. WaitState2 does not apply to on-burst ROM devices. (Default = 11111)
10	RBLE	**Read Byte Lane Enable: 2** 0 = All byte lane strobes nBLE[1:0] held HIGH during reads from off-chip memory (default at System Reset). During writes, the appropriate nBLE[1:0] pins are held LOW and the nWE pin is held HIGH. 1 = All byte lane strobes nBLE[1:0] held LOW during reads from off-chip memory. During writes, nWE and the appropriate nBLE[1:0] pins are held LOW.
9:5	WST1	**Wait State1** For SRAM and ROM: WST1 is the read access time burst access time for burst ROM. For Burst ROM: WST1 is the initial access time. This wait state time is (WST2) × tHCLK. The wait state time is (WST1 + 1) × tHCLK. (Default = 11111)
4	///	**Reserved** Write the reset value.
3:0	IDCY	**Idle Cycle Memory Data Bus Turnaround Time** The turnaround time is (IDCY + 1) × tHCLK. (Default = 1111).

7.3.2.2 Configuration Register for Memory Bank 1

BCR1 is the Configuration Register for Memory Bank 1.

Table 7-12. BCR1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///		MW		BM	WP	WPERR	BUSERR	///							
RESET	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	WST2					RBLE	WST1					///	IDCY			
RESET	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW
ADDR	0xFFFF1000 + 0x04															

Table 7-13. BCR1 Register Definitions

BITS	NAME	DESCRIPTION
31:30	///	**Reserved** Write the reset value.
29:28	MW	**Memory Width** 00 = 8-bit 01 = 16-bit 10 = Reserved 11 = Reserved The MW field defaults to different values for each memory bank at reset. See Table 7-18.
27	BM	**Burst Mode** 0 = Non-burst devices (Default) 1 = Burst ROM
26	WP	**Write Protect** 0 = SRAM, not write protected (Default) 1 = ROM, burst ROM and write-protected SRAM
25	WPERR	**Write Protect Error Status Flag** 0 = No error (Default) 1 = Write-protect error Writing a 1 to this bit clears the write-protect error-status flag.
24	BUSERR	**Bus Transfer Error Status Flag** 0 = No error (Default) 1 = Bus-transfer error Writing a 1 to this bit clears the bus-transfer error-status flag.
23:16	///	**Reserved** Write the reset value.

Table 7-13. BCR1 Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
15:11	WST2	**Wait State2** For SRAM: WST2 is the write access time burst access time for burst ROM. The wait state time is (WST2 + 1) × tHCLK. For Burst ROM: WST2 is the burst access time. This wait state time is (WST2) × tHCLK. Note that tHCLK = system clock period. WaitState2 does not apply to on-burst ROM devices. (Default = 11111)
10	RBLE	**Read Byte Lane Enable:2** 0 = All byte lane strobes nBLE[1:0] are held HIGH during any system reads or writes from memory (default at system reset). 1 = All byte lane strobes nBLE[1:0] are held LOW during any system reads or writes from memory. RBLE is written 0 when interfacing to external 8-bit or non byte-partitioned memory devices. When RBLE is 0, use nOE for read operations and nBLE[1:0] for write operations (nWE is not used). When RBLE is 1, use nOE for read operations and nBLE[1:0] and nWE and nBLE[1:0] for write operations.
9:5	WST1	**Wait State1** For SRAM and ROM: WST1 is the read access time burst access time for burst ROM. For Burst ROM: WST1 is the initial access time. This wait state time is (WST2) × tHCLK. The wait state time is (WST1 + 1) × tHCLK. (*Default = 11111*)
4	///	**Reserved** Write the reset value.
3:0	IDCY	**Idle Cycle Memory Data Bus Turnaround Time** The turnaround time is (IDCY + 1) × tHCLK. (Default = 1111)

7.3.2.3 Configuration Register for Memory Bank 2

BCR2 is the Configuration Register for Memory Bank 2.

Table 7-14. BCR2 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///		MW		BM	WP	WPERR	BUSERR	///							
RESET	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	WST2					RBLE	WST1					///	IDCY			
RESET	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW
ADDR	0xFFFF1000 + 0x08															

Table 7-15. BCR2 Register Definitions

BITS	NAME	DESCRIPTION
31:30	///	**Reserved** Write the reset value.
29:28	MW	**Memory Width** 00 = 8-bit 01 = 16-bit 10 = Reserved 11 = Reserved The MW field defaults to different values for each memory bank at reset. See Table 7-18.
27	BM	**Burst Mode** 0 = Non-burst devices (Default) 1 = Burst ROM
26	WP	**Write Protect** 0 = SRAM, not write protected (Default) 1 = ROM, burst ROM and write-protected SRAM
25	WPERR	**Write Protect Error Status Flag** 0 = No error (*Default*) 1 = Write-protect error Writing a 1 to this bit clears the write-protect error-status flag.
24	BUSERR	**Bus Transfer Error Status Flag** 0 = No error (Default) 1 = Bus-transfer error Writing a 1 to this bit clears the bus-transfer error-status flag.
23:16	///	**Reserved** Write the reset value.

Table 7-15. BCR2 Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
15:11	WST2	**Wait State2** For SRAM: WST2 is the write access time burst access time for burst ROM. The wait state time is (WST2 + 1) × tHCLK. For Burst ROM: WST2 is the burst access time. This wait state time is (WST2) × tHCLK. Note that tHCLK = system clock period. WaitState2 does not apply to on-burst ROM devices. (*Default = 11111*)
10	RBLE	**Read Byte Lane Enable:2** 0 = All byte lane strobes nBLE[1:0] are held HIGH during any system reads or writes from memory (default at system reset). 1 = All byte lane strobes nBLE[1:0] are held LOW during any system reads or writes from memory. RBLE is written 0 when interfacing to external 8-bit or non byte-partitioned memory devices. When RBLE is 0, use nOE for read operations and nBLE[1:0] for write operations (nWE is not used). When RBLE is 1, use nOE for read operations and nBLE[1:0] and nWE and nBLE[1:0] for write operations.
9:5	WST1	**Wait State1** For SRAM and ROM: WST1 is the read access time burst access time for burst ROM. For Burst ROM: WST1 is the initial access time. This wait state time is (WST2) × tHCLK. The wait state time is (WST1 + 1) × tHCLK. (*Default = 11111*)
4	///	**Reserved** Write the reset value.
3:0	IDCY	**Idle Cycle Memory Data Bus Turnaround Time** The turnaround time is (IDCY + 1) × tHCLK. (*Default = 1111*)

7.3.2.4 Configuration Register for Memory Bank 3

BCR3 is the Configuration Register for Memory Bank 3.

Table 7-16. BCR3 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///		MW		BM	WP	WPERR	BUSERR	///							
RESET	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	WST2					RBLE	WST1					///	IDCY			
RESET	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW
ADDR	0xFFFF1000 + 0x0C															

Table 7-17. BCR3 Register Definitions

BITS	NAME	DESCRIPTION
31:30	///	**Reserved** Write the reset value.
29:28	MW	**Memory Width** 00 = 8-bit 01 = 16-bit 10 = Reserved 11 = Reserved The MW field defaults to different values for each memory bank at reset. See Table 7-18.
27	BM	**Burst Mode** 0 = Non-burst device. (*Default*) 1 = Burst ROM
26	WP	**Write Protect** 0 = SRAM, not write protected (*Default*) 1 = ROM, burst ROM and write-protected SRAM
25	WPERR	**Write Protect Error Status Flag** 0 = No error (*Default*) 1 = Write-protect error Writing a 1 to this bit clears the write-protect error-status flag.
24	BUSERR	**Bus Transfer Error Status Flag** 0 = No error (*Default*) 1 = Bus-transfer error Writing a 1 to this bit clears the bus-transfer error-status flag.
23:16	///	**Reserved** Write the reset value.

Table 7-17. BCR3 Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
15:11	WST2	**Wait State2** For SRAM: WST2 is the write access time burst access time for burst ROM. The wait state time is (WST2 + 1) × tHCLK. For Burst ROM: WST2 is the burst access time. This wait state time is (WST2) × tHCLK. Note that tHCLK = system clock period. WaitState2 does not apply to on-burst ROM devices. (*Default = 11111*)
10	RBLE	**Read Byte Lane Enable:2** 0 = All byte lane strobes nBLE[1:0] are held HIGH during any system reads or writes from memory (default at system reset). 1 = All byte lane strobes nBLE[1:0] are held LOW during any system reads or writes from memory. RBLE is written 0 when interfacing to external 8-bit or non byte-partitioned memory devices. When RBLE is 0, use nOE for read operations and nBLE[1:0] for write operations (nWE is not used). When RBLE is 1, use nOE for read operations and nBLE[1:0] and nWE and nBLE[1:0] for write operations.
9:5	WST1	**Wait State1** For SRAM and ROM: WST1 is the read access time burst access time for burst ROM. For Burst ROM: WST1 is the initial access time. This wait state time is (WST2) × tHCLK. The wait state time is (WST1 + 1) × tHCLK. (*Default = 11111*)
4	///	**Reserved** Write the reset value.
3:0	IDCY	**Idle Cycle Memory Data Bus Turnaround Time** The turnaround time is (IDCY + 1) × tHCLK. (*Default = 1111*)

7.3.3 SMC Default Memory Widths

At System Reset, the memory bank default external memory width is as shown in Table 7-18.

Bank 0 can default to either 8 or 16 bits, depending on whether pin PD2/INT2 is HIGH or LOW at reset.

- If PD2/INT2 is pulled HIGH on reset, Bank 0 defaults to a 16-bit memory width.
- If PD2/INT2 is pulled LOW on reset, Bank 0 defaults to an 8-bit memory width.

Pin PD2/INT2 has an internal pull-up resistor that selects Bank 0 to be 16-bit, unless an external pull-down resistor is used on that pin.

NOTE: If PD2/INT2 is LOW while in a Power Down Mode, the internal pull-up resistor will have a DC current path and deplete the battery.

Table 7-18. SMC System Reset Default Memory Width

SMC MEMORY BANK	DEFAULT MEMORY WIDTH
Bank 0	Determined by the state of PD2/INT2 at Reset.
Bank 1	16-bit
Bank 2	16-bit
Bank 3	16-bit

Chapter 8
Static Random Access Memory Controller

The SHARP BlueStreak LH75400/01/10/11 SoCs have 32KB of Static Random Access Memory (SRAM). This SRAM is organized into two 16KB blocks:

- 16KB of Tightly Coupled Memory (TCM) 0 Wait State SRAM is available to the processor as an ARM7TDMI-S bus slave.
- 16KB of internal SRAM is available as an AHB slave and accessible via processor, DMA Controller, and LCD Controller.

Each memory segment is 512MB in size, though the TCM and internal SRAMs are 16KB each in size. Any access beyond the first 16KB is mapped to the lower 16KB, but does not cause a data abort nor a prefetch abort.

Chapter 9
Reset, Clock, and Power Controller

The Reset, Clock, and Power Controller (RCPC) lets users control System Reset, clocks, power management, and external interrupt conditioning via the AMBA APB interface.

Figure 9-1 shows a block diagram of the RCPC.

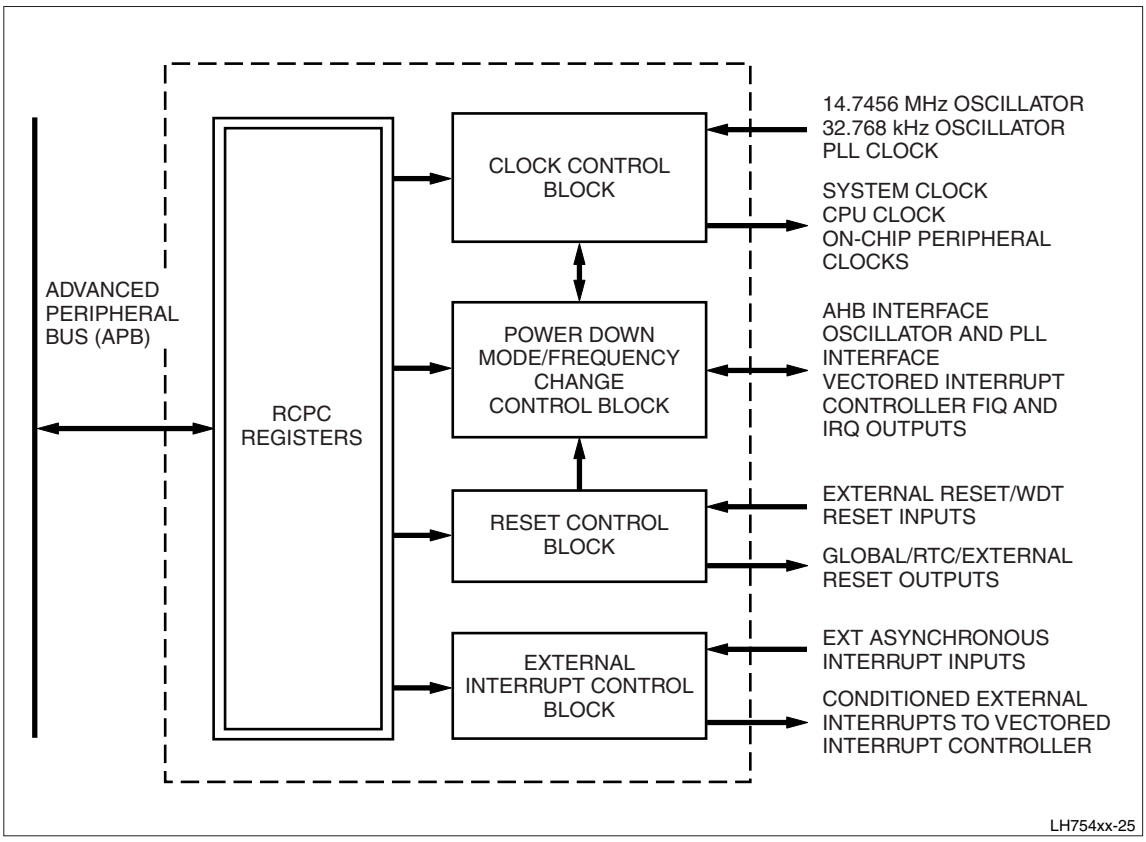


Figure 9-1. RCPC Block Diagram

9.1 RCPC Features

The RCPC provides the following features:

- Manages five Power Modes for minimizing power consumption: Active, Standby, Sleep, Stop1, and Stop2
- Generates the system clock (HCLK) from either the PLL clock or the PLL-bypassed (oscillator) clock, divided by 2, 4, 6, 8, … 30
- Generates three UART clocks from oscillator clock
- Generates the1 Hz RTC clock
- Generates the SSP and LCD clocks from HCLK, divided by 1, 2, 4, 8, 16, 32, or 64
- Provides a selectable external clock output
- Generates system and RTC resets based on an external reset, Watchdog Timer reset, or soft reset
- Configures seven HIGH/LOW-level or rising/falling edge-trigger external interrupts and converts them to HIGH-level trigger interrupt outputs required by the VIC
- Generates remap outputs used by the memory map decoder
- Provides an identification register
- Supports external or watchdog reset status.

9.2 RCPC Theory of Operation

The RCPC allows users to control System Reset, clocks, power management, and external interrupt conditioning via an AMBA APB interface. This control includes:

- Enabling and disabling various clocks
- Managing power-down sequencing
- Selecting the sources for various clocks.

The RCPC provides for an orderly start-up until the crystal oscillator stabilizes and the Phase Lock Loop (PLL) acquires lock. In addition, if users want to change the system clock frequency during normal operation, the RCPC ensures a seamless transition between the old and new frequencies. Note, however, that the same protection is not available when changing the frequency of individual peripheral clocks; as a result, the peripheral must be disabled before the peripheral is changed to a new frequency.

The RCPC manages five Power Modes:

- Active
- Standby
- Sleep
- Stop1
- Stop2.

These modes let users reduce power consumption as necessary, with each mode providing greater power savings (see Section 9.2.3 for more information). Active Mode is the normal operating mode. The other modes are entered from Active Mode via software control. The RCPC returns to Active Mode upon receiving an interrupt.

NOTE: Be sure there are no transmit or receive operations occurring when the LH75400/01/10/11 SoC device enters Standby, Sleep, Stop1, or Stop 2 Mode.

Seven external interrupt sources pass through the RCPC before being sent to the Interrupt Controller. The interrupts entering the RCPC can be individually programmed to be either level-sensitive or edge-triggered and either active-HIGH or active-LOW. All interrupts exiting the RCPC are converted to a format compatible with the VIC.

9.2.1 Reset Generation

The RCPC generates System Reset and RTC Reset outputs. The RTC block is reset by the RTC reset output, with the rest of the chip being reset by the System Reset. The nRESETOUT output pin is driven by the System Reset. The System Reset and RTC Reset are asserted by any of the following events:

- An external reset (a logic LOW signal on the external nRESETIN or nPOR input pins)
- A signal from the internal Watchdog Timer
- A Soft Reset.

A Soft Reset differentiates between the System Reset and RTC Reset.

- The System Reset is generated when 0xDEAD or 0xDEAC is written to the SoftReset Register (see Section 9.3.2.4).
- The RTC Reset is generated only when 0xDEAD is written to the SoftReset Register.

The reset latency depends on the PLL lock state. If the PLL is locked when an external reset is asserted, the System and RTC Reset outputs hold eight system clock (HCLK) cycles after the external reset is released. Since the Watchdog Timer and Soft Reset can be generated only if the system clock is running, the PLL must be locked. If the PLL is not locked when an external reset is deasserted, the RCPC waits until the PLL acquires lock and holds eight system clock cycles before releasing the system and RTC Reset outputs.

9.2.2 Clock Generation

The RCPC generates the system clock, CPU clock, and on-chip peripheral clocks from the:

- Crystal connected to the XTALIN input pin and XTALOUT output pin. This can be any value between 14 and 20 MHz; a value of 14.7456 MHz is recommended for standard UART baud rates. This User's Guide assumes the connection of a 14.7456 MHz crystal.
- 32.768 kHz crystal (connected to the XTAL32IN input pin and XTAL32OUT output pin)
- The internally generated PLL clock.

The PLL circuit multiplies the 14.7456 MHz crystal oscillator's output frequency by seven to produce the 103.22 MHz PLL clock. The system clock and CPU clock are divided from the PLL clock according to the value programmed in the SysClkPrescaler Register (see Section 9.3.2.7). The system clock connected to the DMA is not active after the reset. To activate the DMA system clock, program the AhbClkCtrl Register (see Section 9.3.2.10).

The RTC clock is generated from the 32.768 kHz crystal oscillator output. The 32.768 kHz oscillator's output is divided by 32768 to produce the 1 Hz RTC clock. The UART clocks are generated from the 14.7456 MHz crystal oscillator. To activate the RTC and UART clocks, program the APBPeriphClkCtrl0 Register (see Section 9.3.2.8).

The SSP and LCD clocks are generated from the system clock frequency. These clocks are dividable according to the values programmed in the SSPPrescaler and LCDPrescaler Registers. To activate these clocks, program the APBPeriphClkCtrl1 Register (see Section 9.3.2.9). Leave the reserved registers set to their default values.

9.2.3 RCPC Power Modes

The RCPC supports five Power Modes:

- Active mode
- Standby mode
- Sleep mode
- Stop1 mode
- Stop2 mode.

Table 9-1 shows which clock and enable states are ON and OFF for the various Power Modes.

Table 9-1. Clock and Enable States for Different Power Modes

DEVICE	ACTIVE	STANDBY	SLEEP	STOP1	STOP2
RTC oscillator (32.768 kHz)	ON	ON	ON	ON	ON
Crystal Oscillator (typically 14.7456 MHz)	ON	ON	ON	ON	OFF
PLL	ON	ON	ON	OFF	OFF
System clock	ON	ON	OFF	OFF	OFF
CPU clock	ON	OFF	OFF	OFF	OFF

NOTE: Do not have the LH75400/01/10/11 SoC device enter Standby, Sleep, Stop1, or Stop2 Mode while data is being transmitted or received.

9.2.3.1 Active Mode

Active Mode is the normal Power Mode. The SoC enters this mode after start-up and upon exiting any other Power Mode. After an External Reset, Watchdog Timer Reset, or Soft Reset is released, the System Reset is held active for an extra eight system clock cycles after the PLL is locked.

9.2.3.2 Standby Mode

Standby Mode stops the clocks to the CPU and Watchdog Timer while the rest of the SoC remains active. Standby Mode is entered when software writes 0b001 to the PWR DWN SEL field of the Ctrl Register (see Section 9.3.2.1). When an interrupt is received, the RCPC exits Standby Mode and ensures an orderly transition to Active Mode. An interrupt should be held active until the RCPC exits Standby Mode.

NOTE: Be sure there are no transmit or receive operations occurring when the LH75400/01/10/11 SoC device enters Standby Mode.

9.2.3.3 Sleep Mode

Sleep Mode stops all system clocks, keeping only the PLL and internal oscillators active. The SoC enters this mode when software writes 0b010 to the PWR DWN SEL field of the Ctrl Register (see Section 9.3.2.1). When an interrupt is received, the RCPC exits Sleep Mode and ensures an orderly transition to Active Mode. An interrupt should be held active until the RCPC exits Sleep Mode.

NOTE: Be sure there are no transmit or receive operations occurring when the LH75400/01/10/11 SoC device enters Standby Mode.

9.2.3.4 Stop1 Mode

Stop1 Mode stops all system clocks and disables the PLL, but keeps the internal oscillators active. The SoC enters this mode when software writes 0b011 to the PWR DWN SEL field of the Ctrl Register (see Section 9.3.2.1). When an interrupt is received, the RCPC exits Stop1 Mode and ensures an orderly transition to Active Mode. An interrupt should be held active until the RCPC exits Stop1 Mode.

NOTE: Be sure there are no transmit or receive operations occurring when the LH75400/01/10/11 SoC device enters Stop1 Mode.

9.2.3.5 Stop2 Mode

Stop2 Mode stops all system clocks and disables both the PLL and the internal oscillator that feeds it. However, the 32.768 kHz internal oscillator remains active. The SoC enters this mode when software writes 0b100 to the PWR DWN SEL field of the Ctrl Register (see Section 9.3.2.1). When an interrupt is received, the RCPC exits Stop2 Mode and ensures an orderly transition to Active Mode. An interrupt should be held active until the RCPC exits Stop2 Mode.

NOTE: Be sure there are no transmit or receive operations occurring when the LH75400/01/10/11 SoC device enters Stop2 Mode.

9.3 RCPC Programmer's Model

The base address for the RCPC is:

RCPC Base Address: 0xFFFE2000

The following locations are reserved and must not be used during normal operation:

- Locations at offsets 0x01C through 0x20
- Locations at offsets 0x030 through 0x3C
- Locations at offsets 0x048 through 0x7C
- Location at offset 0x088.

9.3.1 RCPC Register Summary

Table 9-2. RCPC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Ctrl	0x00	RW	0x063	RCPC Control Register
Identification	0x04	R	0x5400	ID Register
Remap	0x08	RW	0x0	Remap Control Register
SoftReset	0x0C	RW	0x0000	Soft Reset Register
ResetStatus	0x10	R	0x1	Reset Status Register
ResetStatusClr	0x14	W		Reset Status Clear Register
SysClkPrescaler	0x18	RW	0xF	System Clock Prescaler Register
///	0x1C-0x20			Reserved
APBPeriphClkCtrl0	0x24	RW	0x3FF	Peripheral Clock Control 0 Register
APBPeriphClkCtrl1	0x28	RW	0x3	Peripheral Clock Control 1 Register
AhbClkCtrl	0x2C	RW	0x1	AHB Clock Control
///	0x30-0x3C			Reserved
LCDPrescaler	0x40	RW	0x00	LCD Prescaler Register
SSPPrescaler	0x44	RW	0x00	SSP Prescaler Register
///	0x48-0x7C			Reserved
IntConfig	0x80	RW	0x0000	External Interrupt Configuration Register
IntClear	0x84	W		External Interrupt Clear Register
///	0x88			Reserved

9.3.2 RCPC Register Definitions

Except where noted, all registers are both writable and readable. Writing other than the default values to any reserved location and bit can cause the system to malfunction.

9.3.2.1 Control Register

Ctrl is the Control Register. The active bits used in this register are Read/Write.

Table 9-3. Ctrl Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						LOCK	///				PWR DWN SEL			///	
RESET	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	1
RW	R	R	R	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x00															

Table 9-4. Ctrl Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:10	///	**Reserved** Write the reset value.
9	LOCK	**Lock** 0 = All RCPC registers accessible through the APB, other than this bit and the IntClear Register (see Section 9.3.2.14), are write-protected. 1 = All RCPC APB-accessible registers are write-enabled (*default*).
8:5	///	**Reserved** Write the reset value.
4:2	PWR DWN SEL	**Power Down Mode Select** 000 = Active Mode 001 = Standby Mode 010 = Sleep Mode 011 = Stop1 Mode 100 = Stop2 Mode Other values = undefined Be sure there are no transmit or receive operations occurring when the LH75400/01/10/11 SoC device enters Standby, Sleep, Stop1, or Stop2 Mode. These bits always read 000 because the RCPC clears them automatically at wakeup.
1:0	///	**Reserved** Write the reset value.

9.3.2.2 Identification Register

ID is the Identification Register. This Read Only contains the last four digits of the part number encoded one part number digit per hex digit. The part number is encoded one part number digit per hex digit.

Table 9-5. ID Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	PART_NUMBER															
RESET	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE2000 + 0x04															

Table 9-6. ID Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Writing to these bits has no effect. Reading returns 0.
15:0	PART_NUMBER	**Part Number Digits** Specifies the last four digits of the part number (for example, '5400', '5401', '5410', or '5411'. Values for bits [4] and [0] vary from those shown in Table 9-6, depending on the device.

9.3.2.3 Remap Control Register

This Remap Register provides a remapping feature for the system memory map. See Chapter 6 for more information about the Remap feature. The active bits used in this register are Read/Write.

Table 9-7. Remap Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///														REMAP	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW
ADDR	0xFFFE2000 + 0x08															

Table 9-8. Remap Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:2	///	**Reserved** Writing to these bits has no effect.
1:0	REMAP	**Remap** Remaps external RAM, TCM and local SRAM, and system peripherals.

9.3.2.4 Soft Reset Register

SoftReset is the Soft Reset Register. This register provides a way for software to activate and deactivate the System Reset. SoftReset must reset the entire chip. The active bits used in this register are Read/Write.

Table 9-9. SoftReset Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SRVAL															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x0C															

Table 9-10. SoftReset Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Writing to these bits has no effect. Reading returns 0.
15:0	SRVAL	**Activate Reset** Writing 0b1101111010101101 (0xDEAD) to these bits activates a System Reset. Do not write any other value to these bits. Writing any other value to these bits can cause unpredictable operation.

9.3.2.5 Reset Status Register

ResetStatus is the Reset Status Register. This Read Only register provides the reset status of the device. It contains the external reset status and the WDT timeout reset status. At external reset, the EXT bit is set and the WDTO bit is cleared. At WDT timeout, only the WDTO bit is set. The EXT and WDTO bits remain set until they are cleared by the Reset Status Clear operation.

Table 9-11. ResetStatus Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	—														WDTO	EXT
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE2000 + 0x10															

Table 9-12. ResetStatus Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:2	///	**Reserved** Writing to these bits has no effect.
1	WDTO	**WDT Timeout** 0 = No WDT timeout has occurred since the flag was last cleared. 1 = WDT timeout has occurred.
0	EXT	**External Reset** 0 = No external reset has occurred since the flag was last cleared. 1 = External reset has occurred.

9.3.2.6 Reset Status Clear Register

ResetStatusClr is the Reset Status Clear Register. This Write Only register clears the Reset Status flags. When writing to this register, each HIGH data bit causes the corresponding bit in the Reset Status Register to be cleared. LOW data bits have no effect on their corresponding bit in the Reset Status register. Writing to undefined bits has no effect on the RCPC.

Table 9-13. ResetStatusClr Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///														WDTO CLR	EXT CLR
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFE2000 + 0x14															

NOTE: The reset value of this register's bits is indeterminate.

Table 9-14. ResetStatusClr Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:2	///	**Reserved** Reads undefined. Write zero only.
1	WDTO CLR	**Clear WDT Timeout** 1 = Clears WDTO in the ResetStatus Register. Reads of this bit are unpredictable.
0	EXT CLR	**Clear External Reset** 1 = Clears EXT in the ResetStatus Register. Reads of this bit are unpredictable.

9.3.2.7 HCLK Prescaler Register

SysClkPrescaler is the HCLK Prescaler Register. This register is a 4-bit value that holds the prescale count for the HCLK prescaler. The active bits used in this register are Read/Write.

Table 9-15. SysClk Prescaler Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///												HCLK			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x18															

Table 9-16. SysClk Prescaler Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:4	///	**Reserved** Writing to these bits has no effect.
3:0	HCLK	**HCLK Prescaler Prescale Count** Shows the prescale count for the HCLK prescaler. See Table 9-17 for valid values for SysClkPrescaler and the corresponding internal clock frequencies. All other SysClkPrescaler values are invalid and ignored.

Table 9-17. SysClkPrescaler Register Values

SYSCLKPRESCALER	DIVIDER VALUE	*f*(HCLK)
0001	2	*f*(source clock)/2
0010	4	*f*(source clock)/4
0011	6	*f*(source clock)/6
0100	8	*f*(source clock)/8
:	:	:
1111	30	*f*(source clock)/30

NOTE: System clock = (XTALIN frequency × 7)/(2 × SysClkPrescaler).

9.3.2.8 Peripheral Clock Control Register 0

APBPeriphClkCtrl0 is the Peripheral Clock Control Register 0. The active bits used in this register are Read/Write.

This register controls the real-time, U2, U1, and U0 peripheral clocks. When writing to this register, setting a data bit to one stops the clock of the corresponding peripheral.

Table 9-18. APBPeriphClkCtrl0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						RTC	—						U2	U1	U0
RESET	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x24															

Table 9-19. APBPeriphClkCtrl0 Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:10	///	**Reserved** Write the reset value.
9	RTC	**RTC Clock** 0 = Real-time clock is running. 1 = Stops the real-time clock.
8:3	///	**Reserved** Write the reset value.
2	U2	**U2 Peripheral Clock** 0 = U2 peripheral clock is running. 1 = Stops the U2 peripheral clock.
1	U1	**U1 Peripheral Clock** 0 = U1 peripheral clock is running. 1 = Stops the U1 peripheral clock.
0	U0	**U0 Peripheral Clock** 0 = U0 peripheral clock is running. 1 = Stops the U0 peripheral clock.

9.3.2.9 Peripheral Clock Control Register 1

APBPeriphClkCtrl1 is the Peripheral Clock Control Register 1. This register controls the LCD and SSP peripheral clocks. When writing to this register, setting a data bit to one stops the clock of the corresponding peripheral. The active bits used in this register are Read/Write.

Table 9-20. APBPeriphClkCtrl1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						///								SSP	LCD
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x28															

Table 9-21. APBPeriphClkCtrl1 Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:2	///	**Reserved** Write the reset value.
1	SSP	**SSP Peripheral Clock** 0 = Does not stop the SSP peripheral clock. 1 = Stops the SSP peripheral clock.
0	LCD	**LCD Peripheral Clock** 0 = Does not stop the LCD peripheral clock. 1 = Stops the LCD peripheral clock.

9.3.2.10 AHB Clock Control Register

AhbClkCtrl is the AHB Clock Control Register. When writing to this register, setting a data bit to one stops the AHB DMA clock. Bit [1] of this register should never be cleared. The bit used in this register is Read/Write.

Table 9-22. AhbClkCtrl Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///							///							///	DMA
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
RW	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x2C															

Table 9-23. AhbClkCtrl Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:1	///	**Reserved** Write the reset value.
0	DMA	**AHB DMA Clock** 0 = AHB DMA clock is running. 1 = Stops the AHB DMA clock.

9.3.2.11 LCD Clock Prescaler Register

LCDPrescaler is the LCD Clock Prescaler Register. The active bits used in this register are Read/Write.

This register divides down the LCD clock frequencies using the appropriate formula:

- If LCDPrescaler > 0: f(LCD) = f(HCLK) ÷ (2 * LCDPrescaler)
- If LCDPrescaler = 0: f(LCD) = f(HCLK)

Table 9-26 shows the valid values for LCDPrescaler and the resulting internal clock frequency. All other LCDPrescaler values are invalid.

Table 9-24. LCDPrescaler Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								LCDPRESCALER							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x40															

Table 9-25. LCDPrescaler Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:8	///	**Reserved** Writing to these bits has no effect. Reading returns 0.
7:0	LCDPRESCALER	**LCD Clock Frequencies** Divides down the LCD clock frequencies (see Table 9-26).

Table 9-26. LCDPrescaler Register Values

LCDPRESCALER	DIVIDER VALUE	f(LCD)
00000000 (default)	1	f(HCLK)
00000001	2	f(HCLK)/2
00000010	4	f(HCLK)/4
00000100	8	f(HCLK)/8
00001000	16	f(HCLK)/16
00010000	32	f(HCLK)/32
00100000	64	f(HCLK)/64
01000000	128	f(HCLK)/128
10000000	256	f(HCLK)/256

9.3.2.12 SSP Clock Prescaler Register

SSPPrescaler is the SSP Clock Prescaler Register. The active bits used in this register are Read/Write.

This register divides down the SSP clock frequencies using the appropriate formula:

- If SSPPrescaler > 0: f(SSP) = f(HCLK) ÷ (2 * SSPPrescaler)
- If SSPPrescaler = 0: f(SSP) = f(HCLK)

Table 9-29 shows the valid values for SSPPrescaler and the resulting internal clock frequency. All other SSPPrescaler values are invalid.

Table 9-27. SSPPrescaler Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	—								SSPPRESCALER							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x44															

Table 9-28. SSPPrescaler Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:8	///	**Reserved** Writing to these bits has no effect. Reading returns 0.
7:0	SSPPRESCALER	**SSP Clock Frequencies** Divides down the SSP clock frequencies (see Table 9-29).

Table 9-29. SSPPrescaler Register Values

SSPPrescaler	DIVIDER VALUE	f(SSP)
00000000 (default)	1	f(HCLK)
00000001	2	f(HCLK)/2
00000010	4	f(HCLK)/4
00000100	8	f(HCLK)/8
00001000	16	f(HCLK)/16
00010000	32	f(HCLK)/32
00100000	64	f(HCLK)/64
01000000	128	f(HCLK)/128
10000000	256	f(HCLK)/256

9.3.2.13 External Interrupt Configuration Register

IntConfig is the External Interrupt Configuration Register. The active bits used in this register are Read/Write.

This register configures the individual external interrupts to be either edge-sensitive or level-sensitive and either active HIGH or active LOW. When reset, all bits are cleared and configure the external interrupts to be active LOW-level sensitive. The corresponding edge-trigger interrupt should be cleared before enabling the edge-trigger interrupt to clear a false interrupt.

Table 9-30. IntConfig Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///		INT6		INT5		INT4		INT3		INT2		INT1		INT0	
RESET	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE2000 + 0x80															

Table 9-31. IntConfig Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:14	///	**Reserved** Write the reset value.
13:12	INT6	**Configures External Interrupt INT6** 00 = Configures INT6 to be a LOW-level trigger. 01 = Configures INT6 to be a HIGH-level trigger. 10 = Configures INT6 to be a falling-edge trigger. 11 = Configures INT6 to be a rising-edge trigger.
11:10	INT5	**Configures External Interrupt INT5** 00 = Configures INT5 to be a LOW-level trigger. 01 = Configures INT5 to be a HIGH-level trigger. 10 = Configures INT5 to be a falling-edge trigger. 11 = Configures INT5 to be a rising-edge trigger.
9:8	INT4	**Configures External Interrupt INT4** 00 = Configures INT4 to be a LOW-level trigger. 01 = Configures INT4 to be a HIGH-level trigger. 10 = Configures INT4 to be a falling-edge trigger. 11 = Configures INT4 to be a rising-edge trigger.
7:6	INT3	**Configures External Interrupt INT3** 00 = Configures INT3 to be a LOW-level trigger. 01 = Configures INT3 to be a HIGH-level trigger. 10 = Configures INT3 to be a falling-edge trigger. 11 = Configures INT3 to be a rising-edge trigger.

Table 9-31. IntConfig Register Definitions (Cont'd)

BITS	FIELD NAME	DESCRIPTION
5:4	INT2	**Configures External Interrupt INT2** 00 = Configures INT2 to be a LOW-level trigger. 01 = Configures INT2 to be a HIGH-level trigger. 10 = Configures INT2 to be a falling-edge trigger. 11 = Configures INT2 to be a rising-edge trigger.
3:2	INT1	**Configures External Interrupt INT1** 00 = Configures INT1 to be a LOW-level trigger. 01 = Configures INT1 to be a HIGH-level trigger. 10 = Configures INT1 to be a falling-edge trigger. 11 = Configures INT1 to be a rising-edge trigger.
1:0	INT0	**Configures External Interrupt INT0** 00 = Configures INT0 to be a LOW-level trigger. 01 = Configures INT0 to be a HIGH-level trigger. 10 = Configures INT0 to be a falling-edge trigger. 11 = Configures INT0 to be a rising-edge trigger.

9.3.2.14 External Interrupt Clear Register

IntClear is the External Interrupt Clear Register. The active bits used in this register are Write Only.

This register individually clears active external interrupts. This register can only clear edge-triggered interrupts. Writing a one to a bit clears the corresponding active edge-triggered interrupt. Writing to undefined bits has no effect on the RCPC.

Table 9-32. IntClear Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									INT6	INT5	INT4	INT3	INT2	INT1	INT0
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W
ADDR	0xFFFE2000 + 0x84															

NOTE: The reset value of this register's bits is indeterminate.

Table 9-33. IntClear Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:8	///	**Reserved** Writing to these bits has no effect.
7	///	**Reserved** Write only 0.
6	INT6	**Clear INT6 Interrupt** 0 = Does not clear the active edge-triggered interrupt INT6. 1 = Clears the active edge-triggered interrupt INT6.
5	INT5	**Clear INT5 Interrupt** 0 = Does not clear the active edge-triggered interrupt INT5. 1 = Clears the active edge-triggered interrupt INT5.
4	INT4	**Clear INT4 Interrupt** 0 = Does not clear the active edge-triggered interrupt INT4. 1 = Clears the active edge-triggered interrupt INT4.
3	INT3	**Clear INT3 Interrupt** 0 = Does not clear the active edge-triggered interrupt INT3. 1 = Clears the active edge-triggered interrupt INT3.
2	INT2	**Clear INT2 Interrupt** 1 = Clears the active edge-triggered interrupt INT2. 0 = Does not clear the active edge-triggered interrupt INT2.
1	INT1	**Clear INT1 Interrupt** 0 = Does not clear the active edge-triggered interrupt INT1. 1 = Clears the active edge-triggered interrupt INT1.
0	INT0	**Clear INT0 Interrupt** 0 = Does not clear the active edge-triggered interrupt INT0. 1 = Clears the active edge-triggered interrupt INT0.

Chapter 10
Vectored Interrupt Controller

The Vectored Interrupt Controller (VIC) provides the principle user interface to the interrupt system.

10.1 Theory of Operation

All internal and external interrupts are routed to the VIC, where interrupt priority is determined by hardware. The VIC is also where the appropriate signal to the processor (IRQ or FIQ) is generated. The CPU services the interrupt as either a vectored interrupt or a default-vectored interrupt.

The VIC uses a set of 32-bit registers to configure, control, and determine the status of the VIC. These registers are described in Section 10.2.2.

On reset, the VIC is configured to pass all interrupts through to the CPU IRQ input as default-vectored IRQ interrupts. In the reset configuration, the VIC Status Registers can be used in a conventional way to service interrupts using the CPU IRQ Exception Vector at address 0x18. Users must configure the VIC to use the vectored interrupt feature. The VIC's software-generated interrupt feature is available immediately out of reset.

10.1.1 Interrupts

The VIC accepts inputs from 32 interrupt source lines:

- Seven of the interrupt source lines are external
- Twenty-three of the interrupt source lines are internal
- Two interrupt source lines that can be used as software interrupts.

All 32 interrupt source lines can be enabled, disabled, and cleared individually, and individual status may be determined. On reset, all interrupts are disabled.

The VIC also accepts software-generated interrupts. A software-generated interrupt can be invoked on any interrupt line to force a specific Interrupt Service Routine (ISR) to execute in the absence of a hardware interrupt. This is useful for lines designated as 'spare;' however, a software interrupt can be generated on any of the 32 interrupt lines. A software-generated interrupt adheres to the same enabling control as hardware-generated interrupts.

The VIC provides 32 interrupts:

- 16 vectored interrupts
- 16 or more default-vectored interrupts.

A vectored interrupt results in a low-latency invocation of the service routine for that particular interrupt. A default-vectored interrupt requires the CPU to perform additional processing to determine which interrupt source caused the interrupt.

Any of the 32 lines can be assigned to any of the 16 interrupt vectors. Any line not explicitly assigned to an interrupt vector is processed as a default-vectored interrupt. At reset, all 32 lines are set to be default-vectored interrupts.

Each interrupt line can be explicitly identified as either an IRQ interrupt type (default) or an FIQ interrupt type. Vectored-interrupt servicing is only available for IRQ interrupts. Although more than one interrupt source can be designated as FIQ, only one source normally is designated to take advantage of the low latency of FIQ exception handling for a specific need.

10.1.2 VIC Interrupt Listing

Table 10-1 lists the 32 interrupt source lines for the VIC. For a detailed description of each interrupt, see the description of the peripheral that generates that interrupt. The source for position 31 is version-specific for each device.

Table 10-1. Interrupt Assignments

POSITION	DESCRIPTION	SOURCE
0	WDT	Watchdog Timer
1	Not Used	Can be used as a software interrupt
2	ARM7 DBGCOMMRX	Sourced by the ARM7TDMI-S Core
3	ARM7 DBGCOMMTX	Sourced by the ARM7TDMI-S Core
4	Timer0 Combined	Timer0
5	Timer1 Combined	Timer1
6	Timer2 Combined	Timer2
7	External Interrupt 0	Sourced by the GPIO Block
8	External Interrupt 1	Sourced by the GPIO Block
9	External Interrupt 2	Sourced by the GPIO Block
10	External Interrupt 3	Sourced by the GPIO Block
11	External Interrupt 4	Sourced by the GPIO Block
12	External Interrupt 5	Sourced by the GPIO Block
13	External Interrupt 6	Sourced by the GPIO Block
14	Not Used	Can be used as a software interrupt
15	RTC_ALARM	Real Time Clock
16	ADC TSCIRQ (combined)	Analog-to-Digital Converter
17	ADC BrownOutINTR	Brown Out Detector
18	ADC PenIRQ	Analog-to-Digital Converter
19	LCD	LCD Controller
20	SSPTXINTR	Synchronous Serial Port
21	SSPRXINTR	Synchronous Serial Port
22	SSPRORINTR	Synchronous Serial Port
23	SSPRXTOINTR	Synchronous Serial Port
24	SSPINTR	Synchronous Serial Port
25	UART1 UARTRXINTR	UART1

Table 10-1. Interrupt Assignments

POSITION	DESCRIPTION	SOURCE
26	UART1 UARTTXINTR	UART1
27	UART1 UARTINTR	UART1
28	UART0 UARTINTR	UART0
29	UART2 Interrupt	UART2
30	DMA	DMA
31	CAN	CAN (LH75401 and LH75400)
		Reserved (LH75411 and LH75410)

10.1.3 Vectored Interrupts

Each interrupt source line must be identified as either an IRQ type or an FIQ type using the IntSelect Register (see Section 10.2.2.4). If the interrupt is designated as an FIQ type, the FIQ interrupt service is non-vectored. Once the VIC causes the FIQ interrupt to be asserted to the core, the FIQ interrupt handler is reached directly by loading the instruction at 0x1C independently of the VIC.

If any default-vectored interrupts will be enabled, set the DefVectAddr Register to the entry address of the ISR which is to handle all default-vectored interrupts (see Section 10.2.2.10).

For each interrupt line that will be enabled as a vectored interrupt:

- Set the corresponding VectAddr(x) Register (where 'x' is 0-15) to the entry address of the ISR which is to handle that specific interrupt.
- Set the IntSource field of the corresponding VectCtrl(x) Register to the interrupt source for that specific vector. Then enable that interrupt source as a vectored interrupt using the E field in that register.

Enable each interrupt line to be enabled, whether vectored or default-vectored, using the IntEnable Register (see Section 10.2.2.5).

10.1.4 External Interrupts

All external interrupts are conditioned by the RCPC module before being presented to the VIC. External interrupt conditioning can be configured to one of four triggers using the RCPC IntConfig Register (see Chapter 9, Section 9.3.2.13):

- Low-level trigger
- High-level trigger
- Falling-edge trigger
- Rising-edge trigger.

On reset, all external interrupt triggers are LOW-level triggers. Exercise care to ensure that all external interrupt input signals are HIGH at reset. External edge-triggered interrupts must be cleared using the RCPC IntClear Register (see Section 9.3.2.14). If the external interrupt is configured as a level-trigger interrupt, the external interrupt must be cleared, reset, or disabled at its source (external to the SoC).

10.1.5 Clearing Interrupts

While the procedure for clearing an interrupt varies from source to source, general clearing actions must be performed:

1. The interrupt must be cleared at its source, regardless of whether the interrupt source is external, internal, or software generated.
   – If an interrupt source is external and configured as edge triggered, the interrupt must be cleared in the RCPC IntClear Register (see Section 9.3.2.14).
   – If an interrupt source is external and configured as level triggered, the interrupt must be cleared, reset, or disabled at its source external to the SoC.
   – If the interrupt source is a software command, the interrupt must be cleared using the SoftIntClear Register (described under Section 10.2.2.8).
   – If the interrupt source is internal, the interrupt must be cleared in a way appropriate to the internal source. Usually this involves setting a bit or clearing a bit in a 'clearing' register specific to the particular internal source. For example, the DMA Controller has a Clr Register that must be written with a value specific to the DMA Controller. Other devices within the SoC have similar device-specific ways of clearing an interrupt generated by that device. See the appropriate chapter in this Technical Data Sheet for information about clearing each device.
2. The interrupt must be cleared within the VIC by writing any value to the VectAddr Register (described in Section 10.2.2.9). Writing a value of '0' is recommended. This action signals the hardware vector address and priority logic that it can assert a new interrupt and its associated address.

10.1.6 Priority

The VIC can assert an FIQ interrupt and an IRQ interrupt simultaneously. When this occurs, the CPU gives the FIQ priority over the IRQ interrupt. Priority arbitration for simultaneously invoked IRQ interrupts is performed in the VIC hardware.

The priority of IRQ interrupt sources is:

- All vectored interrupts have priority over default-vectored interrupts
- Vectored interrupt 0 has the highest priority
- Among the vectored interrupts, the higher numbered vectored interrupt has lower priority
- Within the VIC, all default-vectored interrupts have the same priority, which is the lowest priority.

10.1.7 Sequencing

The sequence of interrupt processing in the SoCs are:

1. An interrupt is asserted.
2. One of the appropriate actions occurs:
   – If the interrupt is an external interrupt, the interrupt is conditioned by the RCPC to an active HIGH signal into the VIC.
   – If the interrupt is an internal interrupt, the signal into the VIC is an active HIGH.
   – If the interrupt is a software interrupt, a bit has been set in the SoftInt Register that causes the equivalent of an HIGH level interrupt on the associated line.
3. The unmasked results appear in the RawIntr Register (described in Section 10.2.2.3), and are submitted to an interrupt-enable masking operation. If the signal has been unmasked (that is, enabled as an interrupt which is indicated by a corresponding bit having been initialized in the IntEnable Register), the VIC continues processing the interrupt.
4. If the signal has been enabled as an interrupt, it is routed according to whether it has been identified as an FIQ or IRQ in the IntSelect Register (described in Section 10.2.2.4).
   – If it has not been enabled, processing stops. Note that the interrupt is not cleared if it has not been enabled.
   – If the interrupt is subsequently enabled, the uncleared interrupt is asserted immediately, producing a spurious interrupt if global interrupts are enabled.
5. If the interrupt has been identified as an FIQ, the active HIGH signal is asserted on the CPU FIQ input line without further processing.
6. If the interrupt has been identified as an IRQ, the active HIGH signal is routed for further processing by interrupt vector and priority logic.
7. The interrupt vector logic establishes whether the interrupt has been associated with a vectored interrupt. If the signal is identified as a source calling for handling by a vectored interrupt 0-15, the signal is routed to the correct vector logic for processing.
8. If the signal is not identified as a source calling for handling by a particular vectored interrupt, the signal is treated as a default-vectored interrupt and is routed to the default-vectored interrupt logic for processing.
9. For the case where the signal is a vectored interrupt, the value in the associated VectAddrX Register (described under Section 10.2.2.11) loads into the VectAddr Register (described under Section 10.2.2.9). The VectAddrX Register is has been initialized with the entry address of the vectored interrupt handler. The signal itself is routed to interrupt priority logic within the VIC. The active HIGH output signal from the priority processing is asserted on the CPU IRQ line.
10. If the signal is a default-vectored interrupt, the processing is similar to that for a vectored interrupt; however, the same vector is used for multiple interrupt source lines. The value in the DefVectAddr Register (described under Section 10.2.2.10) loads into the VectAddr Register. The DefVectAddr Register has been initialized with the entry address of the default-vectored interrupt handler. The signal itself is routed to interrupt priority logic within the VIC. The active HIGH output signal from the priority processing is asserted on the CPU IRQ line.

11. After an FIQ or IRQ line into the CPU is asserted, the CPU takes over processing of the interrupt by switching internally to the applicable Processor Mode (either FIQ or IRQ).
12. If the interrupt is an FIQ interrupt, CPU processing should follow normal ARM conventions for processing FIQ interrupts. For more information, consult the ARM literature at www.arm.com.
13. If the interrupt is an IRQ interrupt, low-latency processing is invoked by the programming the instruction:

    0x18LDR PC, [PC,#-0xFF0]

    at location 0x18 (the CPU IRQ Exception Vector) in the CPU memory map.
14. If global IRQ interrupts are enabled in the CPSR of the CPU, the instruction at address 0x18 is the first instruction executed when any IRQ exception is invoked. Executing this instruction immediately loads the VectAddr Register value into the program counter as the address of the next instruction to be executed. Since the VIC loads VectAddr with the entry address for the specific interrupt handler for the associated signal, ISR processing begins with the next instruction, without needing to determine the interrupt source. Similarly, if the interrupt is default-vectored, the VIC loads the Default Vector Address into the VectAddr Register. For default-vectored interrupts, the ISR determines the source of the interrupt and handles it appropriately.
15. FIQ and IRQ interrupts are globally disabled by the CPU when an FIQ is asserted. IRQ interrupts are globally disabled by the CPU when an IRQ is asserted. The timing and circumstances of re-enabling global interrupts and implementing interrupt nesting are implementation specific, and special precautions to save and restore context must be taken. Global interrupts should be re-enabled upon invoking the conventional means of returning from a single (non-nested) interrupt.
16. Before returning from an interrupt, the interrupt must be cleared at the source and a WRITE operation to the VIC VectAddr Register must be executed — in this order. This sequence clears the interrupt, and notifies the interrupt and priority logic that the next interrupt can be processed by the hardware. The next interrupt might be pending, or it might be asynchronously asserted at a later time; however, the sequence of interrupt processing by the VIC is the same for each.

A pending interrupt is serviced immediately upon the return from handling the current interrupt. This occurs because the VIC hardware immediately reasserts the IRQ line to the CPU when the VIC VectAddr Register is written. However, when interrupt nesting is not permitted, the interrupt is not acted upon until the global IRQ interrupts are re-enabled.

10.1.8 External Level-Sensitive Interrupts

When external interrupts are configured as level-sensitive, the ISR must ensure that there is a sufficient time between the time when the source of the external interrupt is cleared and the time when the interrupt at the VIC is cleared. Otherwise, the source of an external interrupt can pull the line HIGH or LOW (depending on whether the external interrupt input pin is configured as active LOW or active HIGH) before the interrupt is cleared at the VIC. Because the VIC samples the line after the clear, it generates another interrupt to the ARM core if the line is recognized to be still active. To avoid this situation, clear the source of the interrupt as early as practical in the ISR. Doing so ensures a maximum delay between clearing the external interrupt and clearing the interrupt at the VIC.

An interrupt line shared by multiple open-collector devices in a wired-OR configuration with a pull-up resistor can be extremely susceptible to causing multiple interrupts if there is insufficient delay between the time when the source of the external interrupt is cleared and the time when the interrupt at the VIC is cleared. This situation is due to the relatively slow risetime of the interrupt signal when being pulled to its inactive state by the pull-up resistor. The larger the resistor and load capacitance on the interrupt line, the slower the rise time and the greater the delay required.

10.1.9 Software Guidelines

User software that makes changes to the VIC IRQStatus, FIQStatus, or RawIntr register should not immediately be followed by a read to these registers. Instead, at least one Idle cycle must separate the write and read operations. The Idle cycle(s) is necessary because the VIC is a zero-wait-state peripheral that requires two clocks for the write operation to update internal registers. The pipelining of the AHB, along with the VIC not inserting wait states, means that a read access immediately following a write returns the previous register values.

10.2 VIC Programmer's Model

The base address for the VIC is:

VIC Base Address: 0xFFFFF000

The following locations are reserved and must not be used during normal operation:

- Locations at offsets 0x020
- Locations at offsets 0x300
- Locations at offsets 0x304
- Locations at offsets 0x308
- Locations at offsets 0x30C
- Locations at offsets 0x310

10.2.1 VIC Register Summary

Table 10-2. VIC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
IRQStatus	0x000	R	0x00000000	IRQ Status Register
FIQStatus	0x004	R	0x00000000	FIQ Status Register
RawIntr	0x008	R	—	Raw Interrupt Status Register
IntSelect	0x00C	RW	0x00000000	Interrupt Select Register
IntEnable	0x010	RW	0x00000000	Interrupt Enable Register
IntEnClear	0x014	W		Interrupt Enable Clear Register
SoftInt	0x018	RW	0x00000000	Software Interrupt Register
SoftInt Clear	0x01C	W		Software Interrupt Clear Register
///	0x020			Reserved
VectAddr	0x030	RW	0x00000000	Vector Address Register
DefVectAddr	0x034	RW	0x00000000	Default Vector Address Register
VectAddr 0	0x100	RW	0x00000000	Vector Address 0 Register
VectAddr 1	0x104	RW	0x00000000	Vector Address 1 Register
VectAddr 2	0x108	RW	0x00000000	Vector Address 2 Register
VectAddr 3	0x10C	RW	0x00000000	Vector Address 3 Register
VectAddr 4	0x110	RW	0x00000000	Vector Address 4 Register
VectAddr 5	0x114	RW	0x00000000	Vector Address 5 Register
VectAdd 6	0x118	RW	0x00000000	Vector Address 6 Register
VectAddr 7	0x11C	RW	0x00000000	Vector Address 7 Register
VectAddr 8	0x120	RW	0x00000000	Vector Address 8 Register
VectAddr 9	0x124	RW	0x00000000	Vector Address 9 Register
VectAddr 10	0x128	RW	0x00000000	Vector Address 10 Register
VectAddr 11	0x12C	RW	0x00000000	Vector Address 11 Register
VectAddr 12	0x130	RW	0x00000000	Vector Address 12 Register
VectAddr 13	0x134	RW	0x00000000	Vector Address 13 Register
VectAddr 14	0x138	RW	0x00000000	Vector Address 14 Register
VectAddr 15	0x13C	RW	0x00000000	Vector Address 15 Register
VectCtrl 0	0x200	RW	0x00	Vector Control 0 Register
VectCtrl 1	0x204	RW	0x00	Vector Control 1 Register
VectCtrl 2	0x208	RW	0x00	Vector Control 2 Register
VectCtrl 3	0x20C	RW	0x00	Vector Control 3 Register
VectCtrl 4	0x210	RW	0x00	Vector Control 4 Register
VectCtrl 5	0x214	RW	0x00	Vector Control 5 Register
VectCtrl 6	0x218	RW	0x00	Vector Control 6 Register
VectCtrl 7	0x21C	RW	0x00	Vector Control 7 Register
VectCtrl 8	0x220	RW	0x00	Vector Control 8 Register
VectCtrl 9	0x224	RW	0x00	Vector Control 9 Register
VectCtrl 10	0x228	RW	0x00	Vector Control 10 Register
VectCtrl 11	0x22C	RW	0x00	Vector Control 11 Register

Table 10-2. VIC Register Summary (Cont'd)

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
VectCtrl 12	0x230	RW	0x00	Vector Control 12 Register
VectCtrl 13	0x234	RW	0x00	Vector Control 13 Register
VectCtrl 14	0x238	RW	0x00	Vector Control 14 Register
VectCtrl 15	0x23C	RW	0x00	Vector Control 15 Register
///	0x300			Reserved
///	0x304			Reserved
///	0x308			Reserved
///	0x30C	R	0x0	Reserved
///	0x310			Reserved

10.2.2 VIC Register Definitions

10.2.2.1 IRQ Status Register

IRQStatus is the IRQ Status Register. This Read Only register provides the status of interrupts [31:0] after IRQ masking.

Table 10-3. IRQStatus Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	IRQStatus															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	IRQStatus															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFFF000 + 0x000															

Table 10-4. IRQStatus Register Definitions

BIT	NAME	DESCRIPTION
31:0	IRQStatus	**Interrupt Status After Masking** Shows the status of the interrupts after masking by the IntEnable and IntSelect Registers. 0 = Interrupt is not active. 1 = Interrupt is active and generates an IRQ exception to the ARM7TDMI-S core. Bits [31:0] correspond to the interrupt order in the Interrupt Assignments Table (see Table 10-1).

10.2.2.2 FIQ Status Register

FIQStatus is the FIQ Status Register. This Read Only register provides the status of the interrupts after FIQ masking.

Table 10-5. FIQStatus Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	FIQStatus															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	FIQStatus															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFFF000 + 0x004															

Table 10-6. FIQStatus Register Definitions

BIT	NAME	DESCRIPTION
31:0	FIQStatus	**Interrupt Status After Masking** Shows the status of the interrupts after masking by the IntEnable and IntSelect Registers. 0 = Interrupt is not active. 1 = Interrupt is active and generates an FIQ exception to the ARM7TDMI-S core. Bits [31:0] correspond to the interrupt order in the Interrupt Assignments Table (See Table 10-1).

10.2.2.3 Raw Interrupt Status Register

RawIntr is the Raw Interrupt Status Register. This Read Only register provides the status of the source interrupts (and software interrupts) to the Interrupt Controller.

Table 10-7. RawIntr Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	RawInterrupt															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RawInterrupt															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFFF000 + 0x008															

Table 10-8. RawIntr Register Definitions

BIT	NAME	DESCRIPTION
31:0	RawInterrupt	**Interrupt Status After Masking** Shows the status of the interrupts before masking by the Enable Registers. 0 = Appropriate interrupt request is not active before masking. 1 = Appropriate interrupt request is active before masking. Bits [31:0] correspond to the interrupt order in the Interrupt Assignments Table (see Table 10-1).

10.2.2.4 Interrupt Select Register

IntSelect is the Interrupt Select Register. This register selects whether the corresponding interrupt source generates an FIQ or an IRQ interrupt.

Table 10-9. IntSelect Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	IntSelect															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	IntSelect															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFFF000 + 0x00C															

Table 10-10. IntSelect Register Definitions

BIT	NAME	DESCRIPTION
31:0	IntSelect	**Interrupt Type** Selects the type of interrupt for the interrupt request: 0 = IRQ interrupt 1 = FIQ interrupt Bits [31:0] correspond to the interrupt order in the Interrupt Assignments Table (see Table 10-1).

10.2.2.5 Interrupt Enable Register

IntEnable is the Interrupt Enable Register. This register enables the interrupt request lines, by masking the interrupt sources for the IRQ interrupt.

Table 10-11. IntEnable Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	IntEnable															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	IntEnable															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFFF000 + 0x010															

Table 10-12. IntEnable Register Definitions

BIT	NAME	DESCRIPTION
31:0	IntEnable	**Interrupt Enable** Corresponds to the interrupt order in the Interrupt Assignments table (see Table 10-1). When any bit position is read: 0 = Interrupt is disabled. 1 = Interrupt is enabled, allowing interrupt request to ARM7TDMI-S core. When any bit position is written to: 0 = Has no effect. 1 = Enable the corresponding interrupt. On System Reset, all interrupts are disabled.

10.2.2.6 Interrupt Enable Clear Register

IntEnClear is the Interrupt Enable Clear Register. This register clears bits in the IntEnable Register (see Section 10.2.2.5).

Table 10-13. IntEnClear Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	IntEnable Clear															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	IntEnable Clear															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFFF000 + 0x014															

Table 10-14. IntEnClear Register Definitions

BIT	NAME	DESCRIPTION
31:0	IntEnable Clear	**Clear IntEnable Bit** Clears bits in the IntEnable Register. 0 = Has no effect. 1 = Clears the corresponding bit in the IntEnable Register. Bits [31:0] correspond to the interrupt order in the Interrupt Assignments Table (see Table 10-1).

10.2.2.7 Software Interrupt Register

SoftInt is the Software Interrupt Register. This register generates software interrupts.

Table 10-15. SoftInt Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	SoftInt															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SoftInt															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFFF000 + 0x018															

Table 10-16. SoftInt Register Definitions

BIT	NAME	DESCRIPTION
31:0	SoftInt	**Generate Software Interrupt** Setting a bit generates a software interrupt for the specific source interrupt before interrupt masking. 1 = Sets the corresponding bit in the SoftInt Register. 0 = Has no effect. Bits [31:0] correspond to the interrupt order in the Interrupt Assignments Table (see Table 10-1).

10.2.2.8 Software Interrupt Clear Register

SoftIntClear is the Software Interrupt Clear Register. This Write Only register clears bits in the SoftInt Register (see Section 10.2.2.7).

Table 10-17. SoftIntClear Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	SoftInt Clear															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SoftInt Clear															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFFF000 + 0x01C															

Table 10-18. SoftIntClear

BIT	NAME	DESCRIPTION
31:0	SoftInt Clear	**Clear SoftInt Register Bits** Clears bits in the SoftInt Register. 0 = Has no effect. 1 = Clears the corresponding bit in the SoftInt Register. Bits [31:0] correspond to the interrupt order in the Interrupts Assignments Table (see Table 10-1).

10.2.2.9 Vector Address Register

VectAddr is the Vector Address Register. This register contains the ISR address of the currently active interrupt. Reading from this register provides the address of the ISR, and indicates to the priority hardware that the interrupt is being serviced. Writing to this register indicates to the priority hardware that the interrupt has been serviced.

The ISR reads the VectAddr Register:

- When an IRQ interrupt is generated at the end of the ISR.
- When the VectAddr Register is written to.
- To update the priority hardware.

Reading or writing to the register at other times can cause incorrect operation.

Table 10-19. VectAddr Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	VectorAddr															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	VectorAddr															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFFF000 + 0x030															

Table 10-20. VectAddr Register Definitions

BIT	NAME	DESCRIPTION
31:0	VectorAddr	**ISR Address** Contains the address of the currently active ISR. Any writes to this register clear the interrupt.

10.2.2.10 Default Vector Address Register

DefVectAddr is the Default Vector Address Register. This register contains the default ISR address. This address is used for non-vectored IRQs.

Table 10-21. DefVectAddr Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	Default VectorAddr															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	Default VectorAddr															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFFF000 + 0x034															

Table 10-22. DefVectAddr Register Definitions

BIT	NAME	DESCRIPTION
31:0	Default VectorAddr	**Default ISR Handler Address** Contains the address of the default ISR handler.

10.2.2.11 Vector Address Registers

There are 16 Vector Address Registers, designated VectAddr0 through VectAddr15. These registers contain the ISR vector addresses for the vectored IRQ interrupts.

Table 10-23. VectAddr Registers

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	VICVectorAddr															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	VICVectorAddr															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	VectAddr0: 0xFFFFF000 + 0x100 VectAddr1: 0xFFFFF000 + 0x104 VectAddr2: 0xFFFFF000 + 0x108 VectAddr3: 0xFFFFF000 + 0x10C VectAddr4: 0xFFFFF000 + 0x110 VectAddr5: 0xFFFFF000 + 0x114 VectAddr6: 0xFFFFF000 + 0x118 VectAddr7: 0xFFFFF000 + 0x11C VectAddr8: 0xFFFFF000 + 0x120 VectAddr9: 0xFFFFF000 + 0x124 VectAddr10: 0xFFFFF000 + 0x128 VectAddr11: 0xFFFFF000 + 0x12C VectAddr12: 0xFFFFF000 + 0x130 VectAddr13: 0xFFFFF000 + 0x134 VectAddr14: 0xFFFFF000 + 0x138 VectAddr15: 0xFFFFF000 + 0x13C															

Table 10-24. VectAddr Register Definitions

BIT	NAME	DESCRIPTION
31:0	VectorAddr (0 - 15)	**ISR Vector Addresses** Contains ISR vector addresses.

10.2.2.12 Vector Control Registers

There are 16 Vector Control Registers, designated VectCtrl0 through VectCtrl15. These registers select the interrupt source for the given vectored interrupt. The active bits used in these registers are Read/Wrote.

Vectored interrupts are only generated if the interrupt is enabled. The specific interrupt is enabled in the IntEnable Register (see Section 10.2.2.5), and the interrupt is set to generate an IRQ interrupt in the IntSelect Register (Section 10.2.2.4). This prevents multiple interrupts from being generated by a single request if the controller is programmed incorrectly.

Table 10-25. VectCtrl Registers

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										E	IntSource				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW
ADDR	VectCtrl0: 0xFFFFF000 + 0x200 VectCtrl1: 0xFFFFF000 + 0x204 VectCtrl2: 0xFFFFF000 + 0x208 VectCtrl3: 0xFFFFF000 + 0x20C VectCtrl4: 0xFFFFF000 + 0x210 VectCtrl5: 0xFFFFF000 + 0x214 VectCtrl6: 0xFFFFF000 + 0x218 VectCtrl7: 0xFFFFF000 + 0x21C VectCtrl8: 0xFFFFF000 + 0x220 VectCtrl9: 0xFFFFF000 + 0x224 VectCtrl10: 0xFFFFF000 + 0x228 VectCtrl11: 0xFFFFF000 + 0x22C VectCtrl12: 0xFFFFF000 + 0x230 VectCtrl13: 0xFFFFF000 + 0x234 VectCtrl14: 0xFFFFF000 + 0x238 VectCtrl15: 0xFFFFF000 + 0x23C															

Table 10-26. VectCtrl Register Definitions

BIT	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5	E	**Vector Interrupt Enable** Enables the vector interrupt. This bit is cleared on System Reset.
4:0	IntSource	**Interrupt Source Selection** Selects the interrupt source from any of the 32 interrupt sources.

Chapter 11
I/O Configuration

The I/O Configuration (IOCON) peripheral is an AMBA slave block that connects to the APB. The IOCON provides registers for programming the pin muxing on the device, and for controlling the pull-up and pull-down resistors on certain pins of the chip.

11.1 IOCON Theory of Operation

All IOCON Control Registers can be accessed through the APB. All pins associated with GPIO, except bit [14] of the EBI_MUX Register, default to have the GPIO controlling the pin after a System Reset. (Bit [14] of the EBI_MUX Register can default to the data pins, depending on the Boot Mode.) The alternative functions can then be accessed by programming the IOCON registers. The LCD_MUX does not allow individual control of the pins; however, it permits different LCD modes to be programmed. Depending on the mode, different pins either will be used by the LCD or will be GPIO.

The IOCON has five registers for controlling the pull-up/pull-down resistors on certain pins. These registers default to pull-up, pull-down, or no-pull depending on the pin's function.

11.2 IOCON Programmer's Model

The base address for the IOCON is:

IOCON Base Address: 0xFFFE5000

11.2.1 IOCON Register Summary

Table 11-1. IOCON Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
EBI_MUX	0x00	RW	See Note	EBI Interface Muxing Register
PD_MUX	0x04	RW	0x0000	Pins PD6/INT6 to PD0/INT0 Muxing Register
PE_MUX	0x08	RW	0x0000	Pins PE7/SSPRM to PE0/UARTRX2 Muxing Register
TIMER_MUX	0x0C	RW	0x0000	Timer Muxing Register
LCD_MUX	0x10	RW	0x0000	LCD Mode Muxing Register
PA_RES_MUX	0x14	RW	0xAAAA	Pins PA7/D15 to PA0/D8 Resistor Muxing Register
PB_RES_MUX	0x18	RW	0x0555	Pins PB5/nWAIT to PB0/nCS1 Resistor Muxing Register
PC_RES_MUX	0x1C	RW	0x0000	Pins PC7/A23 to PC0/A16 Resistor Muxing Register
PD_RES_MUX	0x20	RW	0x095A	Pins PD6/INT6 to PD0/INT0 Resistor Muxing Register
PE_RES_MUX	0x24	RW	0x4455	Pins PE7/SSPRM to PE0/UARTRX2 Resistor Muxing Register
ADC_MUX	0x28	RW	0x0000	Pins AN3/PJ7 to AN0/PJ0 Muxing Register

NOTE: The reset value is based on whether the system is booted in 16-bit or 8-bit Mode. In 16-bit Mode, the reset value is 0x4000. During 8-bit Mode, the reset value is 0x0000.

11.2.2 IOCON Register Definitions

Multi-bit pin configuration fields may have a bit setting called 'Reset Condition'. Setting the bit field to this configures the pin to be the same as it would be set after a System Reset.

11.2.2.1 EBI Interface Muxing Register

EBI_MUX is the EBI Interface Muxing Register. This register allows the secondary function of the EBI interface pins to be configured as GPIO. The active bits used in this register are Read/Write.

NOTE: The register's reset value is based on whether the system is booted in 16-bit or 8-bit Mode. In 16-bit Mode, the reset value is 0x4000. In 8-bit Mode, the reset value is 0x0000.

Table 11-2. EBI_MUX Register (16-bit Mode)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	DATA	nWAIT	nBLE1	nBLE0	nCE3	nCE2	nCE1	A23	A22	A21	A20	A19	A18	A17	A16
RESET	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x00															

Table 11-3. EBI_MUX Register (8-bit Mode)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	DATA	nWAIT	nBLE1	nBLE0	nCE3	nCE2	nCE1	A23	A22	A21	A20	A19	A18	A17	A16
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x00															

Table 11-4. EBI_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:15	///	**Reserved** Write the reset value.
14	DATA	**PA7/D15 to PA0/D8 Source** 0 = PA7 to PA0 1 = D15 to D8
13	nWAIT	**PB5/nWAIT Source** 0 = PB5 1 = nWAIT
12	nBLE1	**PB4/nBLE1 Source** 0 = PB4 1 = nBLE1
11	nBLE0	**PB3/nBLE0 Source** 0 = PB3 1 = nBLE0
10	nCE3	**PB2/nCE3 Source** 0 = PB2 1 = nCE3
9	nCE2	**PB1/nCE2 Source** 0 = PB1 1 = nCE2
8	nCE1	**PB0/nCE1 Source** 0 = PB0 1 = nCE1
7	A23	**PC7/A23 Source** 0 = PC7 1 = A23
6	A22	**PC6/A22 Source** 0 = PC6 1 = A22
5	A21	**PC5/A21 Source** 0 = PC5 1 = A21
4	A20	**PC4/A20 Source** 0 = PC4 1 = A20
3	A19	**PC3/A19 Source** 0 = PC3 1 = A19
2	A18	**PC2/A18 Source** 0 = PC2 1 = A18
1	A17	**PC1/A17 Source** 0 = PC1 1 = A17
0	A16	**PC0/A16 Source** 0 = PC0 1 = A16

11.2.2.2 Pins PD6/INT6 to PD0/INT0 Muxing Register

PD_MUX is the Pins PD6/INT6 to PD0/INT0 Muxing Register. This register allows the secondary function of the interrupt interface pins to be configured as GPIO. The active bits used in this register are Read/Write.

Table 11-5. PD_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///					INT6		INT5		INT4		INT3		INT2	INT1	INT0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x04															

Table 11-6. PD_MUX Register Definitions

BIT	NAME	DESCRIPTION
31:11	///	**Reserved** Write the reset value.
10:9	INT6	**Pin PD6/INT6/DREQ Source** 00 = PD6 01 = INT6 10 = DREQ 11 = PD6
8:7	INT5	**Pin PD5/INT5/DACK Source** 00 = PD5 01 = INT5 10 = DACK 11 = PD5
6:5	INT4	**Pin PD4/INT4/UARTRX1 Source** 00 = PD4 01 = INT4 10 = UARTRX1 11 = PD4
4:3	INT3	**Pin PD3/INT3/UARTTX1 Source** 00 = PD3 01 = INT3 10 = UARTTX1 11 = PD3
2	INT2	**PD2/INT2 Source** 0 = PD2 1 = INT2
1	INT1	**PD1/INT1 Source** 0 = PD1 1 = INT1
0	INT0	**PD0/INT0 Source** 0 = PD0 1 = INT0

11.2.2.3 Pins PE7/SSPRM to PD0/INT0 Muxing Register

PE_MUX is the Pins PE7/SSPRM to PD0/INT0 Muxing Register. This register allows the secondary function of the pins to be configured as GPIO. The active bits used in this register are Read/Write.

The functions of bits [5:2] vary among all four SoCs:

- LH75401 and LH75400: bits [5:4] are assigned CANTX functions and bits [3:2] are assigned CANRX functions.
- LH75410 and LH75411: bits [5:2] are reserved; always write 0.

Table 11-7. PE_MUX Register (LH75401 and LH75400)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						SSPFRM	SSPCLK	SSPRX	SSPTX	CANTX		CANRX		UARTTX2	UARTRX2
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x08															

Table 11-8. PE_MUX Register (LH75410 and LH75411)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						SSPFRM	SSPCLK	SSPRX	SSPTX	///				UARTTX2	UARTRX2
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x08															

Table 11-9. PE_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:10	///	**Reserved** Write the reset value.
9	SSPFRM	**PE7/SSPFRM Source** 0 = PE7 1 = SSPFRM
8	SSPCLK	**PE6/SSPCLK Source** 0 = PE6 1 = SSPCLK
7	SSPRX	**PE5/SSPRX Source** 0 = PE5 1 = SSPRX
6	SSPTX	**PE4/SSPTX Source** 0 = PE4 1 = SSPTX
5:4	CANTX	**Pin PE3/CANTX/UARTTX0 Source** (LH75401 and LH75400 SoC only) 00 = PE3 01 = CANTX1 10 = UARTTX0 11 = PE3
3:2	CANRX	**Pin PE2/CANRX/UARTRX0 Source** (LH75401 and LH75400 SoC only) 00 = PE2 01 = CANRX1 10 = UARTRX0 11 = PE2
5:2	///	**Reserved** Write the reset value.
1	UARTTX2	**PE1/UARTTX2 Source** 0 = PE1 1 = UARTTX2
0	UARTRX2	**PE0/UARTRX2 Source** 0 = PE0 1 = UARTRX2

11.2.2.4 Timer Muxing Register

TIMER_MUX is the Timer Muxing Register. This register allows the secondary function of the TIMER interface pins to be configured as GPIO. The active bits used in this register are Read/Write.

Table 11-10. TIMER_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	CTCAP2B		CTCAP2A		CTCAP1B		CTCAP1A		CTCAP0E	CTCAP0D	CTCAP0C	CTCAP0B		CTCAP0A		CTCLK
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x0C															

Table 11-11. TIMER_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:14	CTCAP2B	**Pin PF6 /CTCAP2B/CTCMP2B Source** 00 = PF6 01 = CTCAP2B 10 = CTCMP2B 11 = PF6
13:12	CTCAP2A	**Pin PF5/CTCAP2A/CTCMP2A Source** 00 = PF5 01 = CTCAP2A 10 = CTCMP2A 11 = PF5
11:10	CTCAP1B	**Pin PF4/CTCAP1B/CTCMP1B Source** 00 = PF4 01 = CTCAP1B 10 = CTCMP1B 11 = PF4
9:8	CTCAP1A	**Pin PF3/CTCAP1A/CTCMP1A Source** 00 = PF3 01 = CTCAP1A 10 = CTCMP1A 11 = PF3
7	CTCAP0E	**PF2/CTCAP0E Source** 0 = PF2 1 = CTCAP0E

Table 11-11. TIMER_MUX Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
6	CTCAP0D	**PF1/CTCAP0D Source** 0 = PF1 1 = CTCAP0D
5	CTCAP0C	**PF0/CTCAP0C Source** 0 = PF0 1 = CTCAP0C
4:3	CTCAP0B	**Pin PG7/CTCAP1B/CTCMP0B Source** 00 = PG7 01 = CTCAP0B 10 = CTCMP0B 11 = PG7
2:1	CTCAP0A	**Pin PG6/CTCAP0A/CTCMP0A Source** 00 = PG6 01 = CTCAP0A 10 = CTCMP0A 11 = PG6
0	CTCLK	**PG5/CTCLK Source** 0 = PG5 1 = CTCLK

11.2.2.5 LCD Mode Muxing Register

LCD_MUX is the LCD Mode Muxing Register. This register allows the LCD display to be configured to different modes supported by the SoC, freeing GPIO. The active bits used in this register are Read/Write.

Table 11-12. LCD_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///													MODE		
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW
ADDR	0xFFFE5000 + 0x10															

Table 11-13. LCD_MUX Register Definitions (LH75401 and LH75411 SoC Devices)

BIT	NAME	DESCRIPTION
31:3	///	**Reserved** Write the reset value.
2:0	MODE	**LCD Mode** 000 = No LCD 001 = 4-bit Mono STN Mode 010 = 4-bit Mono STN Dual Mode 011 = 8-bit Mono/Color STN Mode 100 = TFT Mode 101 through 111 = No LCD

Table 11-14. LCD_MUX Register Definitions (LH75400 and LH75410 SoC Devices)

BIT	NAME	DESCRIPTION
31:3	///	**Reserved** Write the reset value.
2:0	MODE	**LCD Mode** 000 = No LCD 001 = 4-bit Mono STN Mode 010 = 4-bit Mono STN Dual Mode 011 = 8-bit Mono STN Mode 100 = Reserved 101 through 111 = No LCD

11.2.2.6 Pins PA7/D15 to PA0/D8 Resistor Muxing Register

PA_RES_MUX is the Pins PA7/D15 to PA0/D8 Resistor Muxing Register. This register allows the pull-up/pull-down to be configured as needed. The active bits used in this register are Read/Write.

Table 11-15. PA_RES_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	PA7		PA6		PA5		PA4		PA3		PA2		PA1		PA0	
RESET	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x14															

Table 11-16. PA_RES_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:14	PA7	**Pin PA7/D15 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up
13:12	PA6	**Pin PA6/D14 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up
11:10	PA5	**Pin PA5/D13 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up
9:8	PA4	**Pin PA4/D12 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up
7:6	PA3	**Pin PA3/D11 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up

Table 11-16. PA_RES_MUX Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
5:4	PA2	**Pin PA2/D10 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up
3:2	PA1	**Pin PA1/D9 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up
1:0	PA0	**Pin PA0/D8 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = Pull-up

11.2.2.7 Pins PB5/nWAIT to PB0/nCS1 Resistor Muxing Register

PB_RES_MUX is the Pins PB5/nWAIT to PB0/nCS1 Resistor Muxing Register. This register allows the pull-up/pull-down to be configured as needed. The active bits used in this register are Read/Write.

Table 11-17. PB_RES_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///				PB5		PB4		PB3		PB2		PB1		PB0	
RESET	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x18															

Table 11-18. PB_RES_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:12	///	**Reserved** Write the reset value.
11:10	PB5	**Pin PB5/nWAIT Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
9:8	PB4	**Pin PB4/nBLE1 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
7:6	PB3	**Pin PB3/nBLE0 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
5:4	PB2	**Pin PB2/nCS3 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
3:2	PB1	**Pin PB1/nCS2 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
1:0	PB0	**Pin PB0/nCS1 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up

11.2.2.8 Pins PC7/A23 to PC0/A16 Resistor Muxing Register

PC_RES_MUX is the Pins PC7/A23 to PC0/A16 Resistor Muxing Register. This register allows the pull-up/pull-down to be configured as needed. The active bits used in this register are Read/Write.

Table 11-19. PC_RES_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	PC7		PC6		PC5		PC4		PC3		PC2		PC1		PC0	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x1C															

Table 11-20. PC_RES_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:14	PC7	**Pin PC7/A23 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
13:12	PC6	**Pin PC6/A22 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
11:10	PC5	**Pin PC5/A21 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
9:8	PC4	**Pin PC4/A20 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
7:6	PC3	**Pin PC3/A19 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down

Table 11-20. PC_RES_MUX Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
5:4	PC2	**Pin PC2/A18 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
3:2	PC1	**Pin PC1/A17 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
1:0	PC0	**Pin PC0/A16 Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down

11.2.2.9 Pins PD6/INT6 to PD0/INT0 Resistor Muxing Register

PD_RES_MUX is the Pins PD6/INT6 to PD0/INT0 Resistor Muxing Register. This register allows the pull-up/pull-down to be configured as needed. The active bits used in this register are Read/Write.

Table 11-21. PD_RES_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///		PD6		PD5		PD4		PD3		PD2		PD1		PD0	
RESET	0	0	0	0	1	0	0	1	0	1	0	1	1	0	1	0
RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x20															

Table 11-22. PD_RES_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:14	///	**Reserved** Write the reset value.
13:12	PD6	**Pin PD6/INT6/DREQ Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
11:10	PD5	**Pin PD5/INT5/DACK Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = No pull-up or pull-down
9:8	PD4	**Pin PD4/INT4/UARTRX1 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
7:6	PD3	**Pin PD3/INT3/UARTTX1 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
5:4	PD2	**Pin PD2/INT2 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up

Table 11-22. PD_RES_MUX Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
3:2	PD1	**Pin PD1/INT1 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = No pull-up or pull-down
1:0	PD0	**Pin PD0/INT0 Resistor Source** 00 = Pull-down 01 = Pull-up 10 = No pull-up or pull-down (default) 11 = No pull-up or pull-down

11.2.2.10 Pins PE7/SSPRM to PE0/UARTRX2 Resistor Muxing Register

PE_RES_MUX is the Pins PE7/SSPRM to PE0/UARTRX2 Resistor Muxing Register. This register allows the pull-up/pull-down to be configured as needed. The active bits used in this register are Read/Write.

The functions associated with bits [7:6] and [5:4] vary among the four SoCs:

- LH75400 and LH75401:
  - Bits [7:6] correspond to the Pin PE3/CANTX/UARTTX0 Resistor Source.
  - Bits [5:4] correspond to the Pin PE2/CANRX/UARTRX0 Resistor Source.
- LH75410 and LH75411:
  - Bits [7:6] correspond to the Pin PE3/UARTTX0 Resistor Source.
  - Bits [5:4] correspond to the Pin PE2/UARTRX0 Resistor Source.

Table 11-23. PERES_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	PE7		PE6		PE5		PE4		PE3		PE2		PE1		PE0	
RESET	0	1	9	0	0	1	9	0	0	1	0	1	0	1	0	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x24															

Table 11-24. PE_RES_MUX Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:14	PE7	**Pin PE7/SSPFRM Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
13:12	PE6	**Pin PE6/SSPCLK Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
11:10	PE5	**Pin PE5/SSPRX Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up

Table 11-24. PE_RES_MUX Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
9:8	PE4	**Pin PE4/SSPTX Resistor Source** 00 = Pull-down (default) 01 = Pull-up 10 = No pull-up or pull-down 11 = Pull-down
7:6	PE3	**Pin PE3/CANTX/UARTTX0 Resistor Source (LH75400 and LH75401 SoC)** **Pin PE3/UARTTX0 Resistor Source (LH75410 and LH75411 SoC)** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
5:4	PE2	**Pin PE2/CANRX/UARTRX0 Resistor Source (LH75400 and LH75401 SoC)** **Pin PE2/UARTRX0 Resistor Source (LH75410 and LH75411 SoC)** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
3:2	PE1	**Pin PE1/UARTTX2 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up
1:0	PE0	**Pin PE0/UARTRX2 Resistor Source** 00 = Pull-down 01 = Pull-up (default) 10 = No pull-up or pull-down 11 = Pull-up

11.2.2.11 Pins AN7/PJ7 to AN0/PJ0

ADC_MUX is the Pins AN7/PJ7 to AN0/PJ0 Register. This register allows the secondary function of the ADC interface pins to be configured as General Purpose Inputs. The active bits used in this register are Read/Write.

Table 11-25. ADC_MUX Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE5000 + 0x28															

Table 11-26. ADC_MUX Register

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	PJ7	**Pin AN3/PJ7 Source** 0 = AN3 (LR/Y-) 1 = PJ7
6	PJ6	**Pin AN4/PJ6 Source** 0 = AN4 9Wiper) 1 = PJ6
5	PJ5	**Pin AN9/PJ5 Source** 0 = AN9 1 = PJ5
4	PJ4	**Pin AN2/PJ4 Source** 0 = AN2 (LL/Y+) 1 = PJ3
3	PJ3	**Pin AN8/PJ3 Source** 0 = AN8 1 = PJ3
2	PJ2	**Pin AN1/PJ2 Source** 0 = AN1 (UR/X-) 1 = PJ2
1	PJ1	**Pin AN6/PJ1 Source** 0 = AN6 1 = PJ1
0	PJ0	**Pin AN0/PJ0 Source** 0 = AN0 (UL/X+) 1 = PJ0

Chapter 12
Direct Memory Access Controller

The Direct Memory Access (DMA) Controller provides DMA support for the DMA-capable peripherals listed in Table 12-1. It is not used for the display system. The LCD Controller has its own DMA port for connecting directly to the memory system for retrieving display data. The DMA is controlled by the system clock.

One central DMA Controller services all the peripheral DMA requirements. The controller has an APB slave port for programming of its registers by the ARM and an AHB port for data transfers.

Table 12-1. DMA Controller Stream Assignments and Request Priority

DMA REQUEST SOURCE	DMA STREAM
UART1RX (highest priority)	Stream0
UART1TX	Stream1
UART0RX/External Request (DREQ)	Stream2
UART0TX (lowest priority)	Stream3

12.1 DMA Controller Features

The DMA Controller has the following features:

- Four data streams
- Three modes of transfer:
  - Memory to Memory (selectable on Stream3 only)
  - Peripheral to Memory (all streams)
  - Memory to Peripheral (all streams).
- Built-in data stream arbiter
- Seven registers for each stream:
  - DMA enable
  - Transfer Size (Byte, Half-word, Word)
  - Burst Size (1, 4, 8, or 16)
  - Address Increment Enable
  - Transfer Direction
  - Maximum Count
  - Terminal Count.

- Ability of each stream to indicate a transfer error via an interrupt
- 16-word First-In, First Out (FIFO) array, with pack and unpack logic to handle all input/output combinations of byte, half-word, and word transfers
- APB slave port for programming of its registers by the ARM core
- AHB port for data transfers.

12.2 DMA Theory Of Operation

One central DMA Controller services all DMA requirements for DMA-capable devices (see Table 12-1 for a list of supported devices). The DMA is controlled by the system clock.

The DMA Controller transfers data between AHB peripherals and memory or between memory and memory. The DMA Controller supports four data streams (Stream0, Stream1, Stream2, and Stream3) that can be used to service:

- Four peripheral data streams (peripheral-to-memory or memory-to-peripheral)
- Three peripheral data streams and one memory-to-memory data stream.

The four data streams use a fixed-priority arbitration scheme and share one common 16-word-deep FIFO for buffering burst data. Each of the four data streams has its own independent set of DMA Registers and address/transfer count counters. In addition:

- Stream2 provides a set of external signals for initiating and controlling DMA transfers between external peripherals and memory. These signals (DREQ and DACK) are brought out to external pins that are multiplexed with other functions.
- Stream3 can carry out memory-to-memory DMA transfers under software control.

The DMA Registers are programmed through an APB Slave interface that has a 16-bit data interface. Therefore, 32-bit registers are accessed as two 16-bit registers: a low 16-bit half and a high 16-bit half. See Table 12-1 for more information.

The four data streams share a common 16-word-deep FIFO for buffering DMA data. A stream can be programmed to transfer a number of data units (from 1 to 65535). A data unit represents a group of bits equal in width to the data width of the source peripheral or memory. The source and destination data widths can be programmed independently to be byte, half-word, or word. Data is transferred in bursts, with the burst length programmable to 1, 4, 8, or 16 peripheral data units. The stream has source and destination address registers that can be independently programmed to remain fixed or increment after each data access.

The following steps summarize the DMA process:

1. The external request signal (DREQ) starts a peripheral DMA transfer.
2. The DMA Controller requests use of the AHB.
3. When the AHB arbiter grants the AHB to the DMA Controller, the DMA Controller fills its FIFO with the number of data units specified by the burst length (1, 4, 8, or 16).
4. The DMA Controller continues to request the AHB following the completion of the burst transfer. However, it may lose ownership of the AHB if a higher priority bus master is also requesting the AHB.
5. When the AHB arbiter re-grants the AHB to the DMA Controller, the FIFO is emptied by writing the FIFO contents to the destination with data being of a width set by the destination data width (byte, half-word or word). The filling and emptying of the FIFO for a burst transfer is always completed for the current stream being serviced before another stream DMA request is serviced.
6. As DMA requests are received, the DMA Controller arbitrates between them, assigning a requesting source to be serviced based on the priority indicated in Table 12-1. A data packet transfers from the source to the DMA FIFO, then transfers from the FIFO to the destination.

Exceptions to the DMA process are:

- When the DMA is configured to perform a memory-to-memory transfer followed by a peripheral-to-memory transfer, the transfer starts immediately, without the DMA waiting for the external request signal in step 1. The software workaround to this is:
  - Set up a memory-to-memory access.
  - Let the memory-to-memory complete.
  - Set up the peripheral-to-memory transfer, but without the enable bit set.
  - Perform a second write operation, with the enable bit set.
- When the DMA is configured to perform an external memory-to-peripheral write operation. This operation may cause the processor to hang when:
  - All DMA registers, except CTRL, have been initialized properly.
  - A write to the Status Clear bits (bits [7:0]) in the CLR Register for Data Stream 2 has been performed.
  - The DMA Controller Enable bit (bit [2]) in the CTRL Register has been set.

  To avoid this situation, use this procedure to start an external Memory-to-Peripheral DMA transfer:

  a. Set up the Stream 2 DMA Control Register for the transfer except have the enable bit clear.
  b. Read the value of the current Stream 2 DMA Control Register into a temp variable.
  c. Write the temp variable back to the current Stream 2 Control Register.
  d. Set bit 0 of the temp variable.
  e. Write the temp variable to the Stream 2 Control Register.
- When Stream3 is used for memory-to-memory transfers, the transfer starts when software sets an enable bit in the Control Register for that stream. The transfer is conducted in bursts, with the bursts executing back-to-back until the required number of data units are transferred. The DMA Controller retains ownership of the AHB between successive bursts, unless the AHB Arbiter de-grants the DMA Controller for a higher priority bus master.

12.2.1 Interrupt, Error, and Status Registers

The SoCs provide Interrupt, Error, and Status Registers for controlling the generation of an interrupt, error-handling control, and active-stream monitoring. Each stream has its own interrupt flag, which is set after the last transfer completes. Each of the four interrupt flags can be masked and cleared independently.

Each stream also has its own error flag. An error flag is set when the data stream transfer is aborted due to an ERROR response from an AHB slave. Each of the four error flags can be separately masked and cleared. The masked interrupt and error flags are all combined into a single interrupt output.

12.2.2 DMA Controller Timing Diagrams

Figure 12-1 and Figure 12-2 show examples of DMA timing diagrams.

- Figure 12-1 shows the timing for a peripheral-to-memory data transfer, where SoSize = DeSize and SoBurst = 4.
- Figure 12-2 shows the timing for a memory-to-peripheral data transfer, where SoSize = DeSize and SoBurst = 4.

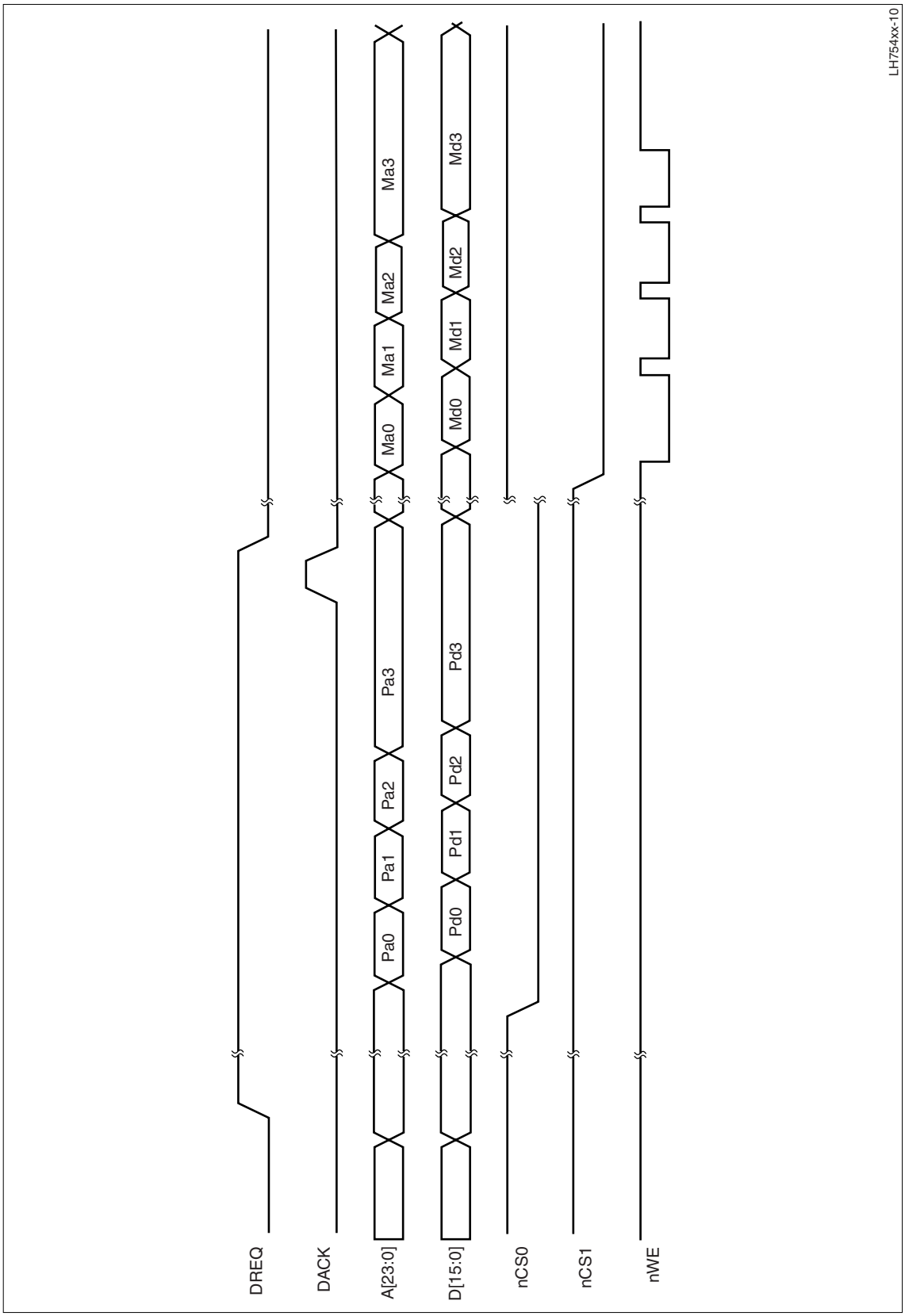


Figure 12-1. Peripheral-to-Memory Data-Transfer Timing

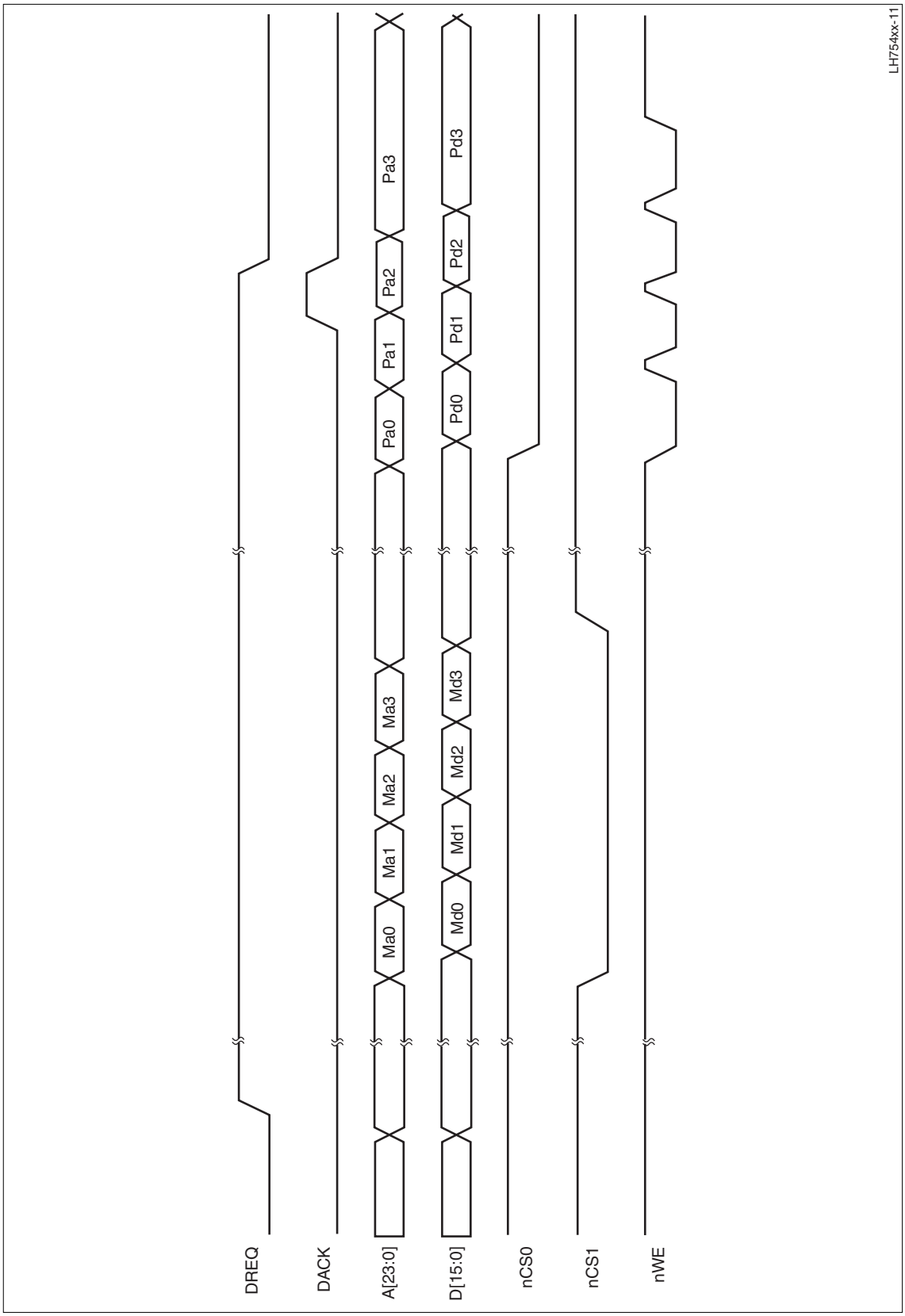


Figure 12-2. Memory-to-Peripheral Data-Transfer Timing

12.3 DMA Programmer's Model

The base address for the DMA Controller is:

DMA Base Address: 0xFFFE1000

The DMA Registers are accessed via the APB; the register data path is 16 bits wide. The register offsets for the DMA Controller are defined in Table 12-2.

Table 12-2. DMA Register Summary

NAME	ADRESS OFFSET	TYPE	DESCRIPTION
Stream0	0x000	RW	Data Stream0 Register Base
Stream1	0x040	RW	Data Stream1 Register Base
Stream2	0x080	RW	Data Stream2 Register Base
Stream3	0x0C0	RW	Data Stream3 Register Base
Mask	0x0F0	RW	DMA Interrupt Mask Register
Clr	0x0F4	W	DMA Interrupt Clear
Status	0x0F8	R	DMA Status Register

12.3.1 DMA Controller Register Summary

The registers from 0x000 to 0x0EC are the Stream Configuration Registers, of which there is a set of registers for each data stream. The registers for each stream are shown in Table 12-3.

Table 12-3. Data Stream Register Summary

NAME	ADDRESS OFFSET	TYPE	DESCRIPTION
SourceLo	0x000	RW	Source base address, lower 16 bits
SourceHi	0x004	RW	Source base address, higher 16 bits
DestLo	0x008	RW	Destination base address, lower 16 bits
DestHi	0x00C	RW	Destination base address, higher 16 bits
Max	0x010	RW	Maximum Count Register
Ctrl	0x014	RW	Control Register
SoCurrHi	0x018	R	Current source address, higher 16 bits
SoCurrLo	0x01C	R	Current source address, lower 16 bits
DeCurrHi	0x020	R	Current destination address, lower 16 bits
DeCurrLo	0x024	R	Current destination address, higher 16 bits
TCnt	0x028	R	Terminal counter
///	0x2C - 0x3C	—	Reserved

12.3.2 DMA Controller Register Definitions

12.3.2.1 Source Base Registers

These two 16-bit registers contain the 32-bit source base address for the next DMA transfer. When the DMA Controller is enabled, the content of the Source Base Address Register is loaded in the Current Source Address Register.

12.3.2.2 Destination Base Register

These two 16-bit registers contain the 32-bit destination base address for the next DMA transfer. When the DMA Controller is enabled, the content of the Destination Base Address Register is loaded in the Current Destination Address Register.

12.3.2.3 Maximum Count Register

This register is programmed with the maximum data unit count of the next DMA transfer. The data unit is equal to the source-to-DMA data width (byte, half-word or word). When the DMA Controller is enabled, the content of the Maximum Count Register is loaded in the Terminal Count Register.

If the Maximum count is programmed to '1', it performs a single transfer only and sets the terminal count. If the maximum count is programmed to '0', the DMA Controller does not perform any function.

The maximum terminal count is limited by a 16-bit value. ($2^{16} - 1$)

12.3.2.4 Control Register

The Control Register reads and writes the configuration of the DMA Controller. The active bits used in this register are Read/Write.

Table 12-4. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///		Dir	///	Mem2Mem	///	AddrMode	DeSize		SoBurst		SoSize		DeInc	SoInc	Enable
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE1000 + 0x14															

Table 12-5. CTRL Register Definitions

BIT	NAME	FUNCTION
31:14	///	**Reserved** Write the reset value.
13	Dir	0 = Peripheral is the source 1 = Peripheral is the destination
12	///	**Reserved** Write the reset value.
11	Mem2Mem	**Stream 3 Memory Transfer** Selects memory-to-memory transfer for Stream3. Ignored for data streams[2:0]. 0 = Stream3 is not configured for memory-to-memory transfer. 1 = Stream3 is configured for memory-to-memory transfer. The DMA Controller disregards any request from UART0TX and transfers data from source to destination as fast as possible until MaxCnt expires.
10	///	**Reserved** Write the reset value.
9	AddrMode	**Current Source/Destination Loading** Determines whether the Current Source Address Register and the Current Destination Address Register load from the Source Base registers and the Destination Base registers, respectively, when the DMA controller is enabled. 0 = Wrapping Address Mode for source and destination. Registers load from their respective Base Address registers when the DMA controller is enabled. (*Default*) 1 = Incremental Address Mode for source and destination. Registers are not reloaded from their respective Base Address registers when the DMA controller is enabled.
8:7	DeSize	**DMA-to-Destination Data Width** See Table 12-6. If you use a DeSize of '01', set the SoBurst value to '01'.
6:5	SoBurst	**Peripheral Burst Size** Defines the number of words in the peripheral burst. When the peripheral is the source, these are the number of source-to-DMA words read into the FIFO before writing FIFO contents to destination. When the peripheral is the destination, the DMA interface automatically reads the correct number of source words to compile a SoBurst of DeWidth data. When stream 3 is configured as a memory-to-memory transfer, SoBurst relates to the source side burst length. Table 12-7 shows valid values. For memory-to-peripheral operations, if bits [6:5] = '00', bits [4:3] must = '00'. If bits [6:5] = '01', bits [4:3] must = '01'.

Table 12-5. CTRL Register Definitions (Cont'd)

BIT	NAME	FUNCTION
4:3	SoSize	**Source-to-DMA data width** See Table 12-6. For memory-to-peripheral operations, if bits [6:5] = '00', bits [4:3] must = '00'. If bits [6:5] = '01', bits [4:3] must = '01'.
2	DeInc	**Current Destination Register Increment** Enables the Current Destination Register increment after each DMA-to-destination data transfer. 0 = Current Destination Register remains unchanged. 1 = Current Destination Register is incremented.
1	SoInc	**Current Source Register Increment** Enables the Current Source Register increment after each source-to-DMA data transfer. 0 = Current Source Register remains unchanged, holding the same value during the entire DMA transfer. 1 = Current Source Register increments as data transfers from a source to the DMA. The value increments by the HSIZE value at the end of the address phase of the AHB transfer.
0	Enable	**DMA Controller Enable/Disable** Enables or disables the DMA Controller. The Source Base, Destination Base, and Maximum Count Registers must be set before the DMA is enabled. The state machine clears this bit when a data transfer finishes. If the software resets this bit during a transfer, that stream interface is reset. 0 = DMA data transfer is disabled. 1 = DMA data transfer is enabled.

Table 12-6. DMA Data Width

SoSize/DeSize	AHB DATA WIDTH
00	1 byte
01	1 half-word (2 bytes)
10	1 word (4 bytes)
11	Reserved

NOTE: If you use a SoSize/DeSize of '01', set the SoBurst value to '01'.

Table 12-7. DMA Burst Size

SoBurst	AHB BURST TYPE
00	Single
01	4 incrementing
10	8 incrementing
11	16 incrementing

12.3.2.5 Current Source Registers

The Current Source Registers are 16-bit Read Only registers that hold the current value of the source address pointer. The value in the registers is used as an AHB address in a source-to-DMA data transfer over the AHB.

If the SoInc bit in the Control Register is set to '1', the value in the Current Source Registers increments as data transfers from a source to the DMA. The value increments at the end of the address phase of the AHB transfer by the HSIZE value.

If the SoInc bit is '0', the Current Source Register will hold the same value during the whole DMA data transfer.

12.3.2.6 Current Destination Registers

The Current Destination Registers are 16-bit Read Only registers that hold the current value of the destination address pointer. The value in the registers is used as an AHB address in a DMA-to-destination data transfer over the AHB.

If the DeInc bit in the Control Register is set to '1', the value in the Current Destination Registers increments as data transfers from the DMA to a destination. The value increments at the end of the address phase of the AHB transfer by the HSIZE value.

If the DeInc bit is '0', the Current Destination Register holds the same value during the entire DMA data transfer.

12.3.2.7 Terminal Count Register

The Terminal Count Register is a 16-bit Read Only register that contains the number of data units remaining in the current DMA transfer. The data unit is equal to the source-to-DMA data width (byte, half-word or word).

The register value is decremented every time data is transferred to the DMA FIFO. When the terminal count reaches zero, the FIFO content transfers to the destination and a DMA transfer is finished.

12.3.2.8 Interrupt Mask Register

The Interrupt Mask Register selects the status flag that can generate an interrupt. When exiting Reset, the default value is 0x00. The active bits used in this register are Read/Write.

Table 12-8. Interrupt Mask Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								MaskE3	MaskE2	MaskE1	MaskE0	Mask3	Mask2	Mask1	Mask0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE1000 + 0x0F0															

Table 12-9. Interrupt Mask Register Definitions

BIT	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7	MaskE3	**Data Stream3 Error Interrupt** 0 = Disables data stream3 error interrupt. 1 = Enables data stream3 error interrupt.
6	MaskE2	**Data Stream2 Error Interrupt** 0 = Disables data stream2 error interrupt. 1 = Enables data stream2 error interrupt.
5	MaskE1	**Data Stream1 Error Interrupt** 0 = Disables data stream1 error interrupt. 1 = Enables data stream1 error interrupt.
4	MaskE0	**Data Stream0 Error Interrupt** 0 = Disables data stream0 error interrupt. 1 = Enables data stream0 error interrupt.
3	Mask3	**Data Stream3 Interrupt** 0 = Disables data stream3 interrupt. 1 = Enables data stream3 interrupt.
2	Mask2	**Data Stream2 Interrupt** 0 = Disables data stream2 interrupt. 1 = Enables data stream2 interrupt.
1	Mask1	**Data Stream1 Interrupt** 0 = Disables data stream1 interrupt. 1 = Enables data stream1 interrupt.
0	Mask0	**Data Stream0 Interrupt** 0 = Disables data stream0 interrupt. 1 = Enables data stream0 interrupt.

12.3.2.9 Interrupt Clear Register

The Interrupt Clear Register clears the status flags. The active bits used in this register are Write Only. This register has no default value after Reset.

Table 12-10. CLR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								ClearE3	ClearE2	ClearE1	ClearE0	Clear3	Clear2	Clear1	Clear0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W
ADDR	0xFFFE1000 + 0x0F4															

Table 12-11. CLR Register Definitions

BIT	NAME	FUNCTION
31:8	///	**Reserved** Do not write.
7	ClearE3	**Clear/Do Not Clear ErrorInt3 Flag** 0 = Does not clear the ErrorInt3 flag in the Status Register. 1 = Clears the ErrorInt3 flag in the Status Register.
6	ClearE2	**Clear/Do Not Clear ErrorInt2 Flag** 0 = Does not clear the ErrorInt2 flag in the Status Register. 1 = Clears the ErrorInt2 flag in the Status Register.
5	ClearE1	**Clear/Do Not Clear ErrorInt1 Flag** 0 = Does not clear the ErrorInt1 flag in the Status Register. 1 = Clears the ErrorInt1 flag in the Status Register.
4	ClearE0	**Clear/Do Not Clear ErrorInt0 Flag** 0 = Does not clear the ErrorInt0 flag in the Status Register. 1 = Clears the ErrorInt0 flag in the Status Register.
3	Clear3	**Clear/Do Not Clear Int3 Flag** 0 = Does not clear the Int3 flag in the Status Register. 1 = Clears the Int3 flag in the Status Register.
2	Clear2	**Clear/Do Not Clear Int2 Flag** 0 = Does not clear the Int2 flag in the Status Register. 1 = Clears the Int2 flag in the Status Register.
1	Clear1	**Clear/Do Not Clear Int1 Flag** 0 = Does not clear the Int1 flag in the Status Register. 1 = Clears the Int1 flag in the Status Register.
0	Clear0	**Clear/Do Not Clear Int0 Flag** 0 = Does not clear the Int0 flag in the Status Register. 1 = Clears the Int0 flag in the Status Register.

12.3.2.10 Status Register

The Status Register is a Read Only register that provides status information regarding the DMA Controller. The default value after Reset is 0x00.

Table 12-12. Status Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///				Active3	Active2	Active1	Active0	ErrorInt3	ErrorInt2	ErrorInt1	ErrorInt0	Int3	Int2	Int1	Int0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE1000 + 0x0F8															

Table 12-13. Status Register Definitions

BIT	NAME	FUNCTION
31:12	///	**Reserved** Write the reset value.
11	Active3	**Data Stream3 Active/Inactive** 0 = Data stream3 is not active. 1 = Data stream3 is active.
10	Active2	**Data Stream2 Active/Inactive** 0 = Data stream2 is not active. 1 = Data stream2 is active.
9	Active1	**Data Stream1 Active/Inactive** 0 = Data stream1 is not active. 1 = Data stream1 is active.
8	Active0	**Data Stream0 Active/Inactive** 0 = Data stream0 is not active. 1 = Data stream0 is active.
7	ErrorInt3	**Data Stream3 Error Interrupt Flag** Specifies the data stream3 error-interrupt flag.
6	ErrorInt2	**Data Stream2 Error Interrupt Flag** Specifies the data stream2 error-interrupt flag.
5	ErrorInt1	**Data Stream1 Error Interrupt Flag** Specifies the data stream1 error-interrupt flag.
4	ErrorInt0	**Data Stream0 Error Interrupt Flag** Specifies the data stream0 error-interrupt flag.
3	Int3	**Data Stream3 Interrupt Flag** Specifies the data stream3 interrupt flag.
2	Int2	**Data Stream2 Interrupt Flag** Specifies the data stream2 interrupt flag.
1	Int1	**Data Stream1 Interrupt Flag** Specifies the data stream1 interrupt flag.
0	Int0	**Data Stream0 Interrupt Flag** Specifies the data stream0 interrupt flag.

The Int0-Int3 bits are the data stream interrupt flags corresponding to data stream0 through data stream3. A data stream sets its corresponding interrupt flag when a data transfer is completed (the whole packet has been transferred to its destination).

The ErrorInt[3:0] bits are the Error interrupt flags corresponding to data stream0 through data stream3. An error interrupt flag is set when its corresponding data stream's transfer is aborted due to an AHB transfer error. When this occurs, the stream is disabled until the Enable bit is again set by software.

The Active flags are used to indicate if a data stream is transferring data. It is HIGH if a data transfer is in progress. The Active flags have the same polarity as the Enable bits in the Data Stream Control Register.

The interrupt flags generated by the DMA Controller are combined and supplied to the Interrupt Controller as a combined interrupt.

Chapter 13
Color Liquid Crystal Display Controller

The Color Liquid Crystal Display (LCD) Controller information in this section pertains to the LH75401 and LH75411 SoC devices only.

This chapter discusses the Color LCD Controller (CLCDC) and its Advanced LCD Interface Peripheral (ALI) for AD-TFT, HR-TFT panels, and any technology of panel compatible with this signal system. The ALI-specific description begins in Section 13.5.

13.1 Introduction

The CLCDC provides all necessary control and data signals to interface the SoC directly to a variety of color and monochrome LCD panels, including STN and TFT panels. The ALI modifies the CLCDC output to allow the chip to connect directly to the Row and Column driver chips on superthin panels, including AD-TFT, HR-TFT, or any panel that supports this method of connection. Figure 13-1 shows a simplified diagram of the two controllers connected to the AHB, to the APB, and to each other.

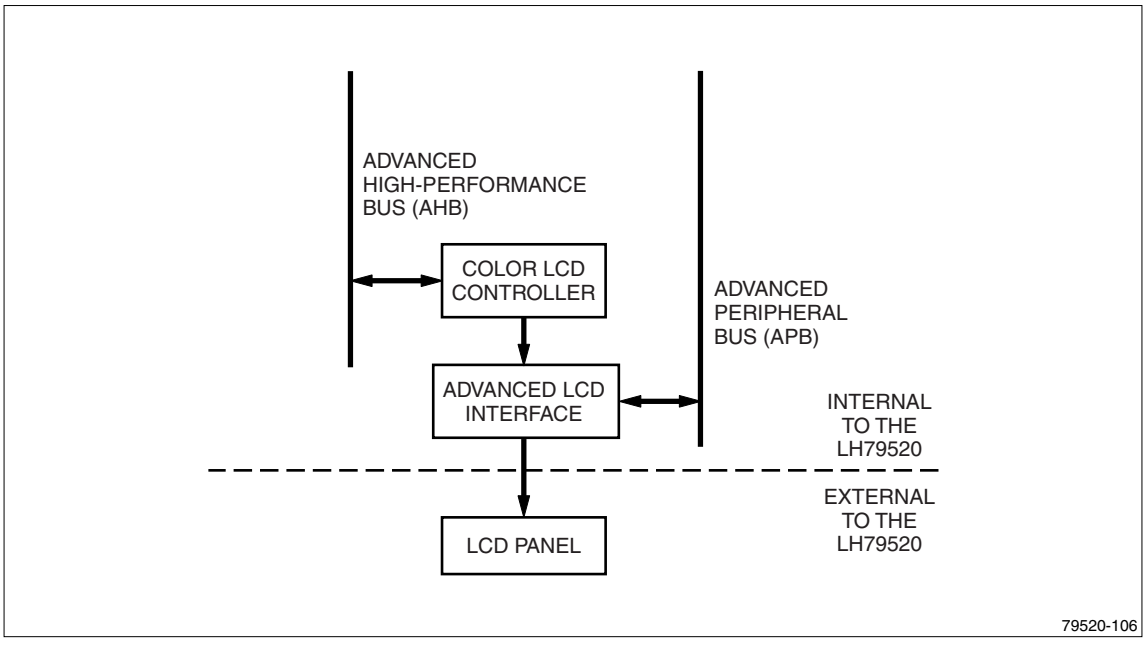


Figure 13-1. LH75401/11 LCD System, Simplified Block Diagram

13.1.1 LCD Panel Architecture

Modern technology panels, including AD-TFT and HR-TFT panels, are thinner than ever. To achieve maximum space savings, they are manufactured without the large ASICs and DC-DC converter blocks built into STN and TFT panels. See Figure 13-2.

The ASIC in STN and TFT panels decodes input data into Row and Column information and builds the timing signals. It supplies this information to the panel's Row and Column driver chips to set the proper pixels at the proper intensity and at the proper times. The DC-DC converter runs the panel's power supplies and illuminator. Including these devices in STN and TFT panels, however, comes at the cost of bulk and weight.

The ALI eliminates the need for a separate Timing ASIC, since it is able to drive the panel's Row and Column driver chips directly. The DC-DC conversion is also handled off-panel, by a separate device operating the panel's high voltage supplies and illuminator. The DC-DC conversion must be handled by a separate device, since the LH75401/11 do not supply this function.

Unless the behavior is different, this User's Guide uses the term TFT to discuss all types of TFT panels whether the panel requires timing support from the ALI or not.

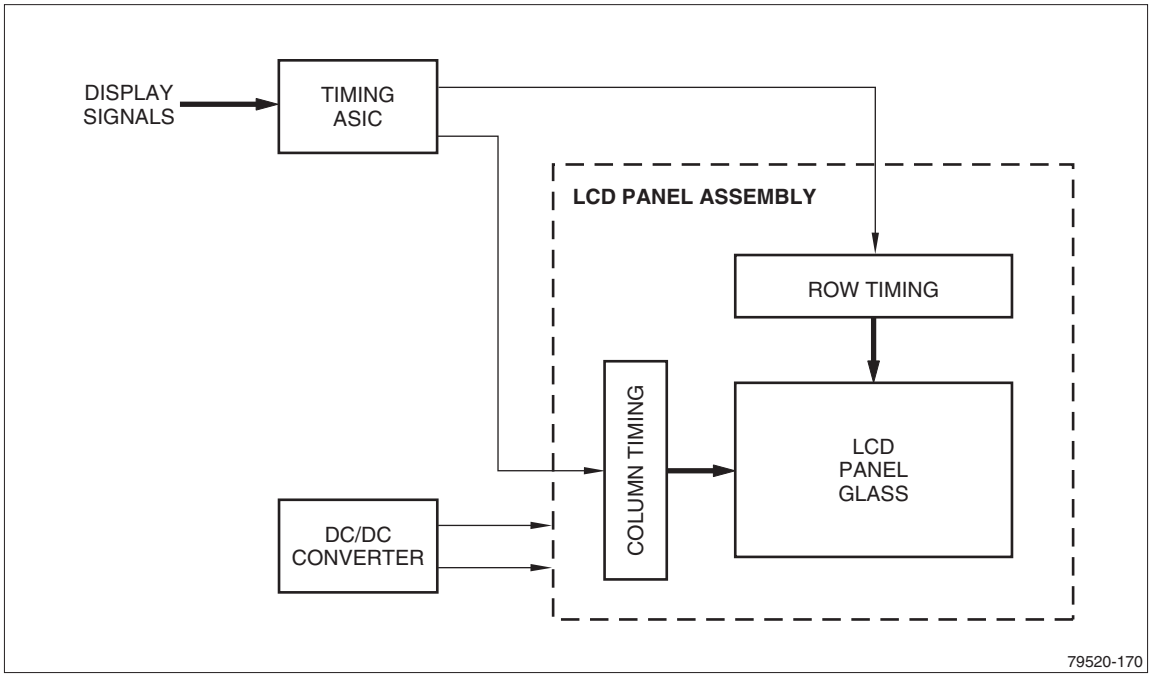


Figure 13-2. Block Diagram of a Typical Advanced LCD Panel

13.2 CLCDC Features

The CLCDC features are:

- STN, Color STN, TFT, HR-TFT, and AD-TFT interfaceable
  - Fully programmable timing controls
  - Integrated Advanced LCD Interface for displays with a lower level of integration, such as HR-TFT and AD-TFT
- Resolutions
  - Up to VGA (640 × 480 DPI), 12-bit Direct Mode Color
  - Up to SVGA (800 × 600 DPI), 8-bit Direct/Palletized Color
  - Up to XGA (1,024 × 768 DPI), 4-bit Direct Color/Grayscale
- Direct or Palletized colors
- Single and dual panels
- Supports SHARP and non-SHARP panels
- CLCDC outputs available as General Purpose Inputs/Outputs (GPIOs) if LCDC is not needed
- Supported Data Format
  - Little Endian
- 256-entry, 16-bit palette RAM physically arranged as a 128 × 32-bit RAM
- AC bias signal for STN panels and a data-enable signal for TFT panels
- Programmable Parameters for different display panels
  - Horizontal
    - Front Porch (HFP)
    - Back Porch (HBP)
    - Synchronization Pulse Width (HSW)
    - Number of Pixels per Line (PPL)
  - Vertical
    - Front Porch (VFP)
    - Back Porch (VBP)
    - Synchronization Pulse Width (VSW)
    - Number of Lines per Panel (LPP)
  - Panel-related Parameters
    - Display type: STN mono/color or TFT
    - Bits-per-pixel
    - STN 4-bit Interface Mode
    - STN Single Panel Mode
    - AC panel bias
    - Panel clock frequency
    - Number of panel clocks per line
    - Signal polarity, active HIGH or LOW
    - Little Endian data format
    - Interrupt-generation event

13.3 CLCDC Theory of Operation

The basic function of the CLCDC is to retrieve image data from system memory (the frame buffer), format the data for the LCD panel, and write it to the panel. The CLCDC also creates the control signals that cause the panel to display the formatted data in the correct place and at the correct intensity and color.

A set of numbers representing the color or gray scale value of each pixel the CLCDC displays is stored in a region of static memory known as the frame buffer. The CLCDC uses its DMA controller to fetch data from the frame buffer into its FIFO (or FIFOs for dual-panel interfaces) when the amount of pixel data falls below the FIFO watermark.

Data moves from the frame buffer into the FIFO via the AHB. The frame buffer can reside on either the external memory bus or in the internal SRAM. Because the TCM SRAM does not reside on the AHB, the frame buffer cannot be placed in TCM SRAM. (See Caution)

The Color LCD Controller (CLCDC) is an AMBA master-slave module that connects to the AHB. See Figure 13-3. The CLCDC translates pixel-coded data into the required formats and timings to drive single/dual monochrome and color LCD panels. The CLCDC supports passive Super Twisted Nematic (STN), active Thin Film Transistor (TFT), Highly Reflective TFT (HR-TFT), and Advanced TFT (AD-TFT) LCD displays.

The CLCDC has an AHB slave interface to its registers and an AHB master interface for the LCD data. The ALI has an APB slave interface to its registers. Image data flows from the AHB, through the CLCDC and the ALI, to an external LCD panel. Although a particular LCD panel may not require the ALI, the ALI must be correctly programmed because all LCD data passes through it, even if it is set to bypass mode for STN and TFT applications.

Packets of pixel-coded data are fed, via the AHB interface, to two independently programmable, 32-bit-wide DMA FIFOs. Each FIFO is 16 words deep by 32 bits wide. In Single Panel STN Mode, the LCD DMA FIFOs are made to appear as a single FIFO of twice the size. The buffered pixel-coded data is then unpacked via a pixel serializer.
See Figure 13-2.

CAUTION

Data for the LCD is always handled via DMA operations, whether to the 16KB SRAM on the AHB, or to external memory. This means that the UPBASE (Upper Panel Base Address) and LPBASE (Lower Panel Base Address) registers must always point to a valid memory location accessible via DMA before the LCD Controller is enabled. In particular, if REMAP = 0x1, the default value of UPBASE and LPBASE of 0x00000000 will cause the CLCDC to try to access TCM SRAM, which is directly accessible only to the Core. Since the LCD DMA is the highest priority on the AHB, this will hang the bus.

In 12-bits-per-pixel Mode, the CLCDC uses the unpacked data directly to generate the pixel value. In all other bit-per-pixel modes, the CLCDC uses the unpacked data to index its palette RAM; the CLCDC uses the value indexed from the palette to generate the pixel value. For STN displays, this value passes to the gray-scaling generator. For TFT displays, this value bypasses the gray-scaling generator and passes directly to the output display drivers.

The CLCDC generates a single combined interrupt to the VIC when an individual interrupt condition becomes true for upper/lower panel DMA FIFO Underflow, Base Address Update Signification, Vertical Compare, or Bus Error.

13.3.1 LCD DMA FIFOs

The CLCDC has an upper LCD DMA FIFO and a lower LCD DMA FIFO. These FIFOs can be independently controlled to cover single- and dual-panel LCDs. Each FIFO is 16 words deep by 32 bits wide. In single-panel modes the LCD DMA FIFOs are made to appear as a single FIFO of twice the size.

The watermarks within each FIFO are set so that each FIFO requests data when the level of data in a FIFO falls below the programmed watermark (either four or eight locations, as specified by bit [16] of the LCD Panel Pixel Parameter Register). An interrupt signal is asserted, if enabled, if either of the two LCD DMA FIFOs is read when they are empty.

13.3.2 Pixel Serializer

The pixel serializer reads the 32-bit-wide LCD data from the output port of the LCD DMA FIFO and extracts 12, 8, 4, 2, or 1 bits per pixel (bpp) data, depending on the operating mode. In Dual Panel Mode, data alternately is read from the upper and lower LCD DMA FIFOs. Depending on the operating mode, the extracted data is either used to point to a color/grayscale value in the palette RAM or directly applied to an LCD panel input.

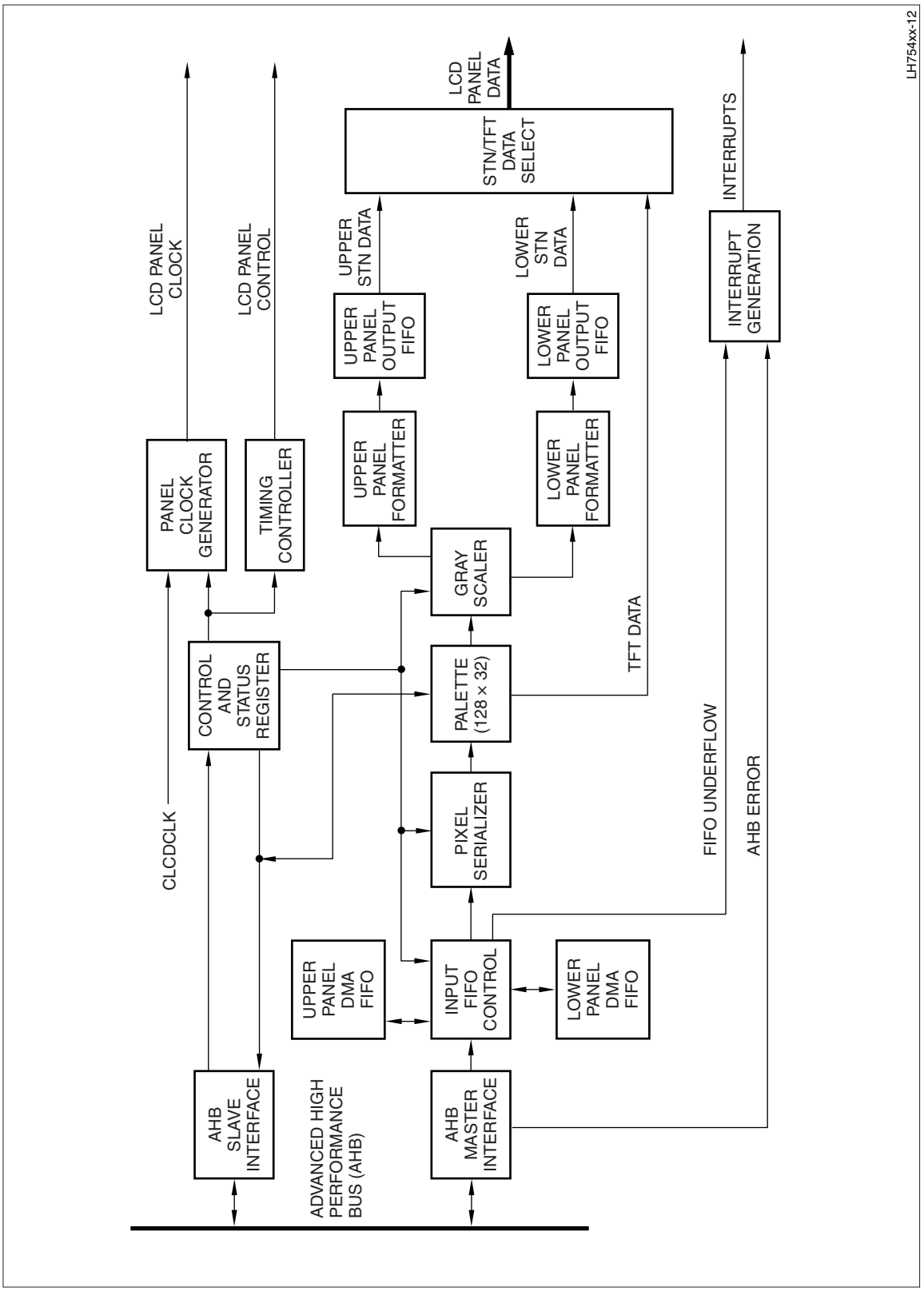


Figure 13-3. Color LCD Controller Block Diagram (LH75401 and LH75411 Only)

13.3.3 How Pixels are Stored in Memory

Table 13-2 and Table 13-3 show the data structure in each DMA FIFO word corresponding to the bpp combinations. The required data for each panel display pixel must be extracted from the data word. The first pixel value in the frame corresponds to the color value encoded in P0, the second corresponds to P1, the third to P2, and so on. This structure is the same for TFT and STN, except for 12 bpp. Table 13-1 shows the pixel arrangement on a display, with the first 32 pixels labeled p0 through p31.

Table 13-1. Pixel Display Arrangement

p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16	p17	p18	p19	p20	p21	p22	p23	p24	p25	p26	p27	p28	p29	p30	p31

Table 13-2. Frame Buffer Pixel Storage Format [31:16]

bpp	DMA FIFO OUTPUT BITS															
1	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	p31	p30	p29	p28	p27	p26	p25	p24	p23	p22	p21	p20	p19	p18	p17	p16
2	p15		p14		p13		p12		p11		p10		p9		p8	
	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4	p7				p6				p5				p4			
	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
8	p3								p2							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
12 (TFT)	p1															
			11	10	9	8		7	6	5	4		3	2	1	0
12 (STN)	p1															
		11	10	9	8		7	6	5	4		3	2	1	0	

Table 13-3. Frame Buffer Pixel Storage Format [15:0]

bpp	DMA FIFO OUTPUT BITS															
1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	p15	p14	p13	p12	p11	p10	p9	p8	p7	p6	p5	p4	p3	p2	p1	p0
2	p7		p6		p5		p4		p3		p2		p1		p0	
	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4	p3				p2				p1				p0			
	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
8	p1								p0							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
12 (TFT)	p0															
			11	10	9	8		7	6	5	4		3	2	1	0
12 (STN)	p0															
		11	10	9	8		7	6	5	4		3	2	1	0	

13.3.4 Palette RAM

The CLCDC includes a 256 × 16 bit dual-port RAM-based palette. The least-significant bit of the serialized pixel data selects the upper or lower half of the palette RAM, based on the Byte Ordering Mode.

- One port of the dual-port palette RAM is used as a Read/Write port and is connected to the AHB slave interface. Palette entries can be written and verified through this port.
- The second port of the dual-port palette RAM is used as a Read Only port and is connected to the unpacker and grayscaler.

Table 13-4 shows the bit representations of each word in the palette for TFT and STN displays.

Table 13-4. Palette Data Storage

BIT (TFT)	BIT (STN)	NAME	DESCRIPTION
31:30	31	///	Unused.
29:26	30:27	B[3:0]	Blue palette data
25	26	///	Unused.
24:21	25:22	G[3:0]	Green palette data
20	21	///	Unused.
19:16	20:17	R[3:0]	Red palette data
15:14	16:15	///	Unused.
13:10	14:11	B[3:0]	Blue palette data
9	10	///	Unused.
8:5	9:6	G[3:0]	Green palette data
4	5	///	Unused.
3:0	4:1	R[3:0]	Red palette data
—	0	///	Unused.

NOTE: Blue and red palette data can swap places, depending on the setting of bit [8] of the LCD Control Register.

13.3.5 Grayscale Algorithm

A patented grayscale algorithm drives the monochrome and color STN panels.

- For monochrome displays, the grayscale algorithm provides 15 gray levels.
- For color displays, the 3-color components (red, green, and blue) are grayscaled simultaneously. This results in 3,375 colors (15 × 15 × 15) being available for Color STN Mode.

The grayscaler transforms each 4-bit gray value into a sequence of activity-per-pixel over several frames, relying partially on the display characteristics to provide the representation of gray scales and color.

Red, green, and blue pixel data bit values from the gray scaler are shifted concurrently into the respective registers in the upper and lower panel formatter. The upper and lower panel formatters each comprise three 3-bit (red, green, and blue) shift-left registers. When enough data is available, a byte is constructed by multiplexing the registered data to the correct bit position to satisfy the LCD panel's RGB data pattern. The byte transfers to the 3-byte FIFO, which has sufficient space to store eight color pixels.

The CLCDC has four individually maskable interrupt conditions associated with a single combined interrupt. The single combined interrupt is asserted if any of the combined interrupt conditions are asserted and unmasked.

13.3.6 LCD Panel Resolutions

LCD panel resolution is expressed as the total number of horizontal pixels multiplied by the total number of vertical pixels. The CLCDC supports STN, TFT, HR-TFT, and AD-TFT LCD panels. The only requirement is for the width to be divisible by 16; there is no limit on height. Examples of resolutions supported include:

- 160 × 120
- 320 × 200
- 320 × 240
- 640 × 200
- 640 × 240
- 640 × 480 8-bit Mode (this resolution requires significant processor bandwidth)

TFT, HR-TFT, and AD-TFT LCD panels utilize color palette RAM. For these panels, each 16-bit palette entry is composed of 5 bpp, plus a common intensity bit. In addition, the total number of supported colors can be doubled from 32,768 to 65,536 if the Intensity bit is utilized and applied simultaneously to all three color components (R, G, and B).

Table 13-5 and Table 13-6 show the bpp supported for TFT, HR-TFT, and AD-TFT panels. Table 13-7 shows the bpp for monochrome STN panels.

Table 13-5. Supported TFT, HR-TFT, and AD-TFT LCD Panels

bpp	SOURCE	HR-TFT (UP TO 16-BIT BUS)
1	Palletized	2 colors selected from 65,536 available colors
2	Palletized	4 colors selected from 65,536 available colors
4	Palletized	16 colors selected from 65,536 available colors
8	Palletized	256 colors selected from 65,536 available colors
12	Direct	4:4:4 RGB

Table 13-6. Supported Color STN LCD Panels

bpp	SOURCE	COLOR STN (SINGLE AND DUAL PANEL, 8-BIT BUS)
1	Palletized	2 colors selected from 3,375 available colors
2	Palletized	4 colors selected from 3,375 available colors
4	Palletized	16 colors selected from 3,375 available colors
8	Palletized	256 colors selected from 3,375 available colors
12	Direct	4:4:4 RGB

NOTE: 3,375 colors = (15 RED) × (15 BLUE) × (15 GREEN).

Table 13-7. Supported Mono-STN LCD Panels

bpp	SOURCE	MONO STN (SINGLE, 4-BIT BUS)
1	Palletized	2-level grayscale selected from 15 levels
2	Palletized	4-level grayscale selected from 15 levels
4	Palletized	15-level grayscale selected from 15 levels

Table 13-8 shows the intensity that can be obtained from each of the 16 possible 4-bit palette combinations. Only 15 of the combinations are useful because the two middle values produce the same result.

Table 13-8. Color STN Intensities From Gray-Scale Modulation

4-BIT PALETTE VALUE	DUTY CYCLE[1]	RESULTING INTENSITY[2]
0b0000	0/90	00.0%
0b0001	10/90	11.1%
0b0010	18/90	20.0%
0b0011	24/90	26.7%
0b0100	30/90	33.3%
0b0101	36/90	40.0%
0b0110	40/90	44.4%
0b0111	45/90	50.0%
0b1000	45/90	50.0%
0b1001	50/90	55.6%
0b1010	54/90	60.0%
0b1011	60/90	66.6%
0b1100	66/90	73.3%
0b1101	72/90	80.0%
0b1110	80/90	88.9%
0b1111	90/90	100.0%

NOTES:
1. Duty cycle is determined by (pixels on ÷ (pixels on + pixels off)).
2. Resulting intensity: 000% = black, 100% = white.

13.3.7 LCD Interface Timing Signals

LCD interface timing signals are categorized as either horizontal or vertical timing signals. These signals are created by the CLCDC, optionally modified by the ALI, and applied directly to an external LCD panel with no additional external hardware required, except for Continuous Grain Silicon (CGS) panels.

13.3.7.1 LCD Horizontal Timing Signals

The horizontal components of LCD timing describe the process of writing one line of LCD data to a LCD panel and include programmable delays before and after the data is written to the panel. A line of data is composed of all pixel information for one displayed line. See Section 13.6 for timing diagrams.

13.3.7.1.1 STN Horizontal Timing Restrictions

The CLCDC's dedicated DMA system requests new data at the start of each horizontal display line. Time must be allowed for the DMA transfer operation to occur. Time must also be allowed for the data to propagate down the FIFO path within the LCD interface. These delays constitute LCD data path latency. The data path latency imposes some restrictions on the usable minimum values for horizontal back porch width when operating in the STN modes. The value restrictions are listed in Table 13-9.

Table 13-9. Usable Minimum Values Affecting STN Back Porch Width

HORIZONTAL TIMING VALUE	SINGLE-PANEL MODE	DUAL-PANEL MODE
TIMING0:HSW	3	3
TIMING0:HBP	5	5
TIMING0:HFP	5	5
TIMING2:PCD	1 × (CLCD CLOCK/3)	5 × (CLCD CLOCK/7)

NOTE: The minimum value for PCD is 4.

13.3.7.2 LCD Vertical Timing Signals

Data is written to an LCD panel in frames. Each frame is composed of a number of horizontal lines. The vertical components of LCD timing describe the process of writing one full frame to an LCD panel.

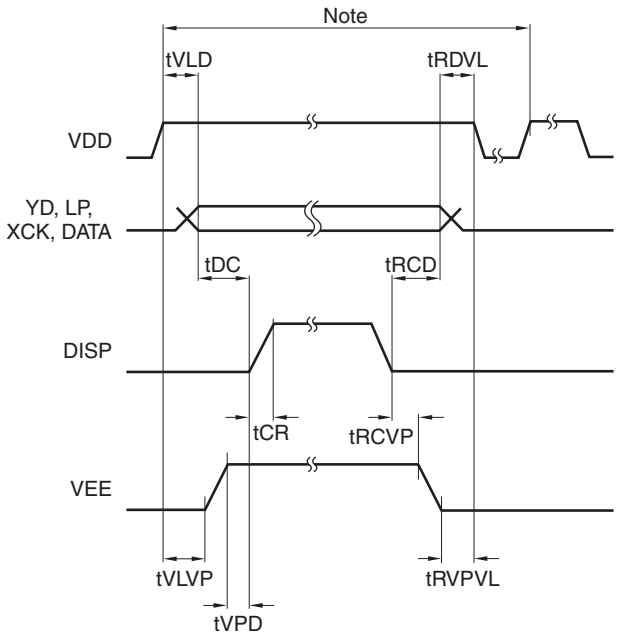
Each frame begins with a frame pulse or vertical synchronization pulse of programmable duration. Each frame pulse is followed by a programmable delay, the vertical back porch. When the vertical back porch expires, all line information for the frame is presented to the LCD panel. See Section 13.3.7.1. The line information is followed by another programmable delay, the vertical front porch.

13.3.8 LCD Power Sequencing at Turn-On and Turn-Off

Many LCD panels require ground, power for the digital logic, and high-voltage power supplies. To extend the life of these panels, the digital power must be applied before the high voltage is applied, and removed after they are removed. The logic signals driving the panel must be active before the panel voltages are applied, and the panel voltages must be removed before the logic signals are removed. This sequencing ensures that the panel is always operated with a net DC bias of 0 VDC.

Software must ensure that these conditions are met. The requisite delay is usually specified in the LCD panel's data sheet. If the proper power sequencing is not followed, the LSI drivers in the panel can latch and the display will freeze. Typically when this happens, the colors will be incorrect on STN panels. In addition the power down sequence must be followed or LCD life can be degraded.

Figure 13-4 is an example of these timing requirements for the SHARP LM057QCTT03 Color STN LCD Panel, and the accompanying timing specifications. Always refer to your specific LCD panel's Data Sheet to determine the specific turn-on and turn-off requirements for the panel being used in your application.



POWER ON

SYMBOL	ALLOWABLE VALUE	
tVLD	0 ms MIN.	1 sec. MAX.
tDC	20 ms MIN.	
tCR		100 ns MAX.
tVPD	0 ms MIN.	
tVLVP	0 ms MIN.	

POWER OFF

SYMBOL	ALLOWABLE VALUE	
tRDVL	0 ms MIN.	1 sec. MAX.
tRCD	20 ms MIN.	
tRCVP	20 ms MIN.	
tRVPVL	0 ms MIN.	

NOTE: 500 ms MIN. ON/OFF cycle time. All signals and power must be switched in this sequence during power ON/OFF cycles.

79520-161

Figure 13-4. LCD Panel Power Sequencing

13.3.8.1 Minimizing a Retained Image on the LCD

While it is very important to follow the power turn-off sequence to ensure longevity of the LCD panel, this sequence alone will not ensure there is no retained image (ghosting) left on the LCD panel after the LCD has been powered down.

This ghost bleeds away slowly after powering down the LCD panel. It is most noticeable with LCD panels utilizing HR-TFT and AD-TFT technologies to light the LCD. With these types of LCD panels the ambient light alone is enough to make the retained image visible. TFT-type LCD panels also have a retained image, but it is typically not as visible once the backlight source is turned off.

To minimize the appearance of a retained image on HR-TFT and AD-TFT LCD panels, software should write a complete frame of all 1's (white) to the LCD just prior to initiating the turn-off sequence.

To minimize the appearance of a retained image on a TFT LCD panel, software should write a complete frame of all 0's (black) just prior to initiating the turn-off sequence.

In all cases the illumination source should be turned off prior to initiating the turn-off sequence.

13.4 CLCDC Programmer's Model

The base address for the CLCDC is:

CLCDC Base Address: 0xFFFF4000

The following locations are reserved and must not be used during normal operation:

- Locations at offsets 0x030 through 0x1FC
- Locations at offsets 0x400 through 0x7FF

13.4.1 CLCDC Register Summary

Table 13-10. CLCDC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Timing0	0x000	RW	0x00000000	Horizontal Axis Timing Control
Timing1	0x004	RW	0x00000000	Vertical Axis Timing Control
Timing2	0x008	RW	0x0000000	Clock and Signal Polarity Control Register
///	0x00C	RW		Reserved
UPBASE	0x010	RW	0x0000000	Upper Panel Frame Buffer Base Address Register
LPBASE	0x014	RW	0x00000000	Lower Panel Frame Buffer Base Address Register
INTRENABLE	0x018	RW	0x00000000	Interrupt Enable Register
Ctrl	0x01C	RW	0x0000	Panel Parameters, Panel Power, and Control
Status	0x020	RW	0x00000000	Raw Interrupt Status Register
Interrupt	0x024	R	0x00000000	Final Masked Interrupts Register
UPCURR	0x028	R	0x00000000	Upper Panel Frame Buffer Current Address Register
LPCURR	0x02C	R	0x00000000	Lower Panel Frame Buffer Current Address Register
///	0x030 - 0x1FC		0x00000	Reserved
Palette	0x200 - 0x3FC	RW		256 × 16-bit Color Palette Register. Palette is addressed at 32 bits.
///	0x400 - 0x7FF			Reserved

13.4.2 CLCDC Register Definitions

13.4.2.1 Horizontal Timing Panel Control Register

The Timing0 Register controls:

- Horizontal Synchronization Pulse Width (HSW)
- Horizontal Front Porch (HFP) period
- Horizontal Back Porch (HBP) period
- Pixels-Per-Line (PPL)

Table 13-11. Timing0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	HBP								HFP							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	HSW								PPL						///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
ADDR	0xFFFF4000 + 0x00															

Table 13-12. Timing0 Register Definitions

BIT	NAME	DESCRIPTION
31:24	HBP	**Horizontal Back Porch** Specifies the number of LCDDCLK periods between the falling edge of the LCDLP signal and the beginning of valid data; that is, the number of pixel clock periods inserted at the beginning of each line or row of pixels. After the line clock for the previous line has been de-asserted, the value in HBP counts the number of pixel clocks to wait before starting the next display line. HBP can generate a delay of 1 to 256 pixel clock cycles. HBP = (LCDDCLK periods) – 1
23:16	HFP	**Horizontal Front Porch** Specifies the number of LCDDCLK periods between the end of valid data and the rising edge of LCDLP; that is, the number of pixel clock intervals at the end of each line or row of pixels before the LCD line clock is pulsed. Once a complete line of pixels is transmitted to the LCD driver, the value in HFP counts the number of pixel clocks to wait before asserting the line clock. HFP can generate a period of 1 to 256 pixel clock cycles. HFP = (LCDDCLK periods) – 1
15:8	HSW	**Horizontal Synchronization Pulse Width** Specifies the width of the LCDLP signal in LCDDCLK periods; that is, the pulse width of the line clock in Passive Mode or the horizontal synchronization pulse in Active Mode. HSW = (LCDDCLK periods) – 1
7:2	PPL	**Pixels-Per-Line** Specifies the number of pixels, between 16 and 1,024, in each line or row of the screen. PPL counts the number of pixel clocks before HFP occurs. PPL = (Actual pixels-per-line/16) – 1 Actual pixels-per-line = 16 × (PPL + 1)
1:0	///	**Reserved** Reading returns 0. Write the Reset value.

13.4.2.2 Horizontal Timing Restrictions

The LCD DMA requests new data at the start of a horizontal display line. Some time must be allowed for the DMA transfer and for the data to propagate down the FIFO path in the LCD interface. The data path latency forces some restrictions on the usable minimum values for horizontal porch width in STN Mode.

The minimum values are HSW = 2 and HBP = 2.

Single Panel Mode:

- HSW = 3
- HBP = 5
- HFP = 5
- Panel Clock Divisor (PCD) = 1 (CLCDCLK/3)

Dual Panel Mode:

- HSW = 3
- HBP = 5
- HFP = 5
- PCD = 5 (CLCDCLK/7)

If sufficient time is given at the start of the line (for example, setting HSW = 6, HBP = 10), data will not get corrupted for PCD = 4 (minimum value).

NOTE: CLCDCLK is a separate clock provided by the Reset Clock and Power Controller (RCPC). For more information, see Chapter 9.

13.4.2.3 Vertical Timing Panel Control Register

The Timing1 Register controls the:

- Number of Lines-Per-Panel (LPP)
- Vertical Synchronization Pulse Width (VSW)
- Vertical Front Porch (VFP) period
- Vertical Back Porch (VBP) period

Table 13-13. Timing1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	VBP								VFP							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	VSW						LPP									
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x04															

Table 13-14. Timing1 Register Definitions

BIT	NAME	DESCRIPTION
31:24	VBP	**Vertical Back Porch** Specifies the number of inactive lines at the start of a frame, after the vertical synchronization period (the number of line clocks inserted at the beginning of each frame). The VBP count starts just after the Vertical Synchronization signal for the previous frame has been negated for Active Mode or the extra line clocks have been inserted as specified by the VSW bit field in Passive Mode. After this has occurred, the count value in VBP sets the number of line clock periods inserted before the next frame. VBP generates from 0 – 255 extra line clock cycles. STN displays: Program to zero, otherwise reduced contrast will result.
23:16	VFP	**Vertical Front Porch** Specifies the number of inactive lines at the end of the frame, before the vertical synchronization period (the number of line clocks to insert at the end of each frame.) Once a complete frame of pixels is transmitted to the LCD display, the value in VFP counts the number of line clock periods to wait. After the count has elapsed, the Vertical Synchronization signal is asserted in Active Mode or extra line clocks are inserted as specified by the VSW bit field in Passive Mode. VFP generates from 0 – 255 line clock cycles. STN Modes: Additional horizontal line clocks (specified by VSW) are inserted after VFP expires. To avoid reduced contrast, program VFP = 0 for STN displays.
15:10	VSW	**Vertical Synchronization Pulse Width** Specifies the pulse width of the vertical synchronization pulse in terms of horizontal synchronization pulse periods. VSW = (LCDLP periods) – 1 STN Modes: Additional horizontal line clocks (specified by VSW) are inserted after VFP expires. To avoid reduced contrast, program VFP = 0 for STN displays.

Table 13-14. Timing1 Register Definitions (Cont'd)

BIT	NAME	DESCRIPTION
9:0	LPP	**Lines Per Panel** Specifies the number of active lines per screen (the total number of lines or rows on the LCD panel). LPP is a 10-bit value allowing between 1 and 1,024 lines. LPP = (Active Lines) – 1 Dual-panel displays: The two panels are assumed to be of equal sizes; program with the number of lines in either panel (but not their sum). ALI Active modes: LPP and LCDREV must be considered together. Program LPP to an odd number of lines for AD-TFT and HR-TFT panels. These panels use the LCDREV signal as an AC bias signal, toggling it HIGH and LOW in alternating frames; this results in a net 0 V DC bias applied to any line of the panel. When correctly programmed, LCDREV will be HIGH for a given line of a panel during one frame, and LOW for that same line during the next frame. If LPP specifies an even number of Lines Per Panel, this signal will be mismatched to the display's lines and these lines will not recieve a 0 V DC net bias, resulting in long-term damage to the panel.

13.4.2.4 Clock and Signal Polarity Control Register

The Timing2 Register controls the CLCDC timing.

Table 13-15. Timing2 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///					BCD	CPL									
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	IOE	IPC	HIS	IVS	ACB					///	PCD				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x08															

Table 13-16. Timing2 Register Definitions

BIT	NAME	DESCRIPTION
31:27	///	**Reserved** Reading returns 0. Write the Reset value.
26	BCD	**Bypass Pixel Clock Divider** 1 = Bypass the pixel clock divider logic 0 = Use the pixel clock divider logic See the description for PCD, below.
25:16	CPL	**Clocks Per Line** Specifies the number of actual LCDDCLK clocks to the LCD panel on each line. This value is the number of pixels per line divided by either 1 (TFT), 4 or 8 (for mono passive), 2-2/3 (for color passive), minus one. This must be correctly programmed in addition to PPL for the LCD Controller to work correctly. TFT panels: CPL = (PPL – 1) 4-bit mono STN panels: CPL = ((PPL/4) – 1) 8-bit mono STN panels: CPL = ((PPL/8) – 1) Color STN panels: CPL = (((3 x PPL) / 8) – 1)
15	///	**Reserved** Reading returns 0. Write the Reset value.
14	IOE	**Invert Output Enable** IOE applies only to TFT modes and should be programmed to 0 for all other modes. In this mode, the LCDEN pin indicates to the LCD panel when valid display data is available. IOE selects the active polarity of this output enable signal. In Active Display Mode, data is driven onto the LCD data lines at the programmed edge of LCDDCLK when LCDEN is asserted. 0 = LCDEN output pin is active HIGH 1 = LCDEN output pin is active LOW
13	IPC	**Invert Panel Clock** Selects the edge of the panel clock on which pixel data is driven out onto the LCD data lines. 0 = Data is driven on the LCDs data lines on the rising-edge of LCDDCLK. 1 = Data is driven on the LCDs data lines on the falling-edge of LCDDCLK.
12	HIS	**Invert Horizontal Synchronization** Inverts the polarity of the LCDLP signal. 0 = LCDLP pin is active HIGH and inactive LOW. 1 = LCDLP pin is active LOW and inactive HIGH.
11	IVS	**Invert Vertical Synchronization** Inverts the polarity of the CLFP signal. 0 = CLFP pin is active HIGH 1 = CLFP pin is active LOW

Table 13-16. Timing2 Register Definitions (Cont'd)

BIT	NAME	DESCRIPTION
10:6	ACB	**AC Bias Pin Frequency** ACB sets the frequency of the LCDENAB signal in STN mode. STN displays require the pixel voltage polarity to be periodically reversed to prevent damage due to DC charge accumulation. ACB = (line clocks) – 1 TFT modes: This field has no effect if the CLCDC is operating in TFT mode because the LCDENAB pin is instead utilized for a Data Enable signal.
5	///	**Reserved** Reading returns 0. Write the Reset value.
4:0	PCD	**Panel Clock Divisor** Derives the LCD panel clock frequency LCDDCLK from the input CLCDC clock frequency. LCDDCLK = CLCDCLK/(PCD + 2) Mono STN modes: LCDDCLK for mono STN panels with a 4- or 8- bit interface should be programmed to be 1/4 (or 1/8) the desired individual pixel clock rate. Color STN modes: Color STN displays receive multiple pixels during each clock cycle. The pixel data for Color STN displays is stored and transferred in packed format, with each pixel represented by three bits (R,G and B). Therefore, one byte contains the pixel data for 2 2/3 pixels (RGB,RGB,RG) and three bytes contain the pixel data for eight complete pixels. For Color STN panels, each LCDDCLK cycle transfers one byte, containing 2 2/3 pixels, to the panel. LCDDCLK should be programmed to be as close as possible to 3/8 the desired individual pixel clock rate. TFT mode: For TFT displays, the pixel clock divider can be bypassed by programming Timing2 BCD bit (bit [26]) = 1.

13.4.2.5 Upper Panel Frame Buffer Base Address Register

The UPBASE Register is one of two Color LCD DMA Base Address Registers (the other is LPBASE). Together with LPBASE, this Read/Write register programs the base address of the frame buffer.

UPBase is used for:

- TFT displays
- Single-panel STN displays
- The upper panel of dual-panel STN displays.

UPBase (and LPBase for dual panels) must be initialized before enabling the CLCDC. Optionally, the value can be changed mid-frame to allow double-buffered video displays to be created. These registers are copied to the corresponding current registers at each LCD vertical synchronization. This event causes the LNBU bit and an optional interrupt to be generated. The LNBU bit indicates that it is safe to update both the UPBASE and LPBASE Registers. The interrupt can be used to reprogram the base address when generating double-buffered video.

Table 13-17. UPBASE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDUPBASE															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDUPBASE														///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
ADDR	0xFFFF4000 + 0x10															

Table 13-18. UPBASE Register Definitions

BIT	NAME	DESCRIPTION
31:2	LCDUPBASE	**LCD Upper Panel Base Address** Specifies the starting address of the upper-panel frame data in memory and is word aligned.
1:0	///	**Reserved** Reading returns 0. Write the Reset value.

13.4.2.6 Lower Panel Frame Buffer Base Address Register

The LPBASE Register is one of two Color LCD DMA Base Address Registers (the other is UPBASE). Together with UPBASE, this Read/Write register programs the base address of the frame buffer.

LPBase is used for the lower panel of dual-panel STN displays. UPBase must be initialized (and LPBase for dual panels) before enabling the CLCDC. Optionally, the value can be changed mid-frame to allow double-buffered video displays to be created. These registers are copied to the corresponding current registers at each LCD vertical synchronization. This event causes the LNBU bit and an optional interrupt to be generated. The LNBU bit indicates that it is safe to update both the UPBASE and LPBASE Registers. The interrupt can be used to reprogram the base address when generating double-buffered video.

Table 13-19. LPBASE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDLPBASE															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDLPBASE														///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
ADDR	0xFFFF4000 + 0x14															

Table 13-20. LPBASE Register Definitions

BIT	NAME	DESCRIPTION
31:2	LCDLPBASE	**LCD Lower Panel Base Address** Specifies the starting address of the lower panel frame data in memory and is word aligned.
1:0	///	**Reserved** Reading returns 0. Write the Reset value.

13.4.2.7 Interrupt Enable Register

INTRENABLE is the Interrupt Enable Register. Setting bits within this register enables the corresponding raw interrupt Status bit values to be passed to the Raw Interrupt Status Register (see Chapter 13). The active bits used in this register are Read/Write.

Table 13-21. INTRENABLE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											MBERRINTRENB	VCOMPINTRENB	LNBUINTRENB	FUFINTRENB	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	R
ADDR	0xFFFF4000 + 0x18															

Table 13-22. INTRENABLE Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Reading returns 0. Write the Reset value.
4	MBERRINTRENB	**AHB Master Error Interrupt Enable** 1 = Enables the AHB Master Error Interrupt to be passed to the Raw Interrupt Status Register.
3	VCOMPINTRENB	**Vertical Compare Interrupt Enable** 1 = Enables the Vector Compare Interrupt to be passed to the Raw Interrupt Status Register.
2	LNBUINTRENB	**Next Base Update Interrupt Enable** 1 = Enables the Next Base Update Interrupt to be passed to the Raw Interrupt Status Register.
1	FUFINTRENB	**FIFO Underflow Interrupt Enable** 1 = Enables the FIFO Underflow Interrupt to be passed to the Raw Interrupt Status Register.
0	///	**Reserved** Reading returns 0. Write the Reset value.

13.4.2.8 Panel Parameters, Panel Power, and Control Register

Ctrl is a Read/Write register that controls the mode in which the CLCDC operates. The active bits used in this register are Read/Write.

Table 13-23. Ctrl Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															WATERMARK
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LDmaFIFOTME	///	LcdVComp		LcdPwr	///		BGR	LcdDual	///	LcdTFT	LcdBW	LcdBpp			LcdEn
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x1C															

Table 13-24. Ctrl Register Definitions

BIT	NAME	DESCRIPTION
31:17	///	**Reserved** Reading returns 0. Write the Reset value.
16	WATERMARK	**LCD DMA FIFO Watermark Level** 0 = WATERMARK is HIGH when either of the two LCD DMA FIFOs have four or more empty locations. 1 = WATERMARK is HIGH when either of the LCD DMA FIFOs have eight or more empty locations.
15	LDmaFIFOTME	**LCD DMA FIFO Test Mode Enable** 0 = LCD DMA FIFO inaccessible to user. 1 = LCD DMA FIFO Read/Write access for FIFO RAM testing. Set this bit only when LCD is disabled via bit [0] of this register.
14	///	**Reserved** Reading returns 0. Write the Reset value.
13:12	LcdVComp	**LCD Vertical Compare** Program to generate an interrupt at: 00 = start of vertical synchronization 01 = start of back porch 10 = start of active video 11 = start of front porch
11	LcdPwr	**LCD Power Enable** This bit causes the LCDVDDEN pin to toggle. 0 = LCD is OFF, LCDVDDEN pin is LOW. 1 = LCD is ON, LCDVDDEN pin is HIGH when LcdEn is HIGH.
10:9	///	**Reserved** Reading returns 0. Write the Reset value.

Table 13-24. Ctrl Register Definitions (Cont'd)

BIT	NAME	DESCRIPTION
8	BGR	**RGB or BGR Format Selection** 0 = RGB normal output. 1 = BGR red and blue swapped.
7	LcdDual	**LCD Interface is Dual-Panel STN** Program to 0 for modes other than dual-panel STN. 0 = Single-panel LCD is in use. 1 = Dual-panel LCD is in use. If this bit is set and bit [4] is cleared, unexpected results occur.
6	///	**Reserved** Always write 0.
5	LcdTFT	**LCD is TFT** 0 = LCD is an STN display — use grayscaler. 1 = LCD is TFT — do not use grayscaler.
4	LcdBW	**STN LCD is Monochrome (Black and White)** 0 = STN LCD is color. 1 = STN LCD is monochrome. If this bit is cleared and bit [7] is set, unexpected results occur.
3:1	LcdBpp	**LCD Bits per Pixel** 000 = 1 bpp 001 = 2 bpp 010 = 4 bpp 011 = 8 bpp 100 = 12 bpp (16 bits fetched from memory) 101 = Reserved 110 = Reserved 111 = Reserved
0	LcdEn	**LCD Controller Enable** 0 = LCD Controller is disabled. 1 = LCD Controller is enabled. LCD displays require the logic signals to be running before power is applied. For this reason, the LCD's power-on control (bit [11]) is not set to 1 unless both LcdEn and LcdPwr are set to 1. Many LCD displays require the LcdEn to be set to 1 approximately 20 ms before LcdPwr is set to 1 for powering up. Likewise, LcdPwr should be set to 0 (20 ms) before LcdEn is set to 0 for powering down. 20 ms is a typical value. Refer to the documentation that came with your panel for the correct time.

13.4.2.9 Raw Interrupt Status Register

Status is the Raw Interrupt Status Register. This register is Read/Write.

- On a read, this register returns five bits that may generate interrupts when set.
- On writes to this register, a bit value of '1' clears the interrupt corresponding to that bit. Writing a '0' has no effect.

Table 13-25. Status Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											MBERROR	Vcomp	LNBU	FUF	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RC	RC	RC	RC	R
ADDR	0xFFFF4000 + 0x20															

Table 13-26. Status Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Reading returns 0. Write the Reset value.
4	MBERROR	Master Bus Error Interrupt Asserted when an ERROR response is received by the master interface during a transaction with a slave. When an ERROR response is encountered, the master interface enters an error state until it receives a signal that the error has been cleared. 1 = Interrupt enabled 0 = Interrupt disabled
3	VCOMP	**Vertical Compare Interrupt** Asserted when one of four vertical display regions, selected via [13:12] of the LCD Control Register, is reached. The interrupt can be made to occur at the start of Vertical Synchronization, Back Porch, Active Video, and Front Porch. 1 = Interrupt enabled 0 = Interrupt disabled
2	LNBU	**LCD Next Base Address Update Interrupt** Asserted when either the UPBASE or the LPBASE values are transferred to the UPCURR or LPCURR incrementer, respectively. This indicates to the system that it can safely update the UPBASE or the LPBASE Register with new frame base addresses if required. 1 = Interrupt enabled 0 = Interrupt disabled
1	FUF	**FIFO Underflow Interrupt** Asserted when internal data is requested from an empty LCD DMA FIFO. Internally, individual upper and lower panel LCD DMA FIFO Underflow Interrupt signals are generated and this is the single combined version of these. 1 = Interrupt enabled 0 = Interrupt disabled
0	///	**Reserved** Reading returns 0. Write the Reset value.

13.4.2.10 Final Masked Interrupts Register

The Interrupt Register is a Read Only register. It is a bit-by-bit logical AND of the Raw Interrupt Status Register (see Section 13.4.2.9) and Interrupt Enable Register (see Section 13.4.2.7). Interrupt lines correspond to each interrupt. A logical OR of all interrupts is provided to the System Interrupt Controller.

Table 13-27. Interrupt Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											MBERRINTRENB	VCOMPINTRENB	LNBUINTRENB	FUFINTRENB	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFF4000 + 0x24															

Table 13-28. Interrupt Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Reading returns 0. Write the Reset value.
4	MBERRORINTR	**AHB master error interrupt status bit** 1 = Interrupt asserted 0 = Interrupt cleared
3	VCOMPINTR	**Vertical compare interrupt status bit** 1 = Interrupt asserted 0 = Interrupt cleared
2	LNBUINTR	**LCD next base address update interrupt status bit** 1 = Interrupt asserted 0 = Interrupt cleared
1	FUFINTR	**FIFO underflow interrupt status bit** 1 = Interrupt asserted 0 = Interrupt cleared
0	///	**Reserved** Reading returns 0. Write the Reset value.

13.4.2.11 LCD Upper Panel Frame Buffer Current Address Register

UPCURR and LPCURR are registers that contain an approximate value of the upper and lower panel data DMA addresses when read. The registers can change at any time and provide a coarse indication of the current LCD DMA memory pointer.

Table 13-29. UPCURR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDUPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDUPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFF4000 + 0x28															

Table 13-30. UPCURR Register Definitions

BIT	NAME	DESCRIPTION
31:0	LCDUPCURR	**Current Upper Panel Data DMA Address** Contains the approximate current upper panel data DMA address.

Table 13-31. LPCURR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDLPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDLPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFF4000 + 0x2C															

Table 13-32. LCDLPCURR Register Definitions

BIT	NAME	DESCRIPTION
31:0	LCDLPCURR	**Current Lower Panel Data DMA Address** Contains the approximate current lower panel data DMA address.

13.4.2.12 256 × 16-bit Color Palette Register

The color map in the SoC of 65,535 colors is mapped into a group of palette entries, comprising the Palette Registers. The upper four bits of each palette entry is used for best contrast in STN displays. The palette is organized as seen in Table 13-33.

The Palette Registers contain 256 palette entries organized as 128 locations of two 16-bit entries per word, forming a 32-bit word. TFT and STN displays use 12 of the palette entry bits, yielding 3,375 colors of the total 32,768 colors. Each word location contains two palette entries. This means that 128 word locations are used for the palette. The color that is displayed is based on the setting of bit [8] of the Control Register (see Section 13.5.5.2).

Due to the hardware requirements of STN displays, each palette entry is shifted left by one bit. For TFT displays, bits 3:0 of the palettes are used. For color STN displays, bits [4:1] of the red, blue and green palettes are used. For monochrome STN displays, only the red palette bits [4:1] are used. Note that the palettes are accessed 32 bits at a time.

Table 13-33. Palette Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///	BLUE PALETTE					GREEN PALETTE					RED PALETTE				
RESET																
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	BLUE PALETTE					GREEN PALETTE					RED PALETTE				
RESET																
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x200 to 0xFFFF4000 + 0x3FC															

Table 13-34. Palette Register Use for TFT and STN

BIT (TFT)	BIT (STN)	NAME	DESCRIPTION
31:30	31	///	Unused
29:26	30:27	B[3:0]	Blue palette data
25	26	///	Unused
24:21	25:22	G[3:0]	Green palette data
20	21	///	Unused
19:16	20:17	R[3:0]	Red palette data
15:14	16:15	///	Unused
13:10	14:11	B[3:0]	Blue palette data
9	10	///	Unused
8:5	9:6	G[3:0]	Green palette data
4	5	///	Unused
3:0	4:1	R[3:0]	Red palette data
	0	///	Unused

NOTE: Blue and red palette data can swap places, depending on the setting of bit [8] of the LCD Control Register.

13.4.3 CLCDC Interrupts

The single combined interrupt, CLCDINTR, is used to drive the Vectored Interrupt Controller (VIC). If any of the four interrupt conditions occurs, this signal is asserted. CLCDINTR drives the VIC.

Each of the four individual maskable interrupt conditions is enabled or disabled by changing the mask bits in the INTRENABLE Register. Provision of individual outputs, along with a combined interrupt output, allows the use of either a global interrupt service routine or modular device drivers to handle interrupts. The status of the individual interrupt sources can be read from the Status Register.

13.5 Advanced LCD Interface

The Advanced LCD Interface (ALI) provides the additional processing required to interface the LH75401 and LH75411 to AD-TFT, HR-TFT, or any display technology that uses this method of connection. Figure 13-5 shows the ALI between the CLCDC and the LCD output pins.

The ALI is programmed via its16-bit APB interface and receives control signals and display data from the CLCDC. The ALI converts the display data to a format suitable for direct connection to the Row and Column driver ICs in AD-TFT, HR-TFT displays, or any display using similar technology.

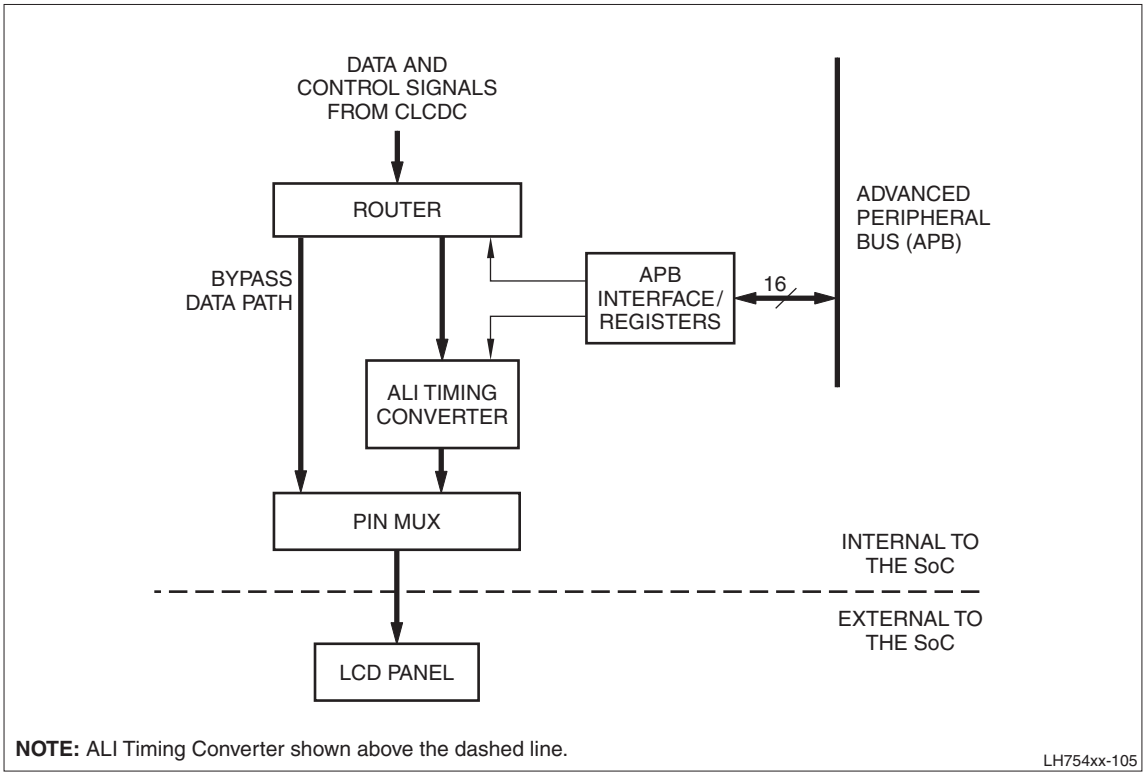


Figure 13-5. ALI Simplified Block Diagram

13.5.1 ALI Theory of Operation

All ALI Control and Status Registers can be accessed through the APB. One of the registers, the ALI Setup Register, can be programmed to select Bypass Mode or Active Mode.

- In Bypass Mode, LCD Controller signals are passed directly to the pins.
- In Active Mode, TFT data and control signals from the LCD Controller generate a set of signals for driving an AD-TFT or HR-TFT display.

NOTE: This Chapter of the User's Guide has been updated. To aid in backward compatibility, the former register names have been included in the Register Reference. There is no change to the registers themselves; either in function or location.

Selecting Active Mode re-times the data to the falling edge of the output clock. The formatter also provides the:

- Features of normal scanning signals for vertical and horizontal scan
- Generation of source driver, gate driver, and voltage-preparation control signals.

The timing parameters for the Active Mode are register-programmable. When using Active Mode, program the ALI Setup Register first, followed by ALI Timing registers 1 and 2. After these registers are programmed, the LCD Controller can be enabled and the ALI Control Register can be used.

The ALI generates the MOD signal automatically. By default, activation of MOD occurs 2 SPS rising edge clocks after activation of the controller. This can be reprogrammed for a longer or shorter wait, or can be overridden via the ALI Control Register.

13.5.2 ALI Operating Modes

The ALI has two operating modes: Bypass Mode or Active Mode. The ALI Setup Register setting specifies the operating mode of the ALI (see Section 13.5.5.1).

13.5.2.1 Bypass Mode

Bypass Mode is the default mode after a System Reset. In this mode, the input signals from the CLCDC are passed directly to the output pins without any signal reformatting. This mode is used when the LCD Controller is required to operate in normal TFT Mode or when using the LCD Controller in STN Mode.

13.5.2.2 Active Mode

In Active Mode, TFT data and the control signals from the LCD Controller generate a set of signals for driving an AD-TFT or HR-TFT display. Active Mode also re-times the data to the falling edge of the output clock. The formatter provides the scanning signals for vertical and horizontal scanning. It also generates the source driver, gate driver, and voltage preparation control signals.

13.5.3 ALI Programmer's Model

The base address for the ALI is:

ALI Base: 0xFFFE4000 (Formerly HRTFTC Base)

Locations at offsets 0x010 through 0xFFF are reserved and must not be used during normal operation.

NOTE: This Chapter of the User's Guide has been updated. To aid in backward compatibility, the former register names have been included in the Register Reference. There is no change to the registers themselves; either in function or location.

13.5.4 ALI Register Summary

Table 13-35. ALI Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
ALISetup	0x000	RW	0x000C	ALI Setup Register (Formerly HRTFTCSetup)
ALICTRL	0x004	RW	0x0000	ALI Control Register (Formerly HRTFTCCTRL)
ALITiming1	0x008	RW	0x1000	ALI Timing Register 1 (Formerly HRTFTCTiming1)
ALITiming2	0x00C	RW	0x0000	ALI Timing Register 2 (Formerly HRTFTCTiming2)
///	0x010 - 0xFFF			Reserved

13.5.5 ALI Register Definitions

13.5.5.1 Advanced LCD Interface Setup Register

The ALISetup Register enables Active mode (Timing Conversion to AD-TFT or HR-TFT) or disables it so that signals from the LCDC pass through unaltered. It also configures the Pixels Per Line when in Active mode. The Pixels Per Line value in this register should be the same as the value entered in the Timing0 register. The active bits used in this register are Read/Write.

This register was formerly known as HRTFTCSetup.

Table 13-36. ALISetup Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///			PPL									///			CR
RESET	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE4000 + 0x000															

Table 13-37. ALISetup Register Definitions

BITS	NAME	FUNCTION
31:13	///	**Reserved** Reading returns 0. Write the Reset value.
12:4	PPL	**Pixels Per Line** Specifies the number of pixels per line. PPL = (Actual Pixels per line) – 1.
3:1	///	**Fixed Bits** Read as 110b. Always write 110b to these bits; otherwise unexpected results may occur.
0	CR	**Active Mode Select** Selects Active mode. Change the ALI mode only when the CLCDC is disabled. 0 = Bypass Mode (Signals pass through unchanged) 1 = Active Mode

13.5.5.2 Advanced LCD Interface Control Register

ALICTRL is the Control Register. The Control Register enables and controls output signals. The active bits used in this register are Read/Write. This register was formerly known as HRTFTCCTRL.

Table 13-38. ALICTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						MODOVRD	MODVAL	///			EN0	DISP	///	CLSEN	SPSEN
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	RW	RW	R	R	R	RW	RW	R	RW	RW
ADDR	0xFFFE4000 + 0x004															

Table 13-39. ALICTRL Register Definitions

BITS	NAME	FUNCTION
31:10	///	**Reserved** Reading returns 0. Write the Reset value.
9	MODOVRD	**MOD Signal Override Enable** Puts the value of MODVAL directly onto the MOD signal: 0 = LCDMOD pin goes HIGH after the SPS periods specified by the MODDEL field of the TIMING1 Register. 1 = LCDMOD pin equals the state of MODVAL bit in this register.
8	MODVAL	**Mod Signal Value** Specifies the value to force onto the MOD signal.
7:5	///	**Reserved** Reading returns 0. Write the Reset value.
4	EN0	**LCDVEEEN Output Enable** General purpose output enable to LCDVEEEN (only in Bypass Mode).
3	DISP	**Display Control Signal Output** Controls the output of the Display Control signal, LCDDSPLEN (only in Bypass Mode).
2	///	**Reserved** Reading returns 0. Write the Reset value.
1	CLSEN	**LCDCLS Signal** Controls the Tristate Enable signal LCDCLS. STN or TFT (Bypass) modes: Reading returns 0. Write the Reset value. Active mode: Enables or disables the generation of the LCDCLS (Gate Driver Clock) signal. 1 = LCDCLS signal enabled 0 = LCDCLS signal disabled
0	SPSEN	**LCDSPS Signal** Controls the Tristate Enable signal LCDSPS. STN or TFT (Bypass) modes: Reading returns 0. Write the Reset value. Active mode: Enables or disables the generation of the LCDSPS (Row Reset) signal. 1 = LCDSPS signal is enabled 0 = LCDSPS signal is disabled

13.5.5.3 Advanced LCD Interface Timing1 Register

The ALITiming1 Register is used for various delays values for output signals. All delays are specified in number of LCD clock (LCDDCLK) periods. The active bits used in this register are Read/Write.

This register was formerly known as HRTFTCTiming1.

Table 13-40. ALITiming1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///		MODDEL		PSDEL/CLSDEL				REVDEL				LPDEL			
RESET	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE4000 + 0x008															

Table 13-41. ALITiming1 Register Definitions

BITS	NAME	FUNCTION
31:14	///	**Reserved** Reading returns 0. Write the Reset value.
13:12	MODDEL	**LCDMOD LOW Delay** Controls the delay (number of LCDSPS rising edges) to hold LCDMOD LOW before transitioning HIGH. MODDEL = (LCDSPS rising edges – 1). Range is from 1 to 4.
11:8	PSDEL/CLSDEL	**CLCD-to-LCDPS Delay** Controls the delay (number of LCDDCLK periods) from the first detected LOW in horizontal sync from the CLCD to the falling edge of LCDPS and the rising edge of LCDCLS. PSDEL/CLSDEL = (LCDDCLK periods – 1). Range is from 3 to 16.
7:4	REVDEL	**CLCD-to-LCDREV Delay** Controls the delay (number of LCDDCLK periods) from the first detected LOW in the horizontal sync from the CLCD to either edge of the generated LCDREV signal. REVDEL = (LCDDCLK periods – 1). Range is from 3 to 16.
3:0	LPDEL	**CLCD-to-LCDLP Delay** Controls the delay (number of LCDDCLK periods) from the first detected LOW in the horizontal sync from the CLCD to the rising edge of the generated LCDLP. REVDEL = (LCDDCLK periods – 1). Range is from 3 to 16.

NOTE: *The LCDREV signal generated by the ALI is intended for input to a grayscaler ASIC associated with an AD-TFT or HR-TFT display. In this application, it acts as a type of AC Bias signal, switching at a horizontal-line rate, synchronized to the LCDDCLK signal. The LCDREV signal is not intended to reverse the panel's direction-of-scan. Panels utilizing the LCDREV signal must be programmed to utilize an 'odd' number of horizontal display lines. Also, if the panel is programmed to display an 'even' number of horizontal lines, then the net AC Bias applied to each line of the panel will not average to 0 VDC and the panel can suffer long-term damage.

13.5.5.4 Advanced LCD Interface Timing2 Register

The ALITiming2 Register is used for various delay values for output signals. All delays are specified in number of LCD clock (LCDDCLK) periods. The active bits used in this register are Read/Write.

This register was formerly known as HRTFTCTiming2.

Table 13-42. ALITiming2 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SPLVALUE							PSDEL2/CLSDEL2								
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE4000 + 0x00C															

Table 13-43. ALITiming2 Register Definitions

BITS	NAME	FUNCTION
31:16	///	**Reserved** Reading returns 0. Write the Reset value.
15:9	SPLVALUE	**SPL Pulse Delay** Delays SPL pulse during vertical front and back porches. The delay must be programmed to a value greater than HSW + HBP. Note that SPLVALUE can be programmed to a value equal to (HSW + HBP), even though such a value will be illegal. SPLVALUE = (LCDDCLK periods) – 1. Range is from: (HSW + HBP) to: 2 × (HSW + HBP) + HFP.
8:0	PSDEL2/ CLSDEL2	**SPL-to-CLS/PS Delay** Controls the delay (number of LCDDCLK periods) from the rising edge of SPL to the falling/rising edge of CLS/PS, respectively. PSDEL2CLSDEL2 = (LCDDCLK periods) – 1. Range is from 3 to 512 cycles.

13.6 Timing Waveforms

This section describes typical output waveform diagrams for the CLCDC and the ALI.

13.6.1 STN Horizontal Timing

Figure 13-6 shows typical horizontal timing waveforms for STN panels. In this figure, the CLCDC Clock (an input to the CLCDC) is scaled within the CLCDC and used to produce the LCDDCLK output. Programmable registers in the CLCDC set the timings (in terms of LCDDCLK pulses) to produce the other signals that control an STN display.

For example, Figure 13-6 shows that the duration of the LCDLP signal is controlled by Timing0:HSW (the HSW bit field in the Timing0 Register). Figure 13-6 also shows that the polarity of the LCDLP signal is set by Timing2:IHS.

13.6.2 STN Vertical Timing

Figure 13-7 shows typical vertical timing waveforms for STN panels.

13.6.3 TFT Horizontal Timing

Figure 13-8 shows typical horizontal timing waveforms for TFT panels.

13.6.4 TFT Vertical Timing

Figure 13-9 shows typical vertical timing waveforms for TFT panels.

13.6.5 AD-TFT/HR-TFT Horizontal Timing Waveforms

Figure 13-10 shows typical horizontal timing waveforms for AD-TFT and HR-TFT panels. The ALI adjusts the normal TFT timing to accommodate these panels.

13.6.6 AD-TFT/HR-TFT Vertical Timing Waveforms

Figure 13-11 shows typical vertical timing waveforms for AD-TFT and HR-TFT panels. The power sequencing and register information is the same as for TFT vertical timing..

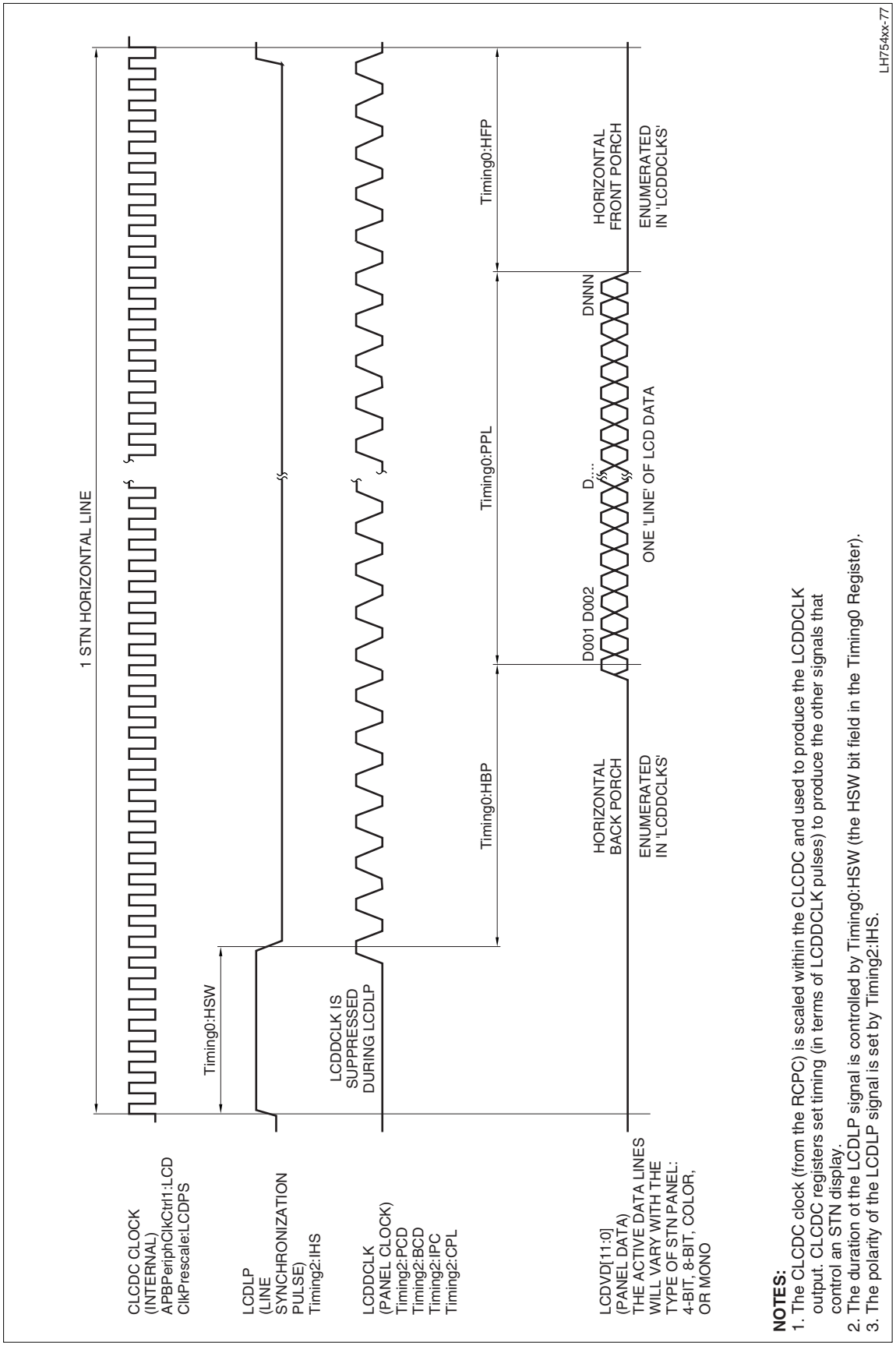


Figure 13-6. STN Horizontal Timing Diagram

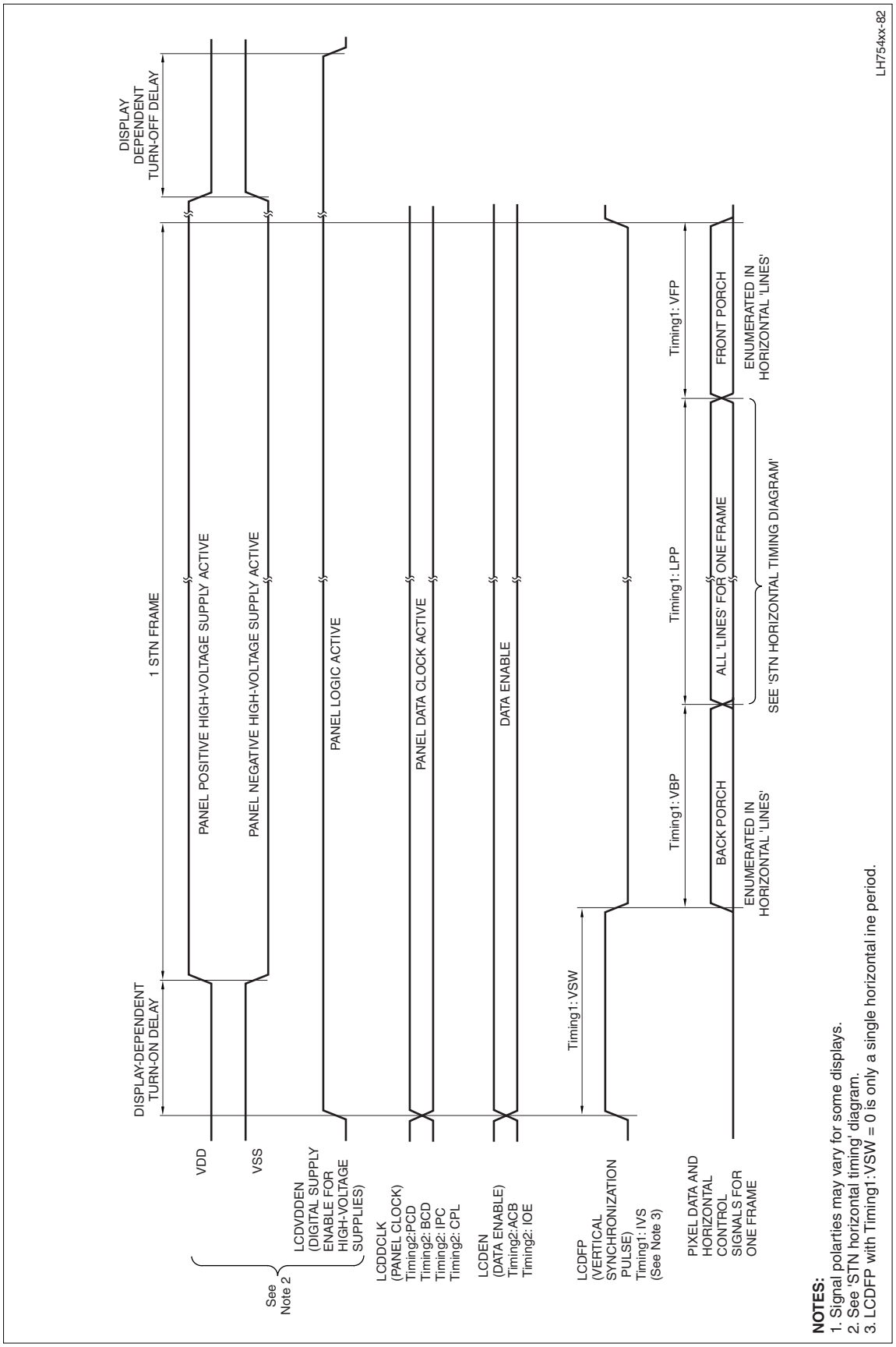
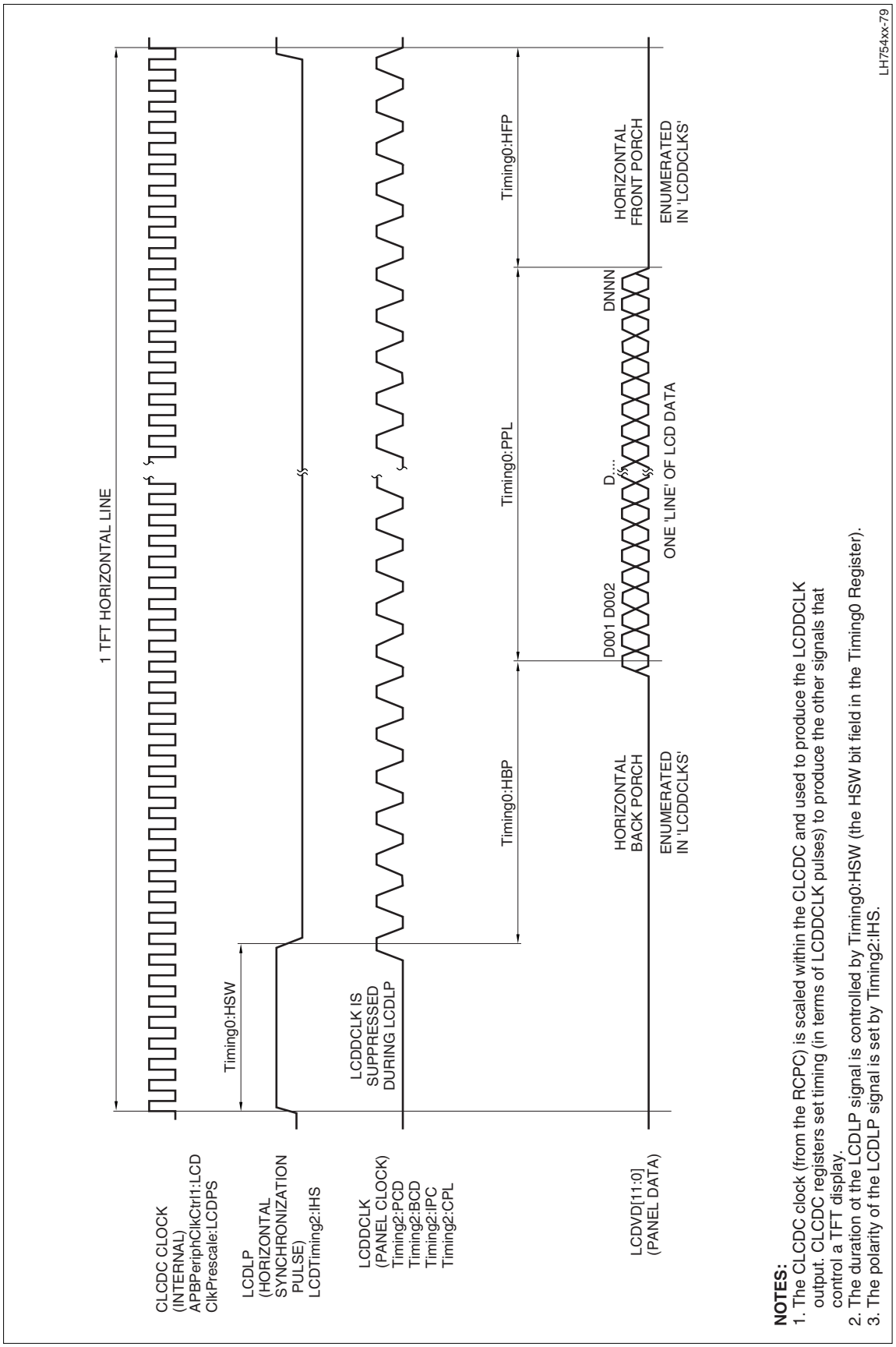


Figure 13-7. STN Vertical Timing Diagram



NOTES:
1. The CLCDC clock (from the RCPC) is scaled within the CLCDC and used to produce the LCDDCLK output. CLCDC registers set timing (in terms of LCDDCLK pulses) to produce the other signals that control a TFT display.
2. The duration ot the LCDLP signal is controlled by Timing0:HSW (the HSW bit field in the Timing0 Register).
3. The polarity of the LCDLP signal is set by Timing2:IHS.

Figure 13-8. TFT Horizontal Timing Diagram

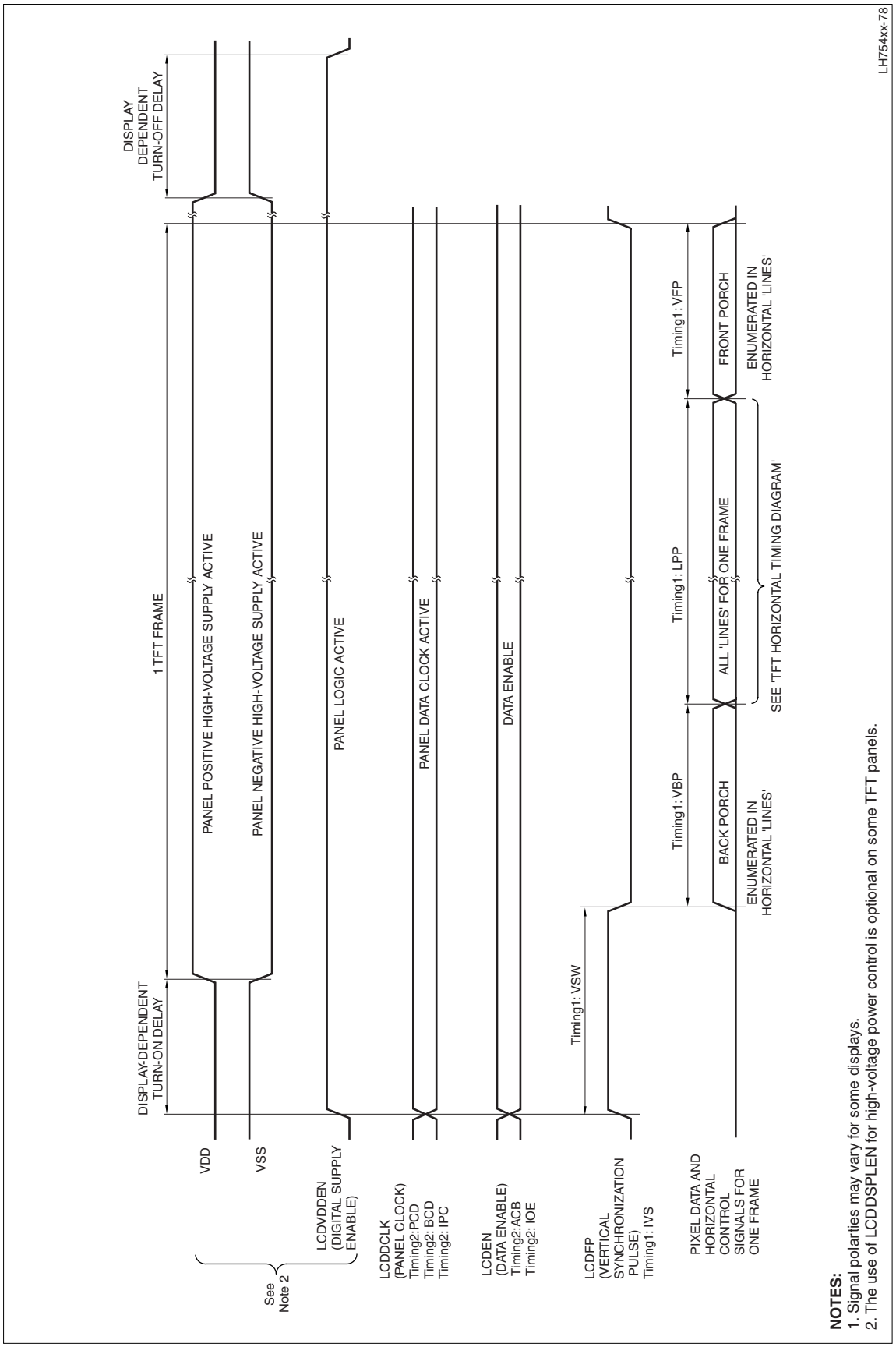


Figure 13-9. TFT Vertical Timing Diagram

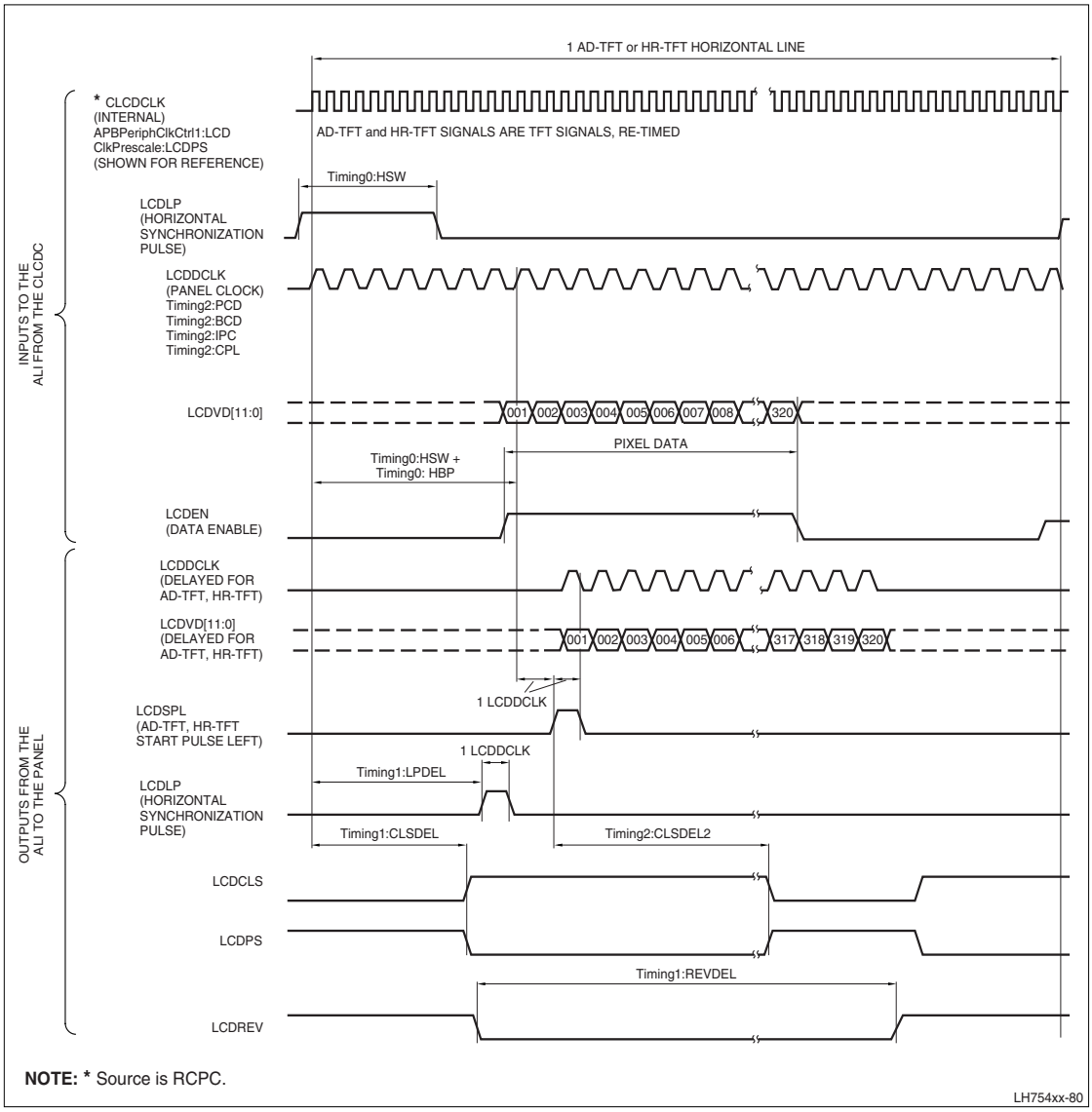


Figure 13-10. AD-TFT, HR-TFT Horizontal Timing Diagram

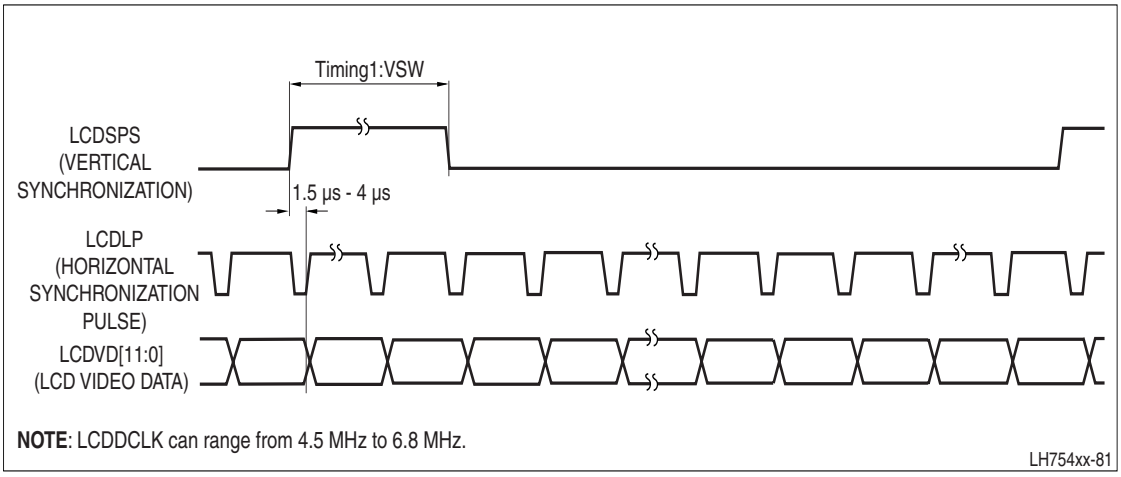


Figure 13-11. AD-TFT, HR-TFT Vertical Timing Diagram

Chapter 14
Liquid Crystal Display Controller

The Liquid Crystal Display (LCD) Controller information in this section pertains to the LH75400 and LH75410 SoC devices only.

The LCD Controller (LCDC) is an AMBA master-slave module that connects to the AHB. The LCDC translates pixel-coded data into the required formats and timings to drive single/dual monochrome LCD panels. The LCDC supports STN LCD displays.

Packets of pixel-coded data are fed, via the AHB interface, to two independently programmable, 32-bit-wide DMA FIFOs. Each FIFO is 16 words deep by 32 bits wide. In single-panel STN Mode, the LCD DMA FIFOs are made to appear as a single FIFO of twice the size. The buffered pixel-coded data is then unpacked via a pixel serializer. Depending on the LCD type and mode, the unpacked data can represent either an actual true display value or an address to a 256 × 16-bit-wide palette RAM value. For STN displays, this value is passed to the gray-scaling generator.

The LCDC generates a single combined interrupt to the VIC when any of the individual interrupt conditions becomes true for upper/lower panel DMA FIFO underflow, base address update signification, vertical compare, or bus error.

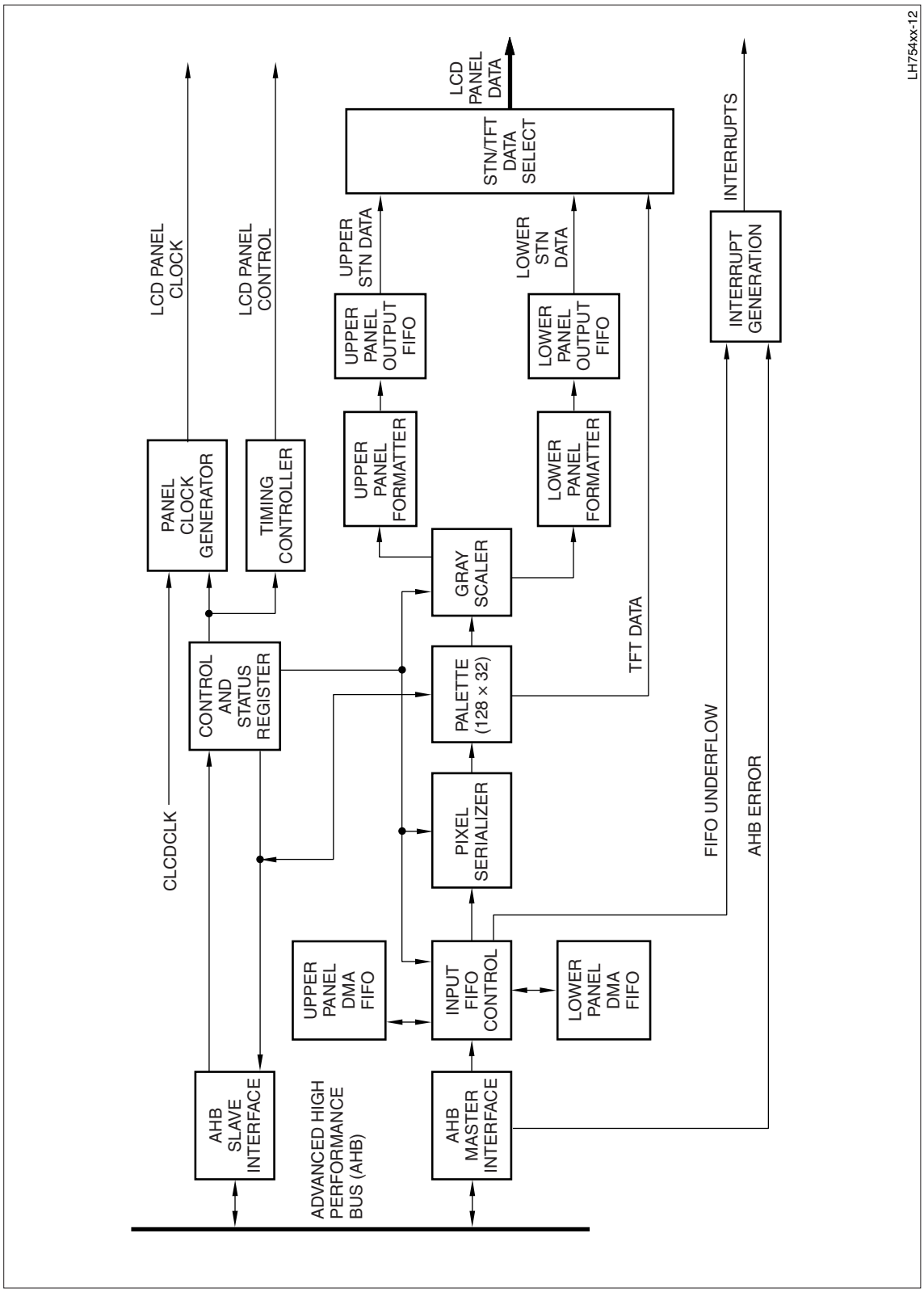


Figure 14-1. LCD Controller Block Diagram (LH75400 and LH75410 Only)

14.1 LCDC Features

The LCDC supports:

- Supported Monochrome STN Panels
  - Single-panel monochrome STN panels (4-bit and 8-bit bus interfaces)
  - Dual-panel monochrome STN panels (4-bit bus interface per panel)
- Supported Resolutions
  - Up to 640 × 480 DPI
- Supported Data Format
  - Little Endian
- Additional Features
  - Programmable timing for different display panels
  - 256-entry, 16-bit palette RAM physically arranged as a 128 × 32-bit RAM
  - AC bias signal for STN panels
- Programmable Parameters
  - Horizontal
    - Horizontal Front Porch (HFP)
    - Horizontal Back Porch (HBP)
    - Horizontal Synchronization Pulse Width (HSW)
    - Number of Pixels per Line (PPL)
  - Vertical
    - Vertical Front Porch (VFP)
    - Vertical Back Porch (VBP)
    - Vertical Synchronization Pulse Width (VSW)
    - Number of Lines per Panel (LPP)
  - Panel-related Parameters
    - Display type: STN mono
    - Bits-per-pixel
    - STN 4-bit Interface Mode
    - STN Single Panel Mode
    - AC panel bias
    - Panel clock frequency
    - Number of panel clocks per line
    - Signal polarity, active HIGH or LOW
    - Little Endian data format
    - Interrupt-generation event

14.2 LCDC Theory of Operation

The LCDC fetches data from Static Memory. The AHB master interface, which is connected directly to the AHB system bus, transfers display data from the memory to the LCD DMA FIFOs. The AHB master interface loads the upper panel base address into the AHB address incrementer when a new frame is recognized. It monitors both the upper and lower LCD DMA FIFO levels, and asserts HBUSREQM to request display data from memory, filling them to a level above the programmed watermark.

14.2.1 LCD DMA FIFOs

The upper and lower LCD DMA FIFOs can be independently controlled to cover single- and dual-panel LCD types. Each FIFO is 16 words deep by 32 bits wide. In single-panel STN Mode, the LCD DMA FIFOs are made to appear as a single FIFO of twice the size.

Synchronization logic is used to transfer the pixel data from the AHB system clock domain, which controls the LCD DMA FIFO, to the LCDDCLK domain. The LCDDCLK is a separate clock provided by the RCPC. The water level marks within each FIFO are set such that each FIFO requests data when at least four locations become available. An interrupt signal is asserted if an attempt is made to read either of the two LCD DMA FIFOs when they are empty.

14.2.2 Pixel Serializer

The pixel serializer reads the 32-bit-wide LCD data from the output port of the LCD DMA FIFO and extracts 4, 2, or 1 bpp data, depending on the operating mode. In Dual Panel Mode, data is alternately read from the upper and lower LCD DMA FIFOs. Depending on the operating mode, the extracted data is used to point to a grayscale value in the palette RAM or is directly applied to a LCD panel input.

14.2.3 How Pixels are Stored in Memory

Table 14-1 and Table 14-2 show the data structure in each DMA FIFO word corresponding to the bpp combinations; where a pixel p(x) is encoded in the bit field for pixels displayed in the order (x = 0, 1, 2, 3…). The required data for each panel display pixel must be extracted from the data word.

Table 14-1. DMA FIFO Output Bits [31:16]

bpp	DMA FIFO OUTPUT BITS															
	31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
1	p31	p30	p29	p28	p27	p26	p25	p24	p23	p22	p21	p20	p19	p18	p17	p16
2	p15		p14		p13		p12		p11		p10		p9		p8	
	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4	p7				p6				p5				p4			
	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0

Table 14-2. DMA FIFO Output Bits [15:0]

bpp	DMA FIFO OUTPUT BITS															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	p15	p14	p13	p12	p11	p10	p9	p8	p7	p6	p5	p4	p3	p2	p1	p0
2	p7		p6		p5		p4		p3		p2		p1		p0	
	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4	p3				p2				p1				p0			
	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0

14.2.4 Palette RAM

The LCDC includes a 256 × 16-bit dual-port RAM-based palette. The least-significant bit of the serialized pixel data is used to select between the upper and lower halves of the palette RAM, based on the Byte Ordering Mode.

- One port of the dual-port palette RAM is used as a Read/Write port and is connected to the AHB slave interface. Palette entries can be written and verified through this port.
- The second port of the dual-port palette RAM is used as a Read Only port and is connected to the unpacker and grayscaler.

Table 14-3 shows the bit representation of each word in the palette.

Table 14-3. Palette Data Storage

BIT	NAME	DESCRIPTION
31:21	///	Reserved
20:17	R[3:0]	Grayscale data
16:5	///	Reserved
4:1	R[3:0]	Grayscale data
0	///	Reserved

14.2.5 Grayscale Algorithm

A patented grayscale algorithm drives monochrome STN panels, providing 15 grayscales. The grayscaler transforms each 4-bit gray value into a sequence of activity-per-pixel over several frames, relying somewhat on the display characteristics, to give the representation of grayscales. See Table 14-4.

Data bit values from the grayscaler are shifted into the register in the upper and lower panel formatter. Each upper and lower panel formatter consists of a Shift Left Register. When enough data is available, a byte is constructed by multiplexing the registered data to the correct bit position to satisfy the data pattern of the LCD panel. The byte is transferred to the FIFO, which has space to store eight pixels.

The LCDC has four individually maskable interrupt conditions going to a single combined interrupt. The single combined interrupt is asserted if any of the combined interrupt conditions are asserted and unmasked.

Table 14-4. STN Intensities From Grayscale Modulation

4-BIT PALETTE VALUE	DUTY CYCLE[1]	RESULTING INTENSITY[2]
0b0000	0/90	00.0%
0b0001	10/90	11.1%
0b0010	18/90	20.0%
0b0011	24/90	26.7%
0b0100	30/90	33.3%
0b0101	36/90	40.0%
0b0110	40/90	44.4%
0b0111	45/90	50.0%
0b1000	45/90	50.0%
0b1001	50/90	55.6%
0b1010	54/90	60.0%
0b1011	60/90	66.6%
0b1100	66/90	73.3%
0b1101	72/90	80.0%
0b1110	80/90	88.9%
0b1111	90/90	100.0%

NOTES:
1. Duty cycle is determined by (pixels on ÷ (pixels on + pixels off)).
2. Resulting intensity: 000% = black, 100% = white.

14.2.6 Supported Grayscale

Table 14-5 shows the number of LCDC supported grayscales.

Table 14-5. LCD Controller Grayscale Support

bpp	MONO STN		NOTES
1 bpp	Palletized	2 shades	1
2 bpp	Palletized	4 shades	1
4 bpp	Palletized	16 shades	1
8 bpp	///	///	2
12 bpp	///	///	2

NOTES:
1. Grayscales selected from 15 grayscales.
2. Greater than four bpp can be programmed; however, using these modes does not make sense, since the maximum number of grayscales supported on this display is 15.

14.3 LCDC Programmer's Model

The base address for the LCDC is:

LDC Base Address: 0xFFFF4000

The following locations are reserved and must not be used during normal operation:

- Locations at offsets 0x030 through 0x1FC
- Locations at offsets 0x400 through 0x7FF.

14.3.1 LCDC Register Summary

Table 14-6. LCDC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Timing0	0x000	RW	0x00000000	Horizontal Timing Panel Control Register
Timing1	0x004	RW	0x00000000	Vertical Timing Panel Control Register
Timing2	0x008	RW	0x0000000	Clock and Signal Polarity Control Register
///	0x00C	RW		Reserved
UPBASE	0x010	RW	0x0000000	Upper Panel Frame Buffer Base Address Register
LPBASE	0x014	RW	0x00000000	Lower Panel Frame Buffer Base Address Register
INTRENABLE	0x018	RW	0x00000000	Interrupt Enable Register
CTRL	0x01C	RW	0x0000	LCD Panel Parameters, LCD Panel Power, and LCDC Control Register
Status	0x020	RW	0x00000000	Raw Interrupt Status Register
Interrupt	0x024	R	0x00000000	Final Masked Interrupts Register
UPCURR	0x028	R	0x00000000	Upper Panel Frame Buffer Current Address Register
LPCURR	0x02C	R	0x00000000	Lower Panel Frame Buffer Current Address Register
///	0x030 - 0x1FC		0x00000	Reserved
Palette	0x200 - 0x3FC	RW		LCD Palette Register. Palette is addressed at 32 bits
///	0x400 - 0x7FF			Reserved

14.3.2 LCDC Register Definitions

14.3.2.1 Horizontal Timing Panel Control Register

The Timing0 Register controls:

- Horizontal Synchronization Pulse Width (HSW)
- Horizontal Front Porch (HFP) period
- Horizontal Back Porch (HBP) period
- Pixels-Per-Line (PPL).

Table 14-7. Timing0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	HBP								HFP							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	HSW								PPL						///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
ADDR	0xFFFF4000 + 0x00															

Table 14-8. Timing0 Register Definitions

BIT	NAME	DESCRIPTION
31:24	HBP	**Horizontal Back Porch** Specifies the number of LCDDCLK periods between the falling edge of LCDLP and the start of active data; that is, the number of pixel clock periods inserted at the beginning of each line or row of pixels. Program with (value – 1). After the line clock for the previous line has been de-asserted, the value in HBP counts the number of pixel clocks to wait before starting the next display line. HBP can generate a delay of 1 to 256 pixel clock cycles.
23:16	HFP	**Horizontal Front Porch** Specifies the number of LCDDCLK periods between the end of active data and the rising edge of LCDLP; that is, the number of pixel clock intervals at the end of each line or row of pixels, before the LCD line clock is pulsed. Program with (value – 1). Once a complete line of pixels is transmitted to the LCD driver, the value in HFP counts the number of pixel clocks to wait before asserting the line clock. HFP can generate a period of 1 to 256 pixel clock cycles.
15:8	HSW	**Horizontal Synchronization Pulse Width** Specifies the width of the LCDLP signal in LCDDCLK periods; that is, the pulse width of the line clock in Passive Mode, or the horizontal synchronization pulse in Active Mode. Program with (value – 1).
7:2	PPL	**Pixels-Per-Line** Specifies the number of pixels, between 16 and 1,024, in each line or row of the screen. PPL counts the number of pixel clocks that occur before the HFP is applied (program the value required divided by 16, minus 1).
1:0	///	**Reserved** Write the reset value.

14.3.2.2 Horizontal Timing Restrictions

The LCD DMA requests new data at the start of a horizontal display line. Some time must be allowed for the DMA transfer and for the data to propagate down the FIFO path in the LCD interface. The data path latency forces some restrictions on the usable minimum values for horizontal porch width in STN Mode.

The minimum values are HSW = 2 and HBP = 2.

Single Panel Mode:

- HSW = 3
- HBP = 5
- HFP = 5
- Panel Clock Divisor (PCD) = 1 (LCDCLK/3).

Dual Panel Mode:

- HSW = 3
- HBP = 5
- HFP = 5
- PCD = 5 (LCDCLK/7).

If sufficient time is given at the start of the line (for example, setting HSW = 6, HBP = 10), data will not get corrupted for PCD = 4 (minimum value).

14.3.2.3 Vertical Timing Panel Control Register

The Timing1 Register controls the:

- Number of Lines-Per-Panel (LPP)
- Vertical Synchronization Pulse Width (VSW)
- Vertical Front Porch (VFP) period
- Vertical Back Porch (VBP) period

Table 14-9. Timing1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	VBP								VFP							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	VSW						LPP									
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x04															

Table 14-10. Timing1 Register Definitions

BIT	NAME	DESCRIPTION
31:24	VBP	**Vertical Back Porch** Specifies the number of inactive lines at the start of a frame, after vertical synchronization period; that is, the number of line clocks inserted at the beginning of each frame. Program to zero on passive displays; otherwise, reduced contrast will result. The VBP count starts just after the Vertical Synchronization signal for the previous frame has been negated for Active Mode, or the extra line clocks have been inserted as specified by the VSW bit field in Passive Mode. After this has occurred, the count value in VBP sets the number of line clock periods inserted before the next frame. VBP generates from 0 – 255 extra line clock cycles.
23:16	VFP	**Vertical Front Porch** Specifies the number of inactive lines at the end of frame, before vertical synchronization period; that is, the number of line clocks to insert at the end of each frame. Program to zero on passive displays; otherwise, reduced contrast will result. Once a complete frame of pixels is transmitted to the LCD display, the value in VFP counts the number of line clock periods to wait. After the count has elapsed, the Vertical Synchronization signal is asserted in Active Mode, or extra line clocks are inserted as specified by the VSW bit field in Passive Mode. VFP generates from 0 – 255 line clock cycles.
15:10	VSW	**Vertical Synchronization Pulse Width** Specifies the number of horizontal synchronization lines; that is, the pulse width of the vertical synchronization pulse. Should be small (for example, program to zero) for passive STN LCDs. Program to the number of lines required minus one. The higher the value, the worse the contrast on STN LCDs. The register is programmed with the number of line clocks in Vsync minus one. Number of horizontal synchronization lines. Should be small (for example, program to zero) for passive STN LCDs. Program to the number of lines required minus one.
9:0	LPP	**Lines Per Panel** Specifies the number of active lines per screen. Program to number of lines required minus 1; that is, the total number of lines or rows on the LCD panel being controlled. LPP is a 10-bit value allowing between 1 and 1,024 lines. The register is programmed with the number of lines per LCD panel minus 1. For dual-panel displays this register is programmed with the number of lines on each of the upper and lower panels.

14.3.2.4 Clock and Signal Polarity Control Register

The Timing2 Register controls the LCDC timing.

Table 14-11. Timing2 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///					BCD	CPL									
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	IOE	IPC	HIS	IVS	ACB					///	PCD				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x08															

Table 14-12. Timing2 Register Definitions

BIT	NAME	DESCRIPTION
31:27	///	**Reserved** Write the reset value.
26	BCD	**Bypass Pixel Clock Divider** 1 = Bypasses the pixel clock divider logic. Always set to 0 for electroluminescent (EL) and STN displays.
25:16	CPL	**Clocks Per Line** Specifies the number of actual LCDDCLK clocks to the LCD panel on each line. This is the number of pixels per line divided by either 4 or 8, minus one. This must be correctly programmed in addition to PPL for the LCD Controller to work correctly.
15	///	**Reserved** Write the reset value.
14	IOE	**Invert Output Enable** Selects the active polarity of the output enable signal. In this mode, the LCDEN pin is used as an enable that indicates to the LCD panel when valid display data is available. In Active Display Mode, data is driven onto the LCD data lines at the programmed edge of LCDDCLK when LCDEN is in its active state. 0 = LCDEN output pin is active HIGH. 1 = LCDEN output pin is active LOW.
13	IPC	**Invert Panel Clock** Selects the edge of the panel clock on which pixel data is driven out onto the LCD data lines. 0 = Data is driven on the LCDs data lines on the rising-edge of LCDDCLK. 1 = Data is driven on the LCDs data lines on the falling-edge of LCDDCLK.
12	HIS	**Invert Horizontal Synchronization** Inverts the polarity of the LCDLP signal. 0 = LCDLP pin is active HIGH and inactive LOW. 1 = LCDLP pin is active LOW and inactive HIGH.
11	IVS	**Invert Vertical Synchronization** Inverts the polarity of the CLFP signal. 0 = CLFP pin is active HIGH and inactive LOW. 1 = CLFP pin is active LOW and inactive HIGH.
10:6	ACB	**AC Bias Pin Frequency** The AC bias pin frequency is only applicable to STN displays, which require the pixel voltage polarity to be periodically reversed to prevent damage due to DC charge accumulation. Program this field with the required value minus one to apply the number of line clocks between each toggle of the AC bias pin (LCDEN).
5	///	**Reserved** Write the reset value.
4:0	PCD	**Panel Clock Divisor** Derives the LCD panel clock frequency LCDDCLK from the input LCDC clock frequency, according to the formula LCDDCLK = LCDC CLOCK/(PCD + 2). For monochrome STN displays with a 4- or 8-bit interface, the panel clock will be a factor of four and eight down on the actual individual pixel clock rate.

14.3.2.5 Upper Panel Frame Buffer Base Address Register

The UPBASE Register is one of two LCD DMA Base Address Registers (the other is LPBASE). Together with LPBASE, this Read/Write register programs the base address of the frame buffer.

UPBase is used for:

- Single-panel STN displays
- The upper panel of dual-panel STN displays.

UPBase (and LPBase for dual panels) must be initialized before enabling the LCDC. Optionally the value can be changed mid-frame to allow double-buffered video displays to be created. These registers are copied to the corresponding current registers at each LCD vertical synchronization. This event causes the LNBU bit and an optional interrupt to be generated. The LNBU bit indicates that it is safe to update both the UPBASE and LPBASE Registers. The interrupt can be used to reprogram the base address when generating double-buffered video.

Table 14-13. UPBASE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDUPBASE															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDUPBASE														///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
ADDR	0xFFFF4000 + 0x10															

Table 14-14. UPBASE Register Definitions

BIT	NAME	DESCRIPTION
31:2	LCDUPBASE	**LCD Upper Panel Base Address** Specifies the starting address of the upper-panel frame data in memory and is word aligned.
1:0	///	**Reserved** Write the reset value.

14.3.2.6 Lower Panel Frame Buffer Base Address Register

The LPBASE Register is one of two LCD DMA Base Address Registers (the other is UPBASE). Together with UPBASE, this Read/Write register programs the base address of the frame buffer.

LPBase is used for the lower panel of dual-panel STN displays. UPBase must be initialized (and LPBase for dual panels) before enabling the LCDC. Optionally the value can be changed mid-frame to allow double-buffered video displays to be created. These registers are copied to the corresponding current registers at each LCD vertical synchronization. This event causes the LNBU bit and an optional interrupt to be generated. The LNBU bit indicates that it is safe to update both the UPBASE and LPBASE Registers. The interrupt can be used to reprogram the base address when generating double-buffered video.

Table 14-15. LPBASE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDLPBASE															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDLPBASE														///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
ADDR	0xFFFF4000 + 0x14															

Table 14-16. LPBASE Register Definitions

BIT	NAME	DESCRIPTION
31:2	LCDLPBASE	**LCD Lower Panel Base Address** Specifies the starting address of the lower panel frame data in memory and is word aligned.
1:0	///	**Reserved** Write the reset value.

14.3.2.7 Interrupt Enable Register

INTRENABLE is the Interrupt Enable Register. Setting bits within this register enables the corresponding raw interrupt Status bit values to be passed to the Interrupt Register.

Table 14-17. INTRENABLE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											MBERRINTRENB	VCOMPINTRENB	LNBUINTRENB	FUFINTRENB	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	R
ADDR	0xFFFF4000 + 0x18															

Table 14-18. INTRENABLE Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	MBERRINTRENB	**AHB Master Error Interrupt Enable** 1 = Enables the AHB Master Error Interrupt to be passed to the Raw Interrupt Status Register.
3	VCOMPINTRENB	**Vertical Compare Interrupt Enable** 1 = Enables the Vector Compare Interrupt to be passed to the Raw Interrupt Status Register.
2	LNBUINTRENB	**Next Base Update Interrupt Enable** 1 = Enables the Next Base Update Interrupt to be passed to the Raw Interrupt Status Register.
1	FUFINTRENB	**FIFO Underflow Interrupt Enable** 1 = Enables the FIFO Underflow Interrupt to be passed to the Raw Interrupt Status Register.
0	///	**Reserved** Write the reset value.

14.3.2.8 LCD Panel Parameters, LCD Panel Power, and LCDC Control Register

CTRL is a Read/Write register that controls the LCDC operating mode.

Table 14-19. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															WATERMARK
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LDmaFIFOTME	///	LcdVComp		LcdPwr	///		///	LcdDual	LcdMono8	///	///	LcdBpp			LcdEn
RESET	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x1C															

Table 14-20. CTRL Register Definitions

BIT	NAME	DESCRIPTION
31:17	///	**Reserved** Write the reset value.
16	WATERMARK	**LCD DMA FIFO Watermark Level** 0 = HBUSREQM is raised when either of the two LCD DMA FIFOs have four or more empty locations. 1 = HBUSREQM is raised when either of the LCD DMA FIFOs have eight or more empty locations.
15	LDmaFIFOTME	**LCD DMA FIFO Test Mode Enable** 0 = LCD DMA FIFO inaccessible to user. 1 = LCD DMA FIFO Read/Write access for FIFO RAM testing. Set this bit only when LCD is disabled via bit [0] of this register.
14	///	**Reserved** Write the reset value.
13:12	LcdVComp	**Generate Interrupt** 00 = At start of vertical synchronization 01 = At start of back porch 10 = At start of active video 11 = At start of front porch
11	LcdPwr	**LCD Power Enable** This bit causes the LCDEN pin to toggle. 0 = LCD is OFF, LCDEN pin is LOW. 1 = LCD is ON, LCDEN pin is HIGH if bit [0] of this register is HIGH. When using this setting, set bit [0] to 1.
10:8	///	**Reserved** Write the reset value.

Table 14-20. CTRL Register Definitions (Cont'd)

BIT	NAME	DESCRIPTION
7	LcdDual	**LCD Interface is Dual-Panel STN** This bit has no meaning in other modes, program to zero. 0 = Single-panel LCD is in use. 1 = Dual-panel LCD is in use.
6	LcdMono8	**Monochrome LCD 4-/8-bit Interface** This bit controls whether monochrome STN LCD uses a 4- or 8-bit parallel interface. It has no meaning in other modes, program to zero. 0 = Mono LCD uses 4-bit interface. 1 = Mono LCD uses 8-bit interface.
5:4	///	**Reserved** Write the reset value.
3:1	LcdBpp	**LCD Bits per Pixel** 000 = 1 bpp 001 = 2 bpp 010 = 4 bpp 011 = 8 bpp 100 = 12 bpp (16 bits fetched from memory) 101 = Reserved 110 = Reserved 111 = Reserved
0	LcdEn	**LCD Controller Enable** 0 = LCD Controller is disabled. 1 = LCD Controller is enabled. LCD displays require that the LCD is running before power is applied. For this reason, the LCD's power-on control (bit [11]) is not set to 1 unless both LcdEn and LcdPwr are set to 1. Most LCD displays require the LcdEn to be set to 1 approximately 20 ms before LcdPwr is set to 1 for powering up. Likewise, LcdPwr should be set to 0 (20 ms) before LcdEn is set to 0 for powering down. 20 ms is a typical value. Refer to the documentation that came with your panel for the correct time.

14.3.2.9 Raw Interrupt Status Register

Status is the Raw Interrupt Status Register. This register is Read/Write.

- On a read, this register returns five bits that may generate interrupts when set.
- On writes to this register, a bit value of '1' clears the interrupt corresponding to that bit. Writing a '0' has no effect.

Table 14-21. Status Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											MBERROR	Vcomp	LNBU	FUF	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RC	RC	RC	RC	R
ADDR	0xFFFF4000 + 0x20															

Table 14-22. Status Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	MBERROR	**Master Bus Error Interrupt** Asserted when an ERROR response is received by the master interface during a transaction with a slave. When such an error is encountered, the master interface enters an error state and remains in this state until clearance of the error has been signaled to it.
3	VCOMP	**Vertical Compare Interrupt** Asserted when one of four vertical display regions, selected via the LCD Control Register with bits [13:12], is reached. The interrupt can be made to occur at the start of Vertical Synchronization, Back Porch, Active Video, and Front Porch.
2	LNBU	**LCD Next Base Address Update Interrupt** Asserted when either the UPBASE or the LPBASE values have been transferred to the UPCURR or LPCURR incrementer, respectively. This indicates to the system that it can safely update the UPBASE or the LPBASE Register with new frame base addresses if required.
1	FUF	**FIFO Underflow Interrupt** Asserted when internal data is requested from an empty LCD DMA FIFO. Internally, individual upper and lower panel LCD DMA FIFO Underflow Interrupt signals are generated, and this is the single combined version of these.
0	///	**Reserved** Write the reset value.

14.3.2.10 Final Masked Interrupts Register

The Interrupt Register is a Read Only register. It is a bit-by-bit logical AND of the Raw Interrupt Status Register (see Section 14.3.2.9) and INTRENABLE Register (see Section 14.3.2.7). Interrupt lines correspond to each interrupt. A logical OR of all interrupts is provided to the System Interrupt Controller.

Table 14-23. Interrupt Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											MBERRINTRENB	VCOMPINTRENB	LNBUINTRENB	FUFINTRENB	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFF4000 + 0x24															

Table 14-24. Interrupt Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	MBERRORINTR	AHB master error interrupt status bit
3	VCOMPINTR	Vertical compare interrupt status bit
2	LNBUINTR	LCD next base address update interrupt status bit
1	FUFINTR	FIFO underflow interrupt status bit
0	///	**Reserved** Write the reset value.

14.3.2.11 LCD Upper Panel Frame Buffer Current Address Register

UPCURR and LPCURR are registers that contain an approximate value of the upper and lower panel data DMA addresses when read. The registers can change at any time and provide a coarse indication of the current LCD DMA memory pointer.

Table 14-25. UPCURR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDUPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDUPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFF4000 + 0x28															

Table 14-26. UPCURR Register Definitions

BIT	NAME	DESCRIPTION
31:0	LCDUPCURR	**Current Upper Panel Data DMA Address** Contains the approximate current upper panel data DMA address.

Table 14-27. LPCURR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	LCDLPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	LCDLPCURR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFF4000 + 0x2C															

Table 14-28. LCDLPCURR Register Definitions

BIT	NAME	DESCRIPTION
31:0	LCDLPCURR	**Current Lower Panel Data DMA Address** Contains the approximate current lower panel data DMA address.

14.3.2.12 LCD Palette Register

The color map in the SoC is 65,535 colors. These are mapped into a group of palette entries, comprising the Palette Registers. The upper four bits of each palette entry is used for best contrast. The palette is organized as seen in Table 14-29.

Palette Registers contain 256 palette entries organized as 128 locations of two entries per word. STN monochrome displays use 8 of the palette entry bits, yielding 16 grayscale shades. Each word location contains two palette entries.

The LH75400 and LH75410 support only monochrome displays; the register locations and data bits for the Red portions of the palette are used to create and display the monochrome signals. The Blue and Green portions of the palette are reserved. Writing to them will not allow creation of color signals for a color display. Note that the palettes are accessed 32 bits at a time.

Table 14-29. Palette Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///											RED PALETTE				///
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											RED PALETTE				///
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFF4000 + 0x200 to 0xFFFF4000 + 0x3FC															

Table 14-30. Palette Register Definitions

BIT	NAME	DESCRIPTION
31:21	///	Reserved
20:17	R[3:0]	Grayscale data
16:5	///	Reserved
4:1	R[3:0]	Grayscale data
0	///	Reserved

14.3.3 LCDC Interrupts

The single combined interrupt, CLCDINTR, drives the VIC. If any of the four interrupt conditions occurs, this signal is asserted.

Each of the four individual maskable interrupt conditions is enabled or disabled by changing the mask bits in the INTRENABLE Register. Provision of individual outputs, along with a combined interrupt output, allows the use of either a global interrupt service routine or modular device drivers to handle interrupts. The status of the individual interrupt sources can be read from the Status Register.

Chapter 15
Timers

The timer block consists of three 16-bit timers:

- Timer 0 has five Capture Registers and two Compare Registers.
  - Capture Registers: CAP0, CAP1, CAP2, CAP3, and CAP4
  - Compare Registers: CMP0 and CMP1
- Timer 1 has two Capture Registers and two Compare Registers.
  - Capture Registers: CAP0 and CAP1
  - Compare Registers: CMP0 and CMP1
- Timer 2 has two Capture Registers and two Compare Registers.
  - Capture Registers: CAP0 and CAP1
  - Compare Registers: CMP0 and CMP1

The timers are clocked by the system clock, but have an internal scaled-down system clock that is used for the PWM and compare functions.

All counters are incremented by an internal prescaled counter clock or external clock and can generate an overflow interrupt when the counter goes from 0xFFFF to 0x0000. All three timers have separate internal prescaled counter clocks, with either a common external clock (CTCLK) or a prescaled version of the system clock.

The SoCs support nine Capture Registers. The Capture Registers have edge-selectable inputs and can generate an interrupt, if desired.

The SoCs support six Compare Registers. The Compare Registers can force the compare output pin either HIGH or LOW upon a match.

Each timer can generate a separate interrupt. The interrupt becomes active if any enabled compare, capture, or overflow interrupt conditions occurs. The interrupt remains active until all compare, capture, and overflow interrupts are cleared.

Figure 15-1 shows a block diagram that includes all three timers.

Figure 15-2 shows a block diagram of Timer 0.

Figure 15-3 shows a block diagram representing Timer 1 and Timer 2. These two timers are identical.

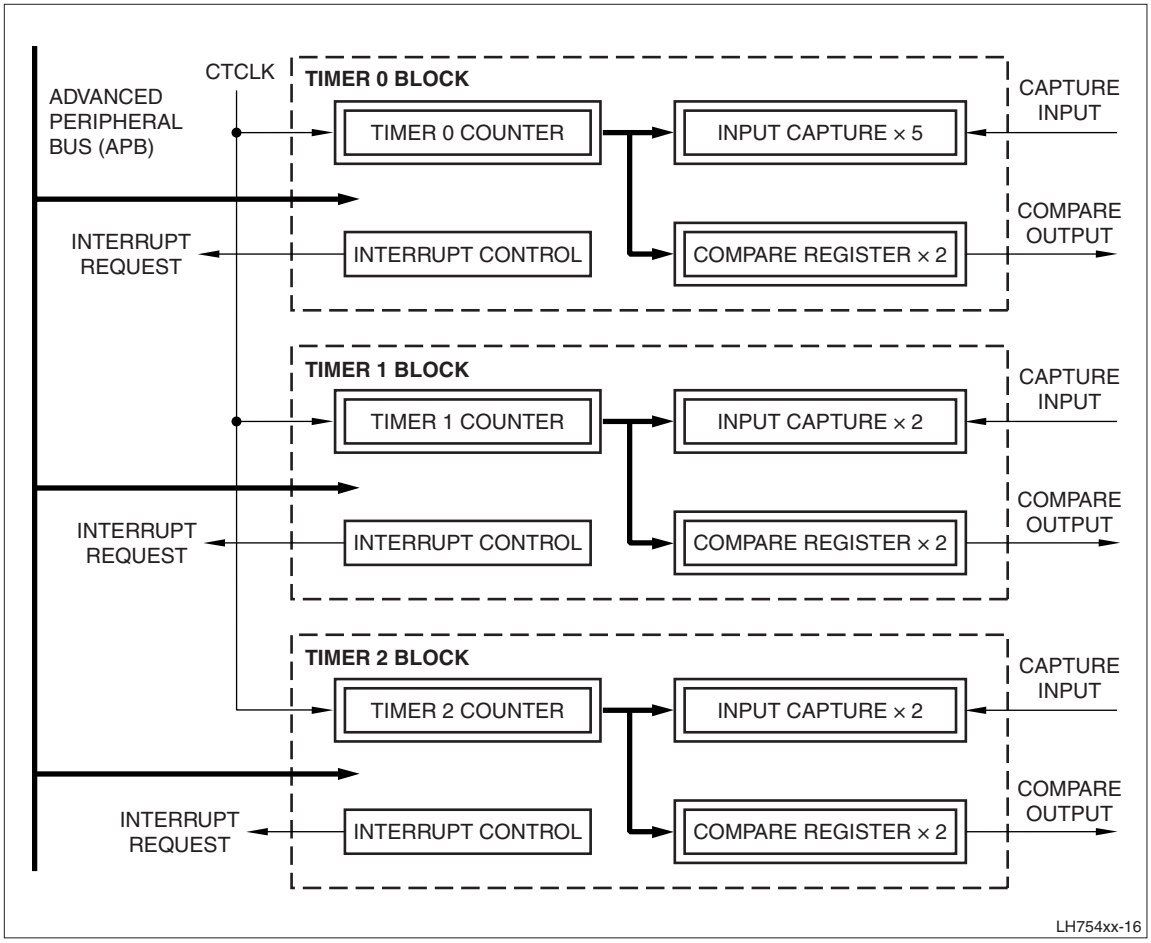


Figure 15-1. Overall Timer Block Diagram

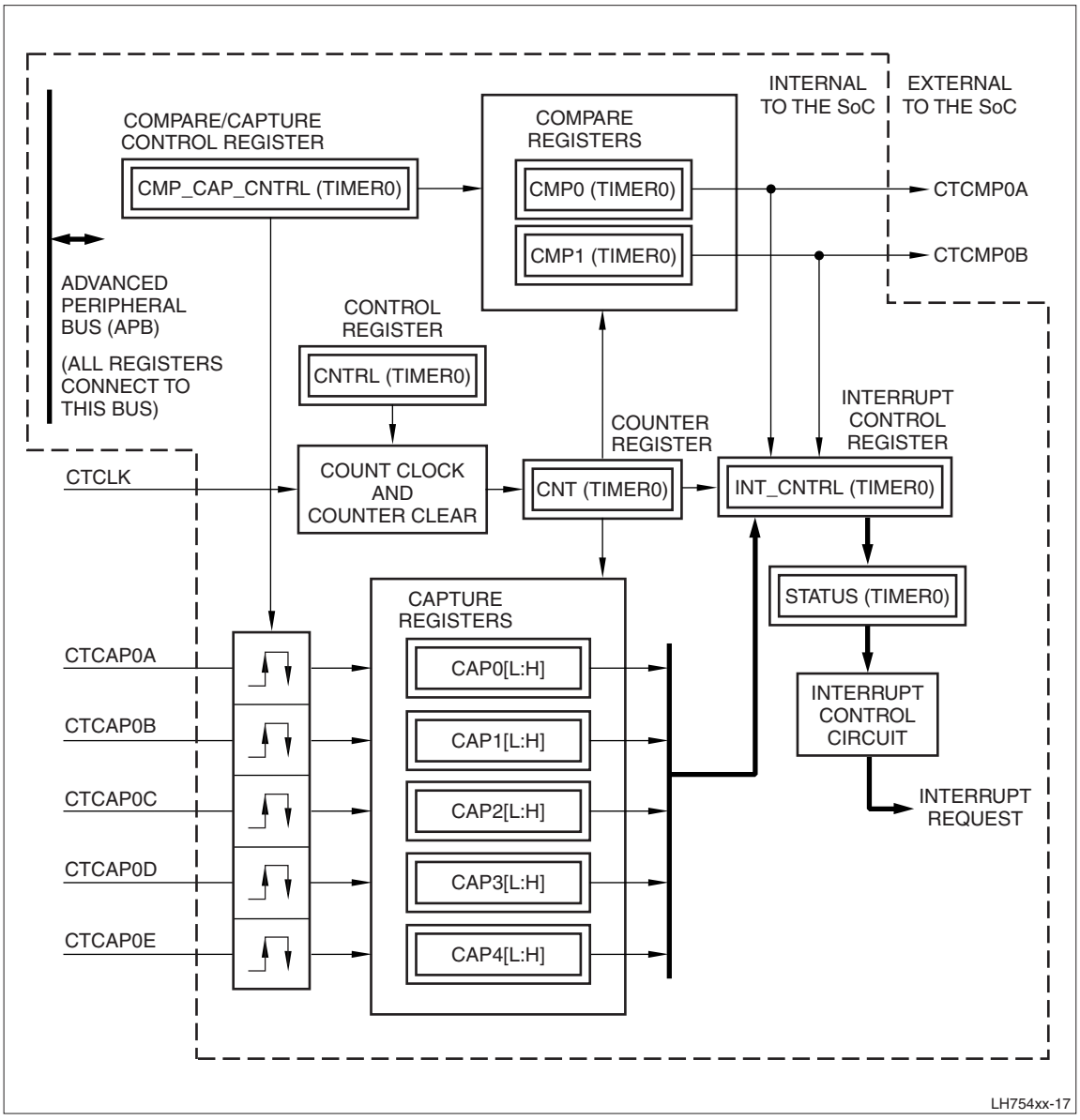


Figure 15-2. Timer 0 Block Diagram

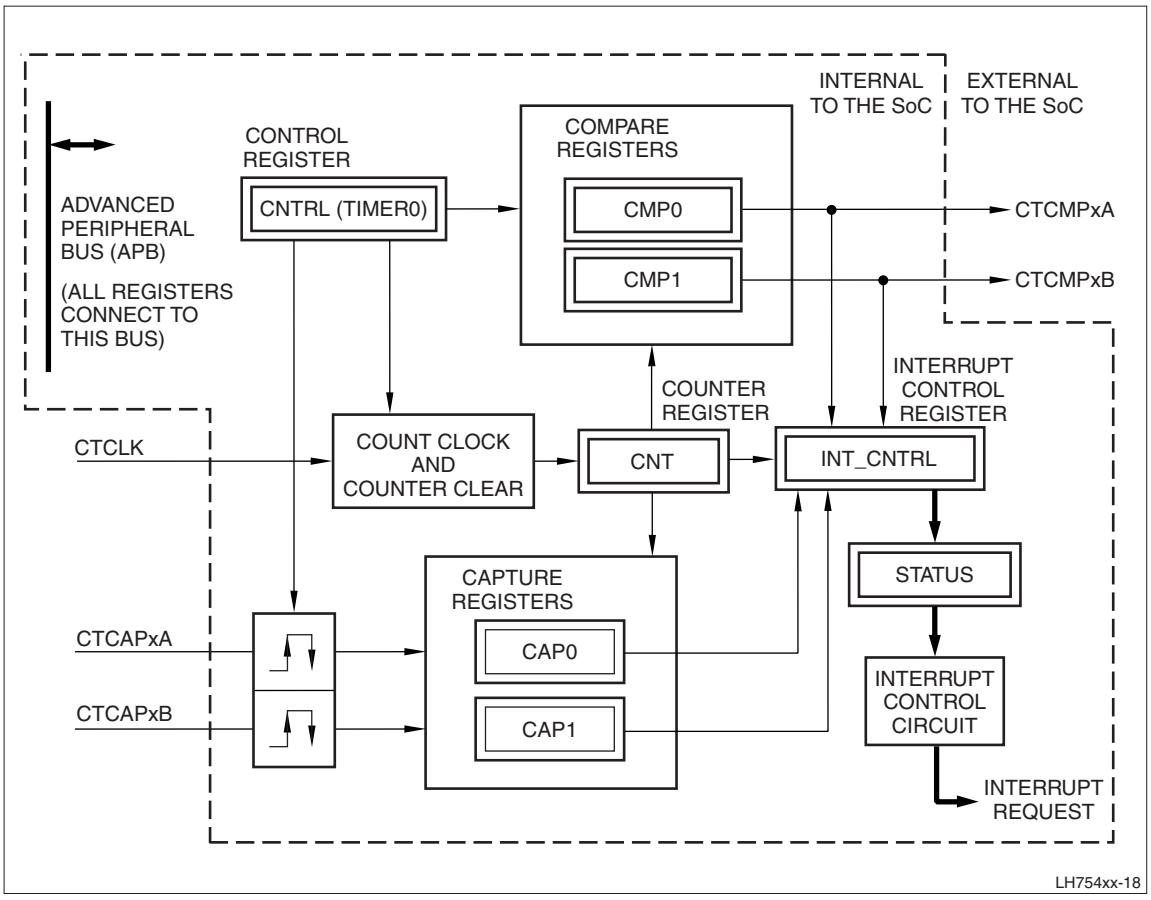


Figure 15-3. Timers 1 and 2 Block Diagram

15.1 Timer Theory of Operation

15.1.1 Count Timing

Each counter can use either one of the supported internal divided-by-n system clocks or an external clock as its count clock. Selection between one of the internal system clocks or the external clock is accomplished using the Timer Control Register that corresponds to the timer that is to be programmed.

The count clock to be applied to the counter can be changed only when the counter is in Stop Mode. If you try to change the count clock while the counter is running, the new count clock is ignored.

To change the count clock:

1. Stop the counter by writing a 0 to the CS bit of the appropriate Timer Control Register.
2. Select a desired clock by writing a value to the SEL bits of the same Timer Control Register accessed in step 1.
3. Start the counter by writing a 1 to the CS bit of the same Timer Control Register accessed in the previous steps.

If you select CTCLK in step 2, the timer increments the counter of the corresponding Timer Control Register on the third rising edge of system clock after a rising edge by CTCLK. The pulse length of CTCLK must be equal to or longer than two system clock periods, plus the setup and hold time. Shorter pulses can cause wrong counts. If CTCLK is not in phase with the system clock, inaccurate counts can occur.

Figure 15-4 shows the timing of CTCLK with respect to system clock when the two are in phase. Figure 15-5 shows the timing of CTCLK with respect to system clock when the two are not in phase.

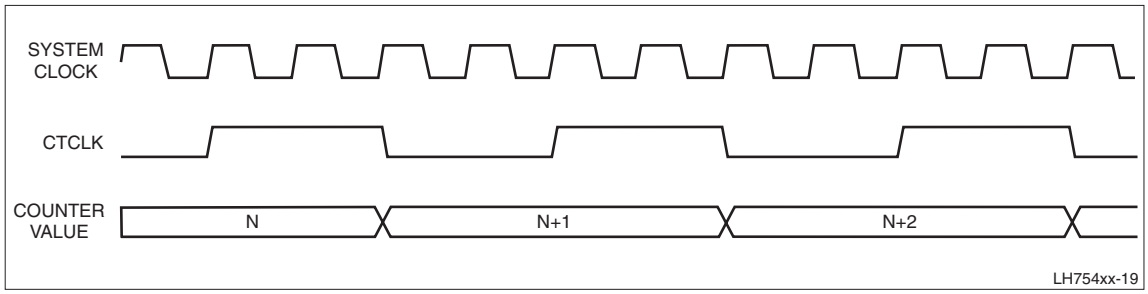


Figure 15-4. Count Clock Timing (System Clock in Phase with CTCLK)

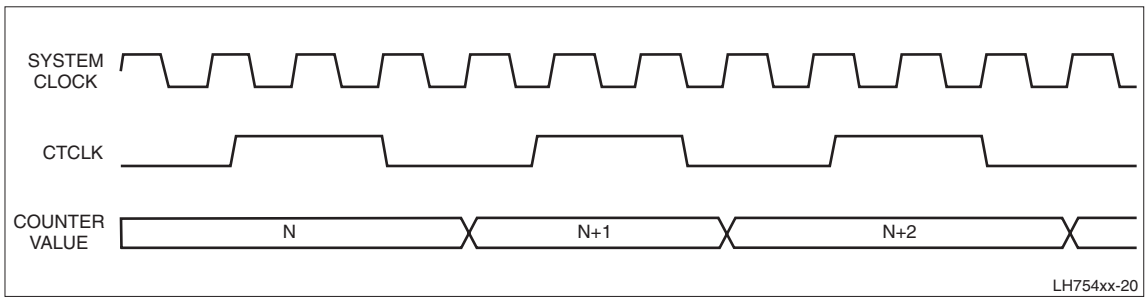


Figure 15-5. Count Clock Timing (System Clock not in Phase with CTCLK)

15.1.2 Counter Clear Upon Compare Match

A compare match occurs when the contents of the Timer Counter Register matches the value of the corresponding Timer Compare Register. When there is a compare match, one of two actions occurs, based on the setting of the TC bit in the corresponding Timer Control Register:

- If TC is set to 0, the counter is not cleared and continues counting.
- If TC is set to 1, the counter is cleared on the rising edge of the internal count clock.

Figure 15-6 shows an example of a timer's count being cleared when the count reaches the compare match value. This function is available with the compare value stored by the CMP1 Register. The Compare Match Detect signal shown in Figure 15-6 can be output through either:

- CTCCMP(n)A if the Timer Compare Register CMP0 is used.
- CTCCMP(n)B if the Timer Compare Register CMP1 is used.

The compare function is clocked by the internal count clock, which is the prescaled system clock.

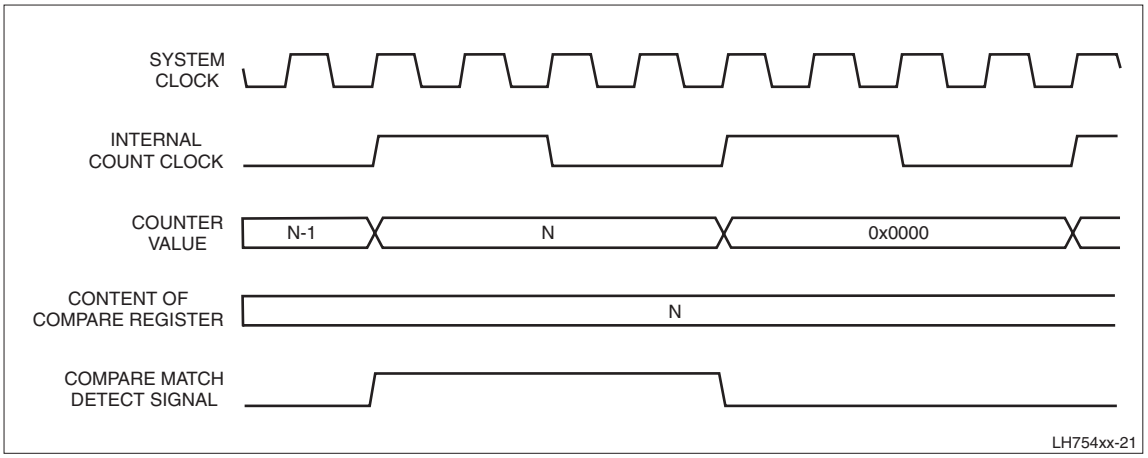


Figure 15-6. Compare Match and Counter Clear

15.1.3 Capture Signal Sampling

The capture signal is sampled on the rising edge of the system clock. The pulse width of a capture signal must be equal to or longer than two system clock periods plus, the setup time for the signal to be correctly read in. After sampling, the external capture signal is synchronized to the rising edge of the system clock. This synchronization process takes two system clock periods. After the external capture signal is synchronized, the value of the counter is stored in the appropriate CAP(n) Registers. The external capture signal is edge-selectable and can use a rising or falling edge to capture the counter value.

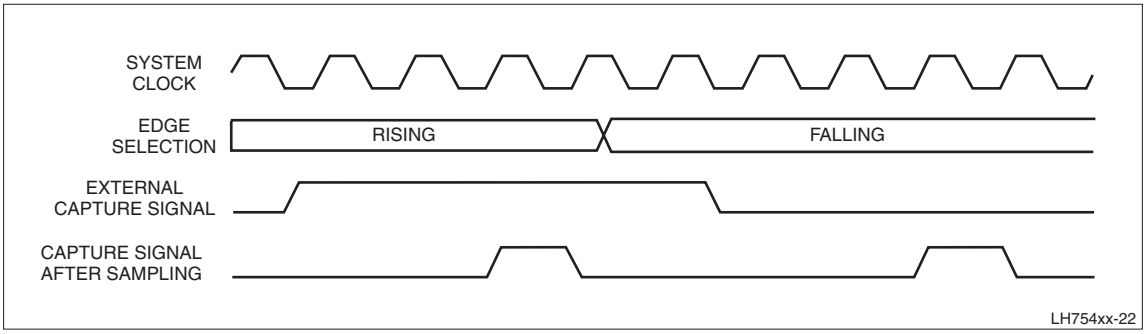


Figure 15-7. Capture Signal Synchronization Timing

15.1.4 PWM Mode

The timers support a Pulse Width Modulator (PWM) Mode. This mode uses the two Timer Compare Registers associated with a timer to create a PWM. In this mode, the PWM signal is output through CTCMP(n)A.

To enable PWM Mode for a timer, set the PWM bit in the appropriate Timer Control Register to 1. If PWM Mode is enabled, set bit TC of the same Timer Control Register to 1. This setting clears the counter when the Timer Control Register value matches the CMP1 value. The state of CTCMP*n*A matches the value of CMP1 starting from the time the counter matches the CMP1 Register until it matches the CMP0 Register. The state of CTCMP(n)A then matches the value of CMP0.

To create a proper PWM, set the CMP0 bit of the Timer Control Register opposite to the setting for the CMP1 bit. For example, if CMP1 of the CMP_CAP_CTRL Register is set to 10 (output 1 to CTCMP0B), CMP0 of the same register must be set to 01 (output 0 to CTCMP0A). Failing to follow this requirement results in a steady logic 1 or 0.

In PWM Mode, PWM CTCMP(n)B remains at the logic level (1 or 0) programmed into the CMP0 bits. When programming the compare values, the CMP1 Register must always be greater than the CMP0 Register to create a PWM signal.

- The value in CMP1 Register is the period of the PWM plus one.
- The value in CMP0 is the duty cycle of the PWM plus one.

The PWM is clocked by the internal count clock, which is the prescaled system clock.

Figure 15-8 shows an example of PWM output signal timing. To support the timing shown in this figure, the following values need to be programmed into the registers.

- CMP1 = 0x0005 → Period
- CMP0 = 0x0001 → Period

Timer 0 settings:

- CMP_CAP_CTRL[15] = 1 → PWM mode
- CMP_CAP_CTRL[14] = 1 → Counter Clear
- CMP_CAP_CTRL[13:12] = 01 → CMP1
- CMP_CAP_CTRL[11:10] = 10 → CMP0

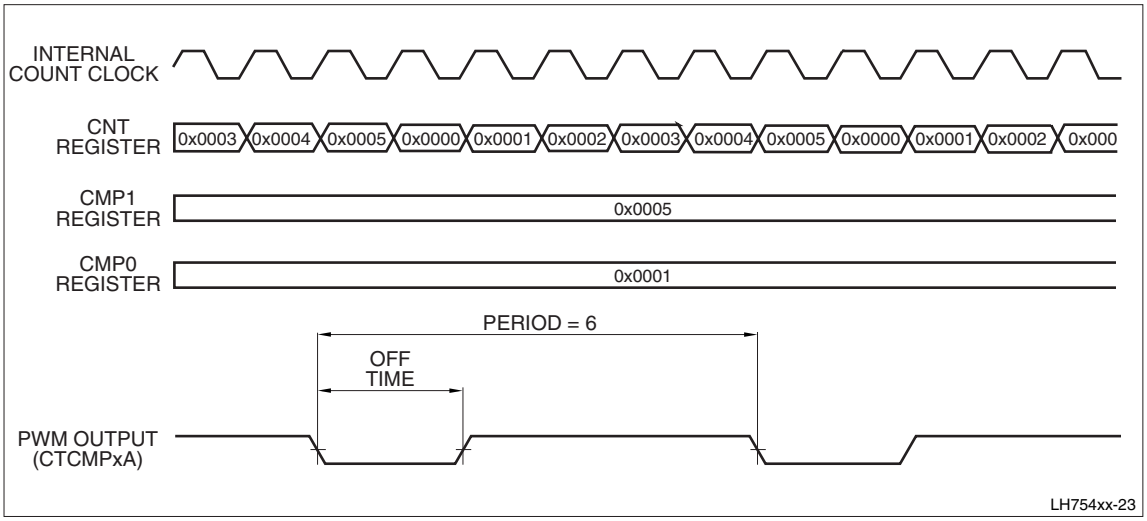


Figure 15-8. PWM Output Signal Timing

15.2 Timer Programmer's Model

The base address for the timers is:

Timer Base Address: 0xFFFC4000

15.2.1 Timer Register Summary

Table 15-1. Timer 0 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x00	RW	0x0000	Timer 0 Control Register
CMP_CAP_CTRL	0x04	RW	0x0000	Timer 0 Compare/Capture Control Register
INT_CTRL	0x08	RW	0x0000	Timer 0 Interrupt Control Register
STATUS	0x0C	RW	0x0000	Timer 0 Status Register
CNT	0x10	RW	0x0000	Timer 0 Counter Register
CMP0	0x14	RW	0xFFFF	Timer 0 Compare Register 0
CMP1	0x18	RW	0xFFFF	Timer 0 Compare Register 1
CAP0	0x1C	R	0x0000	Timer 0 Capture Register 0
CAP1	0x20	R	0x0000	Timer 0 Capture Register 1
CAP2	0x24	R	0x0000	Timer 0 Capture Register 2
CAP3	0x28	R	0x0000	Timer 0 Capture Register 3
CAP4	0x2C	R	0x0000	Timer 0 Capture Register 4

Table 15-2. Timer 1 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x30	RW	0x0000	Timer 1 Control Register
INT_CTRL	0x34	RW	0x0000	Timer 1 Interrupt Control Register
STATUS	0x38	RW	0x0000	Timer 1 Status Register
CNT	0x3C	RW	0x0000	Timer 1 Counter Register
CMP0	0x40	RW	0xFFFF	Timer 1 Compare Register 0
CMP1	0x44	RW	0xFFFF	Timer 1 Compare Register 1
CAP0	0x48	R	0x0000	Timer 1 Capture Register 0
CAP1	0x4C	R	0x0000	Timer 1 Capture Register 1

Table 15-3. Timer 2 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x50	RW	0x0000	Timer 2 Control Register
INT_CTRL	0x54	RW	0x0000	Timer 2 Interrupt Control Register
STATUS	0x58	RW	0x0000	Timer 2 Status Register
CNT	0x5C	RW	0x0000	Timer 2 Counter Register
CMP0	0x60	RW	0xFFFF	Timer 2 Compare Register 0
CMP1	0x64	RW	0xFFFF	Timer 2 Compare Register 1
CAP0	0x68	R	0x0000	Timer 2 Capture Register 0
CAP1	0x6C	R	0x0000	Timer 2 Capture Register 1

15.2.2 Timer Register Definitions

15.2.2.1 Timer 0 Control Register

Table 15-4. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											SEL			CS	CCL
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x00															

Table 15-5. CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4:2	SEL	**Count Clock Select** Specifies the system count clock. 000 = System clock/2 001 = System clock/4 010 = System clock/8 011 = System clock/16 100 = System clock/32 101 = System clock/64 110 = System clock/128 111 = CTCLK The CS field (bit [1]) must be clear for changes to the SEL field to take effect.
1	CS	**Start/Stop Counter 0** Specifies whether counter 0 is stopped or started. 0 = Stops counter 0 1 = Starts counter 0 This bit must be cleared for changes to the SEL bit (bit [2]) to take effect. For more information, see Section 15.1.1.
0	CCL	**Counter 0 Clear** Writes a 1 to clear the contents of timer 0 to 0x0000. Write operations of 0 are ignored. This bit always reads as zero and is reset to 0 as the counter is cleared.

15.2.2.2 Timer 0 Compare/Capture Control Register

CMP_CAP_CTRL is the Timer 0 Compare / Capture Control Register.

Table 15-6. CMP_CAP_CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	PWM	TC	CMP1		CMP0		CAP4		CAP3		CAP2		CAP1		CAP0	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x04															

Table 15-7. CMP_CAP_CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15	PWM	**PWM Output** Allows CTCMP0A to be used as a PWM output. This is done by properly setting up this bit as well as other bits in this register. Refer to Section 15.1.1 for a complete explanation. 0 = Outputs CTCMP0A is normal and works only with the CMP0 Register. 1 = Outputs CTCMP0A is in PWM Mode.
14	TC	**Timer 0 Counter Operation** Determines whether the Timer 0 counter is to operate as either a free running counter or as an interval timer. When set, the counter clears upon matching the CMP1 Register for Timer 0. This operation is only available with the CMP1 Register. Refer to Section 15.1.1 for a complete explanation. 0 = Inhibits counter clear (operates as free running counter). 1 = Clears counter when CNT for Timer 0 and CMP1 for Timer 0 match.
13:12	CMP1	**Output Value Select** Sets the reference value (at which compare match should occur) that is to be output to CTCMP0B when the CNT Register for Timer 0 matches the CMP1 Register for Timer 0. 00 = No change occurs to the output CTCMP0B. 01 = Output 0 to CTCMP0B. 10 = Output 1 to CTCMP0B. 11 = Toggle the output to CTCMP0B.
11:10	CMP0	**Output Value Select** Sets the reference value (at which compare match should occur) that is to be output to CTCMP0A when the CNT Register for Timer 0 matches the CMP0 Register for Timer 0. 00 = No change occurs to the output CTCMP0A. 01 = Output 0 to CTCMP0A. 10 = Output 1 to CTCMP0A. 11 = Toggle the output to CTCMP0A.

Table 15-7. CMP_CAP_CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
9:8	CAP4	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP0E is ignored 01 = Rising edge of CTCAP0E 10 = Falling edge of CTCAP0E 11 = Both edges of CTCAP0E
7:6	CAP3	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP0D is ignored 01 = Rising edge of CTCAP0D 10 = Falling edge of CTCAP0D 11 = Both edges of CTCAP0D
5:4	CAP2	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP0C is ignored 01 = Rising edge of CTCAP0C 10 = Falling edge of CTCAP0C 11 = Both edges of CTCAP0C
3:2	CAP1	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP0B is ignored 01 = Rising edge of CTCAP0B 10 = Falling edge of CTCAP0B 11 = Both edges of CTCAP0B
1:0	CAP0	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP0A is ignored 01 = Rising edge of CTCAP0A 10 = Falling edge of CTCAP0A 11 = Both edges of CTCAP0A

15.2.2.3 Timer 0 Interrupt Control Register

Table 15-8. INT_CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								CAP4_EN	CAP 3_EN	CAP2_EN	CAP1_EN	CAP0_EN	CMP1_EN	CMP0_EN	OVF_EN
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x08															

Table 15-9. INT_CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	CAP4_EN	**Timer 0 Interrupt Enable During Capture 4 Operation** 0 = No interrupt request occurs for capture 4. 1 = Interrupt request occurs for capture 4.
6	CAP3_EN	**Timer 0 Interrupt Enable During Capture 3 Operation** 0 = No interrupt request occurs for capture 3. 1 = Interrupt request occurs for capture 3.
5	CAP2_EN	**Timer 0 Interrupt Enable During Capture 2 Operation** 0 = No interrupt request occurs for capture 2. 1 = Interrupt request occurs for capture 2.
4	CAP1_EN	**Timer 0 Interrupt Enable During Capture 1 Operation** 0 = No interrupt request occurs for capture 1. 1 = Interrupt request occurs for capture 1.
3	CAP0_EN	**Timer 0 Interrupt Enable During Capture 0 Operation** 0 = No interrupt request occurs for capture 0. 1 = Interrupt request occurs for capture 0.
2	CMP1_EN	**Timer 0 Interrupt Enable Upon Compare 1** 0 = No interrupt request occurs for compare 1. 1 = Interrupt request occurs for compare 1.
1	CMP0_EN	**Timer 0 Interrupt Enable Upon Compare 0** 0 = No interrupt request occurs for compare 0. 1 = Interrupt request occurs for compare 0.
0	OVF_EN	**Timer 0 Interrupt Overflow Enable** 0 = No interrupt request occurs when counter overflows. 1 = Interrupt request occurs when counter overflows.

15.2.2.4 Timer 0 Status Register

The Status Register bits are independent of the individual interrupt enables. They are set to 1 upon all compare, capture, and overflow occurrences. To clear the status bits, write a 1 to the individual bits. This action clears the bit that was set in the register and clears the corresponding interrupt, with the following exception. If the timer is stopped and the Timer 0 Compare Register (CMP0 or CMP1) value matches the Timer 0 Counter Register (CNT), the corresponding status bit cannot be cleared until either the Timer 0 Compare Register or the Timer 0 Counter Register value is changed.

Writing a 0 to any of the status bits does not affect the corresponding interrupt. Similarly, writing a 1 to a bit that is not set does not affect the Status Register or interrupt.

Table 15-10. Status Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								CAP4_ST	CAP 3_ST	CAP2_ST	CAP1_ST	CAP0_ST	CMP1_ST	CMP0_ST	OVF_ST
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x0C															

Table 15-11. Status Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	CAP4_ST	**Timer 0 Capture 4 Status** To clear, write 1.
6	CAP3_ST	**Timer 0 Capture 3 Status** To clear, write 1.
5	CAP2_ST	**Timer 0 Capture 2 Status** To clear, write 1.
4	CAP1_ST	**Timer 0 Capture 1 Status** To clear, write 1.
3	CAP0_ST	**Timer 0 Capture 0 Status** To clear, write 1.
2	CMP1_ST	**Timer 0 Compare 1 Status** To clear, write 1.
1	CMP0_ST	**Timer 0 Compare 0 Status** To clear, write 1.
0	OVF_ST	**Timer 0 Overflow Status** To clear, write 1.

15.2.2.5 Timer 0 Counter Register

The CNT Register is a 16-bit, Read/Write up counter. The counter can be read or written to while it is operating. As a result, counts can be read at any time or the current count can be changed.

Table 15-12. CNT Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	TM0CNT															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x10															

Table 15-13. CNT Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	TM0CNT	**Timer 0 Count** Value of the 16-bit Up Counter.

15.2.2.6 Timer 0 Compare Registers

There are two CMP(n) Registers for Timer 0. They are designated:

- CMP0
- CMP1

Each register is a 16-bit, read/write register. Contents of these registers are compared continuously with the counter CNT. When both register and counter values match, a trigger signal is generated.

Table 15-14. CMP(n) Registers

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	TM0CMP															
RESET	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	CMP0: 0xFFFC4000 + 0x14 CMP1: 0xFFFC4000 + 0x18															

Table 15-15. CMP(n) Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	TM0CMP	**Timer 0 Compare** 16-bit compare register value.

15.2.2.7 Timer 0 Capture Registers

There are five CAP(n) Registers for Timer 0. They are designated:

- CAP0
- CAP1
- CAP2
- CAP3
- CAP4

Each register is a 16-bit, Read Only register. When a capture condition occurs, the contents of the counter CNT are stored into the associated Capture Register. Capture Registers correspond to the input signals CTCAP0A through CTCAP0E, respectively. The edge of the input signal used to trigger the capturing operation is determined by setting the CMP_CAP_CTRL Register.

Table 15-16. CAP(n) Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	CAP0															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	CAP0: 0xFFFC4000 + 0x1C CAP1: 0xFFFC4000 + 0x20 CAP2: 0xFFFC4000 + 0x24 CAP3: 0xFFFC4000 + 0x28 CAP4: 0xFFFC4000 + 0x2C															

Table 15-17. CAP(n) Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	CAP0	**Capture Register** 16-bit capture register value.

15.2.2.8 Timer 1 Control Register

Table 15-18. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	PWM	TC	CMP1		CMP0		CAP1		CAP0		SEL			CS	CCL
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x30															

Table 15-19. CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:15	///	**Reserved** Write the reset value.
14	PWM	**PWM Output** This bit allows the use of CTCMP1A as a PWM output. This is done by properly setting up this bit as well as other bits in this register. Refer to Section 15.1.1 for a complete explanation. 0 = Output CTCMP1A is normal and works only with the CMP1 Register. 1 = Output CTCMP1A is in PWM Mode.
13	TC	**Timer 1 Operation** This bit determines whether Timer 1 counter is to operate as either a free running counter or as an interval timer. When set, the counter clears upon matching CMP1 for Timer 1. This operation is only available with the CMP1 Register for Timer 1. Refer to Section 15.1.1 for a complete explanation. 0 = Inhibit counter clear (operates as free running counter). 1 = Clear counter when CNT for Timer 1 matches CMP1 for Timer 1.
12:11	CMP1	**Output Value Select** Sets the reference value (at which compare match should occur) that is to be output to CTCMP1B when CNT for Timer 1 matches the CMP1 Register for Timer 1. 00 = No change occurs to the output CTCMP1B. 01 = Output '0' to CTCMP1B. 10 = Output '1' to CTCMP1B. 11 = Toggle the output to CTCMP1B.
10:9	CMP0	**Output Value Select** Sets the reference value (at which compare match should occur) that is to be output to CTCMP1A when the CNT Register for Timer 1 matches the CMP0 Register for Timer 1. 00 = No change occurs to the output CTCMP1A. 01 = Output 0 to CTCMP1A. 10 = Output 1 to CTCMP1A. 11 = Toggle the output to CTCMP1A.

Table 15-19. CTRL Register Definitions (Cont'd)

BITS	FIELD NAME	DESCRIPTION
8:7	CAP1	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP1B is ignored 01 = Rising edge of CTCAP1B 10 = Falling edge of CTCAP1B 11 = Both edges of CTCAP1B
6:5	CAP0	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP1A is ignored 01 = Rising edge of CTCAP1A 10 = Falling edge of CTCAP1A 11 = Both edges of CTCAP1A
4:2	SEL	**Count Clock Select** 000 = System clock/2 001 = System clock/4 010 = System clock/8 011 = System clock/16 100 = System clock/32 101 = System clock/64 110 = System clock/128 111 = CTCLK The CS field (bit [1]) must be clear for changes to the SEL field to take effect.
1	CS	**Start/Stop Counter 1** 0 = Stop counter 1 1 = Start counter 1
0	CCL	**Counter 1 Clear** Write a 1 to clear the contents of Timer 1 to 0x0000. Write operations of 0 are ignored. This bit always reads 0 and is reset to 0 as the counter is cleared.

15.2.2.9 Timer 1 Interrupt Control Register

Table 15-20. INT_CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											CAP1_EN	CAP0_EN	CMP1_EN	CMP0_EN	OVF_EN
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x34															

Table 15-21. INT_CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	CAP1_EN	**Timer 1 Interrupt Enable During Capture 1 Operation** 0 = No interrupt request occurs for Capture 1. 1 = Interrupt request occurs for Capture 1.
3	CAP0_EN	**Timer 1 Interrupt Enable During Capture 0 Operation** 0 = No interrupt request occurs for Capture 0. 1 = Interrupt request occurs for Capture 0.
2	CMP1_EN	**Timer 1 Interrupt Enable Upon Compare 1** 0 = No interrupt request occurs for Compare 1. 1 = Interrupt request occurs for Compare 1.
1	CMP0_EN	**Timer 1 Interrupt Enable Upon Compare 0** 0 = No interrupt request occurs for Compare 0. 1 = Interrupt request occurs for Compare 0.
0	OVF_EN	**Timer 1 Interrupt Overflow Enable** 0 = No interrupt request occurs when counter overflows. 1 = Interrupt request occurs when counter overflows.

15.2.2.10 Timer 1 Status Register

The Status Register status bits are independent of the individual interrupt enables. They are set to 1 upon all compare, capture, and overflow occurrences. To clear the status bits, write 1s to the individual bits. This action clears the bit that was set in the register and clears the corresponding interrupt, with the following exception. If the timer is stopped and the Timer 1 Compare Register (CMP0 or CMP1) value matches the Timer 1 Counter Register (CNT), the corresponding status bit cannot be cleared until either the Timer 1 Compare Register or the Timer 1 Counter Register value is changed.

Writing a 0 to a status bit does not affect the corresponding interrupt. Similarly, writing a 1 to a bit that is not set does not affect the Status Register or Interrupt.

Table 15-22. Status Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											CAP1_ST	CAP0_ST	CMP1_ST	CMP0_ST	OVF_ST
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x38															

Table 15-23. Status Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	CAP1_ST	**Timer 1 Capture 1 Status** To clear, write 1.
3	CAP0_ST	**Timer 1 Capture 0 Status** To clear, write 1.
2	CMP1_ST	**Timer 1 Compare 1 Status** To clear, write 1.
1	CMP0_ST	**Timer 1 Compare 0 Status** To clear, write 1.
0	OVF_ST	**Timer 1 Overflow Status** To clear, write 1.

15.2.2.11 Timer 1 Counter Register

The CNT Register is a 16-bit, Read/Write up counter. The counter can be read or written to while it is operating. As a result, counts can be read at any time or the current count can be changed.

Table 15-24. CNT Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	TM1CNT															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x3C															

Table 15-25. CNT Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	TM1CNT	**Timer 1 Count** 16-bit Up Counter value.

15.2.2.12 Timer 1 Compare Registers

There are two CMP(n) Registers for Timer 1. They are designated:

- CMP0
- CMP1

Each register is a 16-bit, Read/Write register. Contents of these registers are compared continuously with the counter CNT. When both register and counter values match, a trigger signal is generated.

Table 15-26. CMP(n) Registers

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	TM1CMP															
RESET	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	CMP0: 0xFFFC4000 + 0x40 CMP1: 0xFFFC4000 + 0x44															

Table 15-27. CMP(n) Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	TM1CMP	**Timer 1 Compare** 16-bit Compare Register Value.

15.2.2.13 Timer 1 Capture Registers

There are two CAP(n) Registers for Timer 1. They are designated:

- CAP0
- CAP1

Each register is a 16-bit, Read Only register. When a capture condition occurs, the contents of the counter CNT are stored into the associated Capture Register. Capture Registers correspond to the input signals CTCAP1A through CTCAP1B, respectively. The edge of the input signal used to trigger the capturing operation is determined by setting the CTRL Register.

Table 15-28. CAP(n) Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	CAP1															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	CAP0: 0xFFFC4000 + 0x48 CAP1: 0xFFFC4000 + 0x4C															

Table 15-29. CAP(n) Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	CAP1	**Timer 1 Capture** 16-bit Capture Register Value.

15.2.2.14 Timer 2 Control Register

Table 15-30. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///	PWM	TC	CMP1		CMP0		CAP1		CAP0		SEL			CS	CCL
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x50															

Table 15-31. CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:15	///	**Reserved** Write the reset value.
14	PWM	**PWM Output** This bit allows the use of CTCMP2A as a PWM output. This is done by properly setting up this bit as well as other bits in this register. Refer to Section 15.1.1 for a complete explanation of this feature. 0 = Output CTCMP2A is normal and works only with the CMP2 Register. 1 = Output CTCMP2A is in PWM Mode.
13	TC	**Timer 2 Counter Operation** This bit determines whether the counter is to operate as a free running counter or interval counter. When set, the counter clears upon matching CMP1 for Timer 2. This operation is only available with the CMP1 Register for Timer 2. Refer to Section 15.1.1 for a complete explanation of this feature. 0 = Inhibit counter clear (operates as free running counter). 1 = Clear counter when CNT for Timer 2 matches CMP1 for Timer 2.
12:11	CMP1	**Output Value Select** Sets the reference value (at which compare match should occur) that is to be output to CTCMP2B when the CNT Register for Timer 2 matches the CMP1 Register for Timer 2. 00 = No change occurs to the output CTCMP2B. 01 = Output 0 to CTCMP2B. 10 = Output 1 to CTCMP2B. 11 = Toggle the output to CTCMP2B.
10:9	CMP0	**Output Value Select** Sets the reference value (at which compare match should occur) that is to be output to CTCMP2A when the CNT Register for Timer 2 matches the CMP0 Register for Timer 2. 00 = No change occurs to the output CTCMP2A. 01 = Output 0 to CTCMP2A. 10 = Output 1 to CTCMP2A. 11 = Toggle the output to CTCMP2A.

Table 15-31. CTRL Register Definitions (Cont'd)

BITS	FIELD NAME	DESCRIPTION
8:7	CAP1	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP2B is ignored. 01 = Rising edge of CTCAP2B. 10 = Falling edge of CTCAP2B. 11 = Both edges of CTCAP2B.
6:5	CAP0	**Input Edge Select** Selects the rising edge, falling edge, both edges, or ignores all changes of the input signal that is used as the capture trigger. 00 = Capture input CTCAP2A is ignored. 01 = Rising edge of CTCAP2A. 10 = Falling edge of CTCAP2A. 11 = Both edges of CTCAP2A.
4:2	SEL	**Count Clock Select** 000 = System clock/2 001 = System clock/4 010 = System clock/8 011 = System clock/16 100 = System clock/32 101 = System clock/64 110 = System clock/128 111 = CTCLK The CS field (bit [1]) must be clear for changes to the SEL field to take effect.
1	CS	**Start/Stop Counter 2** 0 = Stop Counter 2. 1 = Start Counter 2.
0	CCL	**Counter 2 Clear** Write a 1 to clear the contents of Timer 2 to 0x0000. Write operations of 0 are ignored. This bit always reads 0 and is reset to 0 as the counter is cleared.

15.2.2.15 Timer 2 Interrupt Control Register

Table 15-32. INT_CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											CAP1_EN	CAP0_EN	CMP1_EN	CMP0_EN	OVF_EN
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x54															

Table 15-33. INT_CTRL Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	CAP1_EN	**Timer 2 Interrupt Enable During Capture Operation** 0 = No interrupt request occurs for Capture 1. 1 = Interrupt request occurs for Capture 1.
3	CAP0_EN	**Timer 2 Interrupt Enable During Capture Operation** 0 = No interrupt request occurs for Capture 0. 1 = Interrupt request occurs for Capture 0.
2	CMP1_EN	**Timer 2 Interrupt Enable Upon Compare 1** 0 = No interrupt request occurs for Compare 1. 1 = Interrupt request occurs for Compare 1.
1	CMP0_EN	**Timer 2 Interrupt Enable Upon Compare** 0 = No interrupt request occurs for Compare 0. 1 = Interrupt request occurs for Compare 0.
0	OVF_EN	**Timer 2 Interrupt Overflow Enable** 0 = No interrupt request occurs when Counter 1 overflows. 1 = Interrupt request occurs when Counter 1 overflows.

15.2.2.16 Timer 2 Status Register

The Status Register bits are independent of the individual interrupt enables and are set to 1 upon all compare, capture, and overflow occurrences. To clear the status bits, write 1s to the individual bits. This action clears the bit that was set in the register and clears the corresponding interrupt, with the following exception. If the timer is stopped and the Timer 2 Compare Register (CMP0 or CMP1) value matches the Timer 2 Counter Register (CNT), the corresponding status bit cannot be cleared until either the Timer 2 Compare Register or the Timer 2 Counter Register value is changed.

Writing a 0 to any of the status bits does not affect the corresponding interrupt. Similarly, writing a 1 to a bit that is not set does not affect the Status Register or interrupt.

Table 15-34. Status Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											CAP1_ST	CAP0_ST	CMP1_ST	CMP0_ST	OVF_ST
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x58															

Table 15-35. Status Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	CAP1_ST	**Timer 2 Capture 1 Status** To clear, write 1.
3	CAP0_ST	**Timer 2 Capture 0 Status** To clear, write 1.
2	CMP1_ST	**Timer 2 Compare 1 Status** To clear, write 1.
1	CMP0_ST	**Timer 2 Compare 0 Status** To clear, write 1.
0	OVF_ST	**Timer 2 Overflow Status** To clear, write 1.

15.2.2.17 Timer 2 Counter Register

The CNT Register is a 16-bit, read/write up counter. The counter can be read or written to while it is operating. As a result, counts can be read at any time or the current count can be changed.

Table 15-36. CNT Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	TM2CNT															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC4000 + 0x5C															

Table 15-37. CNT Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	TM2CNT	**Timer 2 Counter** 16-bit up-counter value.

15.2.2.18 Timer 2 Compare Registers

There are two CMP(n) Registers for Timer 2. They are designated:

- CMP0
- CMP1

Each register is a 16-bit, Read/Write register. Contents of these registers are compared continuously with the counter CNT. When both register and counter values match, a trigger signal is generated.

Table 15-38. CMP(n) Registers

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	TM2CMP															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	CMP0: 0xFFFC4000 + 0x60 CMP1: 0xFFFC4000 + 0x64															

Table 15-39. CMP(n) Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	TM2CMP	**Timer 2 Compare** 16-bit Compare Register Value.

15.2.2.19 Timer 2 Capture Registers

There are two CAP(n) Registers for Timer 2. They are designated:

- CAP0
- CAP1

Each register is a 16-bit, Read Only register. When a capture condition occurs, the contents of the counter CNT are stored into the associated Capture Register. Capture Registers correspond to the input signals CTCAP2A through CTCAP2B, respectively. The edge of the input signal used to trigger the capturing operation is determined by setting the CTRL Register.

Table 15-40. CAP(n) Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	CAP2															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	CAP0: xFFFC4000 + 0x68 CAP1: xFFFC4000 + 0x6C															

Table 15-41. CAP(n) Register Definitions

BITS	FIELD NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	CAP2	**Timer 2 Capture** 16-bit Capture Register Value.

15.2.2.20 Timer Interrupts

The timer interrupts are:

- Timer 0 Combined Interrupt — a combined interrupt formed by ORing the two compare, five capture, and one overflow interrupts in Timer 0.
- Timer 1 Combined Interrupt — a combined interrupt formed by ORing the two compare, two capture, and one overflow interrupts in Timer 1.
- Timer 2 Combined Interrupt — a combined interrupt formed by ORing the two compare, two capture, and one overflow interrupts in Timer 2.

Each Timer ORs its individual interrupt to provide a single combined interrupt.

If an individual interrupt is enabled and the corresponding interrupt condition (compare, capture, or overflow) occurs, a combined interrupt also occurs. Once the interrupt condition occurs, the combined Interrupt Output signal goes active to '1'. It remains active until all compare, capture, and overflow interrupts are cleared in the appropriate Status Register or disabled.

Chapter 16
Watchdog Timer

The Watchdog Timer (WDT) is a programmable timer that software has to reset at regular intervals. Failing to reset the timer causes an interrupt to the system. Failing to service the interrupt within the timeout period causes the WDT to set a flag that forces the RCPC to generate a System Reset. At the WDT timeout, the RCPC sets the WDTO bit in the Reset Status Register.

Figure 16-1 shows a block diagram of the WDT.

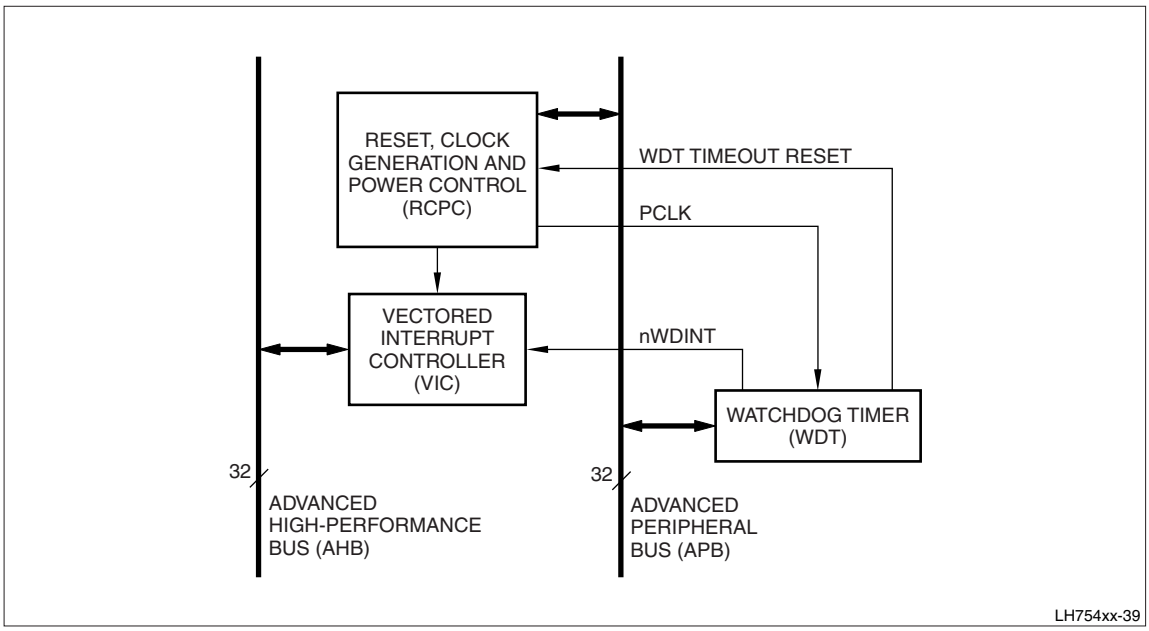


Figure 16-1. Watchdog Timer Block Diagram

16.1 WDT Features

The WDT has the following features:

- The counter generates an interrupt at a set interval and the count reloads from the preset value after reaching zero.
- The default timeout period is set to the minimum timeout of 2^{16} system clock cycles.
- The WDT is driven by the APB.
- A built-in protection mechanism guards against interrupt-service failure.
- The WDT can be programmed to trigger a System Reset on a timeout.
- The WDT can be programmed to trigger an interrupt on the first timeout; then, if the service routine fails to clear the interrupt, the next WDT timeout triggers a System Reset.

16.2 WDT Theory of Operation

All Control and Status Registers for the Watchdog Timer can be accessed through the APB. The Watchdog Timer consists of a 32-bit down-counter that causes a selectable time-out interval to detect malfunctions. The timer needs to be periodically reset by software. Failure to do so results in a time-out that causes an interrupt to be taken or a System Reset to be issued by the RCPC. There are 16 selectable time intervals for a time-out of 2^{16} through 2^{31} system clock cycles.

See Chapter 9 for a complete description about reset generation.

16.3 WDT Programmer's Model

The base address for the WDT is:

WDT Base Address: 0xFFFE3000

16.3.0.1 WDT Register Summary

Table 16-1. WDT Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x00	R W	0x00	Watchdog Control Register
CNTR	0x04	W	—	Watchdog Counter Reset
TSTR	0x08	R W	0x40	Watchdog Register
CNT0	0x0C	R	0x00	WDT Counter Section 0
CNT1	0x10	R	0x00	WDT Counter Section 1
CNT2	0x14	R	0x01	WDT Counter Section 2
CNT3	0x18	R	0x00	WDT Counter Section 3

16.3.1 WDT Register Definitions

16.3.1.1 Control Register

CTRL is the Control Register. The active bits used in this register are Read/Write.

Table 16-2. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								TOP				FRZ	///	RSP	EN
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	R	RW	RW
ADDR	0xFFFE3000 + 0x00															

Table 16-3. CTRL Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:4	TOP	**Time Out Period** Selects 1 of 16 possible values to load into the counter to determine the time-out period. Example: 0x0 = 2^{16}, 0xF = 2^{31} HCLK cycles. When setting or changing the time-out period, the new value will not come into affect until a counter-reset command has been issued or when the WDT times out.
3	FRZ	**Freeze EN Bit (set-only)** 0 = Do not stop the EN bit from being cleared when the watchdog is enabled. 1 = Stop the EN bit from being cleared when the watchdog is enabled. This avoids accidental write operations that disable the watchdog, and can only be cleared by a System Reset.
2	///	**Reserved** Write the reset value.
1	RSP	**Time-out Response** Determines the output response on a time-out period. 0 = Only a System Reset is generated on a time-out period. 1 = An interrupt is generated on the first time-out period. If this is not cleared, a System Reset is generated on the second time-out period.
0	EN	**Watchdog Enable/Disable** 0 = Watchdog disabled. Counter does not decrement, and no interrupts or System Resets will be generated by the watchdog. 1 = Watchdog enabled. Allows the counter to decrement, causing interrupts to be generated if the count is not periodically reset to stop the count value from reaching zero.

16.3.1.2 Counter Reset Register

CNTR is the Counter Reset Register. The active bits used in this register are Write Only.

Table 16-4. CNTR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	WDCNTR															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFE3000 + 0x04															

Table 16-5. CNTR Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	WDCNTR	**Time-out Count Down** Writing 0x1984 to this register causes the counter to start counting down the time-out period from the beginning and clears any interrupts generated by the watchdog.

16.3.1.3 Status Register

STR is the Status Register.

Table 16-6. STR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								nWDINTR	///	RSP		///			
RESET	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE3000 + 0x08															

Table 16-7. STR Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads 0.
7	nWDINTR	**nWDINTR Interrupt Status** 1 = WDT interrupt has triggered. 0 = WDT interrupt has not triggered.
6	///	**Reserved** Reads 1.
5:4	RSP	**Response Status** Holds the required response as set in CTRL Register (described in Section 16.3.1.1). The RSP field is, in fact, two copies of the RSP bit in the CTRL Register.
3:0	///	**Reserved** Reads 0.

16.3.1.4 Counter Section 0 Register

CNT0 is the Counter Section 0 Register.

Table 16-8. CNT0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Counter Sub-Section 0							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE3000 + 0x0C															

Table 16-9. CNT0 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads 0.
7:0	Counter Sub-Section 0	**Current Count Value** Holds bits [7:0] of the current count value.

16.3.1.5 Counter Section 1 Register

CNT1 is the Counter Section 1 Register.

Table 16-10. CNT1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Counter Sub-Section 1							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE3000 + 0x10															

Table 16-11. CNT1 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads 0.
7:0	Counter Sub-Section 1	**Current Count Value** Holds bits [15:8] of the current count value.

16.3.1.6 Counter Section 2 Register

CNT2 is the Counter Section 2 Register.

Table 16-12. CNT2 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Counter Sub-Section 2							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE3000 + 0x14															

Table 16-13. CNT2 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads 0.
7:0	Counter Sub-Section 2	**Current Count Value** Holds bits [23:16] of the current count value.

16.3.1.7 Counter Section 3 Register

CNT3 is the Counter Section 3 Register.

Table 16-14. CNT3 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Counter Sub-Section 3							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE3000 + 0x18															

Table 16-15. CNT3 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads 0.
7:0	Counter Sub-Section 3	**Current Count Value** Holds bits [31:24] of the current count value.

Chapter 17
Real-Time Clock

The Real Time Clock (RTC) is an AMBA slave module that connects to the APB. The RTC can provide a basic alarm function or act as a long-time base counter. This is achieved by generating an interrupt signal after counting for a programmed number of cycles of an RTC input. Counting in 1-second intervals is achieved using a 1 Hz clock input to the RTC.

Figure 17-1 shows a block diagram of the RTC.

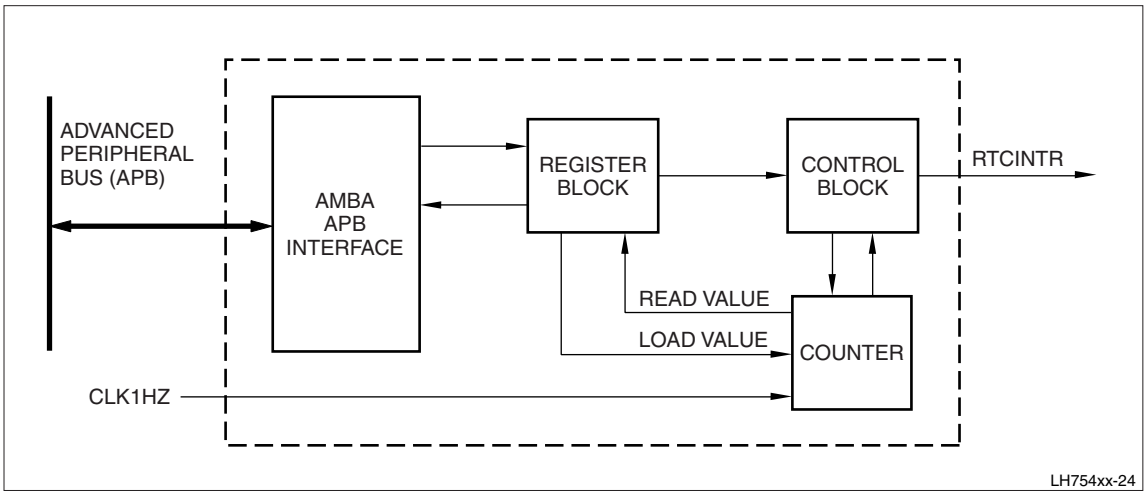


Figure 17-1. RTC Block Diagram

17.1 RTC Features

The RTC provides the following features:

- AMBA APB interface
- 32-bit up-counter with programmable load
- Programmable 32-bit match Compare Register
- Software-maskable interrupt that is set when the Counter and Compare Registers have identical values.

17.2 RTC Theory of Operation

The SoC reads and writes data and control/status information via the AMBA APB interface. The 32-bit counter increments on successive rising edges of the 1 Hz clock from the RCPC. This clock is generated in the RCPC by dividing down the 32.768 kHz crystal input to a 1 Hz frequency. The counter is loaded with a start value by writing to the load registers LR0 and LR1.

- LR0 loads the least-significant 16 bits.
- LR1 loads the most-significant 16 bits.

When loading a new counter value, write to LR0 first, then to LR1. Note, however, that a new start value does not load into the counter until the first rising edge of 1 Hz clock after LR1 is written to. For more information about these registers, refer to Section 17.3.2.6 and Section 17.3.2.7.

The counter value can be obtained at any time by reading Data Registers DR0 and DR1:

- DR0 contains the lower 16 bits of the counter.
- DR1 contains the upper 16 bits of the counter.

When reading the counter value, read DR0 first. This is because DR1 contains the value of the upper 16 bits when DR0 was last read, rather than the current value of the upper 16 bits. When the counter reaches the maximum value, 0xFFFFFFFF, it wraps to zero and continues incrementing. For more information about these registers, refer to Section 17.3.2.1 and Section 17.3.2.2.

Writing to MR0 and MR1 programs the least-significant 16 bits and most-significant 16 bits, respectively, of the Match Register. The counter and match values are compared in a comparator. When both values are equal, the interrupt RTCINTR is asserted HIGH. The software can use the interrupt to implement a basic time alarm function. Writing any data value to the Interrupt Clear Register, EOI, clears the interrupt. The value in the match register can be read at any time. For more information about these registers, refer to Section 17.3.2.3 and Section 17.3.2.4.

When using the RTC, resets that need to be done to the RTC should use bit [3] in the CTRL Register (see Section 17.3.2.8). Failing to reset the RTC using this bit can cause unpredictable results.

The RTCINTR interrupt can be masked by writing to the CTRL Register. The status of the interrupt can be obtained by reading the STAT Register.

17.3 RTC Programmer's Model

The base address for the RTC is:

RTC Base Address: 0xFFFE0000

17.3.1 RTC Register Summary

Table 17-1. RTC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
DR0	0x00	R		Lower 16-bit Data Register
DR1	0x04	R		Upper 16-bit Data Register
MR0	0x08	RW		Lower 16-bit Match Register
MR1	0x0C	RW		Upper 16-bit Match Register
STAT/EOI	0x10	RW		Interrupt Status Register (read)/ Interrupt Clear Register (write)
LR0	0x14	RW		Lower 16-bit Counter Load Register
LR1	0x18	RW		Upper 16-bit Counter Load Register
CTRL	0x1C	RW		Control Register

NOTE: All other address locations are reserved and must not be used during normal operation.

17.3.2 RTC Register Definitions

17.3.2.1 Data Register 0

DR0 is the Lower 16-bit Read Data Register. Reads from this register return the current value of the lower 16 bits of the counter. When this register is read, the DR1 Register is updated with the current value of the upper 16 bits of the counter.

Table 17-2. DR0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCDR0															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE0000 + 0x00															

NOTE: The reset value of this register's bits is indeterminate.

Table 17-3. DR0 Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved**
15:0	RTCDR0	**RTC Data Register 0** Specifies the current lower 16-bit counter value.

17.3.2.2 Data Register 1

DR1 is the Upper 16-bit Read Data Register. Reads from this register return the value of the upper 16 bits of the counter when DR0 was last accessed. The DR0 Register should be read before reading the DR1 Register to avoid the mismatch between the DR0 and the DR1 Registers due to a counter rollover.

Table 17-4. DR1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCDR1															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFE0000 + 0x04															

NOTE: The reset value of this register's bits is indeterminate.

Table 17-5. DR1 Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved**
15:0	RTCDR1	**RTC Data Register 1** Specifies the upper 16-bit counter value from when DR0 was last accessed.

17.3.2.3 Match Register 0

MR0 is the Lower 16-bit Read/Write Match Register. Writes to this register load the lower 16-bit Match Register. Reads return the last written value.

Table 17-6. MR0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCMR0															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE0000 + 0x08															

NOTE: The reset value of this register's bits is indeterminate.

Table 17-7. MR0 Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved**
15:0	RTCMR0	**RTC Match Register 0** Specifies the lower 16-bit Match Register.

17.3.2.4 Match Register 1

MR1 is the upper 16-bit Read/Write Match Register. Writes to this register load the Upper 16-bit Match Register; reads return the last written value.

Table 17-8. MR1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCMR1															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE0000 + 0x0C															

NOTE: The reset value of this register's bits is indeterminate.

Table 17-9. MR1 Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved**
15:0	RTCMR1	**RTC Match Register 1** Specifies the upper 16-bit Match Register.

17.3.2.5 Interrupt Status/Clear

STAT/EOI is the Interrupt Status/Clear Register. The write location is a virtual address with no physical storage element. A write to this location clears the RTCINTR interrupt line and the corresponding status bit. A read from bit 0 returns the value of RTCINTR.

Table 17-10. STAT/EOI Register (Write Operations)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	RTCEOI															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCEOI															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFE0000 + 0x10															

NOTE: The reset value of this register's bits is indeterminate.

Table 17-11. STAT/EOI Register Definitions (Write Operations)

BITS	NAME	DESCRIPTION
31:0	RTCEOI	**End Of Interrupt** A write to this register clears RTCINTR, regardless of the data value written.

Table 17-12. STAT/EOI Register (Read Operations)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///															RTCINTR
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW
ADDR	0xFFFE0000 + 0x10															

NOTE: The reset value of this register's bits is indeterminate.

Table 17-13. STAT/EOI Register Definitions (Read Operations)

BITS	NAME	DESCRIPTION
31:1	///	**Reserved** Reads are undefined.
0	RTCINTR	**Interrupt Status** 1 = RTCINTR interrupt is asserted. 0 = RTCINTR interrupt is not asserted.

17.3.2.6 Read/Write Load Register 0

LR0 is the Lower 16-bit Read/Write Load Register. Writes to this register load the least-significant 16 bits of an Intermediate Register. The intermediate Register is not loaded into the free-running counter until the rising edge of a 1 Hz clock follows a write operation to LR1. Read operations return the last value written.

Table 18. LR0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCLR0															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE0000 + 0x14															

NOTE: The reset value of this register's bits is indeterminate.

Table 19. LR0 Register Definitions

BIT	NAME	FUNCTION
31:16	///	**Reserved** Reads are undefined.
15:0	RTCLR0	**RTC Load Register 0** Specifies the lower 16-bit Counter Load Register.

17.3.2.7 Read/Write Load Register 1

LR1 is the Upper 16-bit Read/Write Load Register. Writes to this register load the most-significant 16 bits of an Intermediate Register. The Intermediate Register is not loaded into the free-running counter until the rising edge of CLK1HZ. LR0 should be written to first, followed by LR1, when the counter is to be reinitialized. Reads return the last written value.

Table 20. LR1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RTCLR1															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE0000 + 0x18															

NOTE: The reset value of this register's bits is indeterminate.

Table 21. LR1 Register Definitions

BIT	NAME	FUNCTION
31:16	///	**Reserved** Reads are undefined.
15:0	RTCLR1	**RTC Load Register 1** Specifies the upper 16-bit Counter Load Register.

17.3.2.8 Control Register

CTRL is a 1-bit Control Register that controls the masking of the interrupt generated by the RTC. Writing a '1' to bit position 0 enables the interrupt. Writing a '0' disables the interrupt. Reads to this register return the last value written at bit position 0.

Table 22. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///															MIE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFE0000 + 0x1C															

Table 23. CTRL Register Definitions

BIT	NAME	FUNCTION
31:1	///	**Reserved** Write the reset value.
0	MIE	**Match Interrupt Enable** 0 = Match interrupt is not enabled. 1 = Match interrupt is enabled.

17.3.3 RTC Interrupts

The RTC generates a single maskable active HIGH interrupt, RTCINTR, when a match occurs between the counter and the MR Registers. This interrupt is enabled or disabled by changing the mask bit in the CTRL Register. To enable the interrupt, set bit [0] of the CTRL Register to HIGH.

The status of the interrupt mask can be read via the CTRL Register. The RTCINTR status can be read from bit [0] in the STAT Register. Writing to EOI clears the RTCINTR flag.

Chapter 18
Synchronous Serial Port

The Synchronous Serial Port (SSP) is a master-only interface for synchronous serial communication with slave peripheral devices that have Motorola SPI, National Semiconductor Microwire, or Texas Instruments DSP-compatible synchronous serial interfaces. Figure 18-1 shows a block diagram of the SSP.

18.1 SSP Features

The SSP block provides the following features.

- Synchronous Serial Interface in Master Only Mode. The SSP performs serial communications as a master device in one of three modes:
  - Motorola SPI
  - Texas Instruments DSP-compatible synchronous serial interface
  - National Semiconductor Microwire
- Two 16-bit-wide, 8-entry-deep FIFOs, one for transmitting data and one for receiving data:
  - The transmit FIFO takes data written to it and transmits it on the serial interface.
  - The receive FIFO parallellizes the serial data stream and presents it in a FIFO for access by other devices.
- If the receive FIFO is not empty, the SSP can assert an interrupt after a specified number of clock ticks elapses following the start of a receive transfer. This feature permits interrupt-driven data transfers that are greater than the FIFO watermark, but not an even multiple of it.
- Programmable clock bit rate.
- Programmable data frame size, from 4 to 16 bits long, depending on the size of data programmed. Each frame transmits starting with the most-significant bit.
- Four interrupts, each of which can be individually enabled or disabled using the SSP Control Register bits. A combined interrupt is also generated as an OR function of the individual interrupt requests.
- Loopback Test Mode.

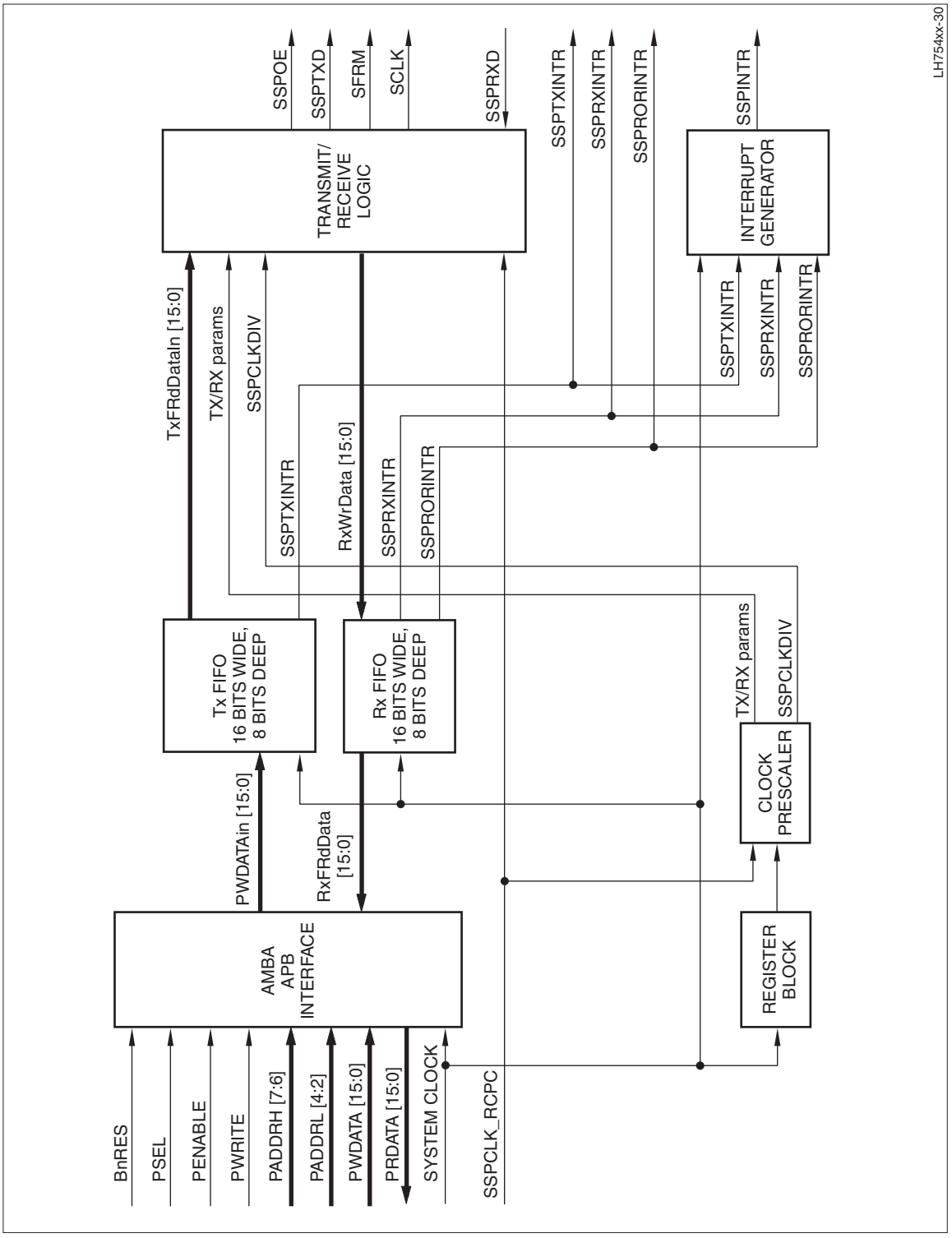


Figure 18-1. Synchronous Serial Port Block Diagram

18.2 SSP Theory Of Operation

The SSP performs serial-to-parallel conversion on data received from a peripheral device. The transmit and receive paths are buffered with internal FIFO memories. These memories can store eight 16-bit values independently in both transmit and receive modes. Serial data is transmitted on SSPTX and received on SSPRX. For the SSP to behave properly, all registers must be written to.

During transmission, data writes to the transmit FIFO via the APB interface. The transmit data is queued up for parallel-to-serial conversion onto the transmit interface. The transmit logic formats the data into one of three basic frame types:

- Motorola SPI
- Texas Instruments DSP-compatible Synchronous Serial Interface
- National Semiconductor Microwire.

Table 18-1 describes these modes.

For all three formats, the serial clock (SSPCLK) is held inactive while the SSP is idle and transitions at the programmed frequency only during active transmission of data. The SSPCLK pin can be HIGH during idle in SPI Mode if the SPO bit in the Control Register is set.

For Motorola SPI and National Semiconductor Microwire frame formats, the serial frame (SSPFRM) pin is active LOW and asserted (pulled down) during the entire frame transmission. Both formats output data on the falling edge of the clock and latch input data on the rising edge of the clock.

Table 18-1. Feature Comparison

MODE	DESCRIPTION	DATA TRANSFERS	MISCELLANEOUS
SPI	Lets the SSP communicate with Motorola SPI-compatible devices.	Full-duplex, 4-wire synchronous	Clock polarity and phase are programmable.
SSI	Lets the SSP communicate with Texas Instruments DSP-compatible Serial Synchronous Interface devices.	Full-duplex, 4-wire synchronous	
Microwire	Lets the SSP communicate with National Semiconductor Microwire-compatible devices.	Half-duplex synchronous, using 8-bit control messages	

18.3 SSP Timing Waveforms

Figure 18-2 shows the standard set of SSP timing waveforms.

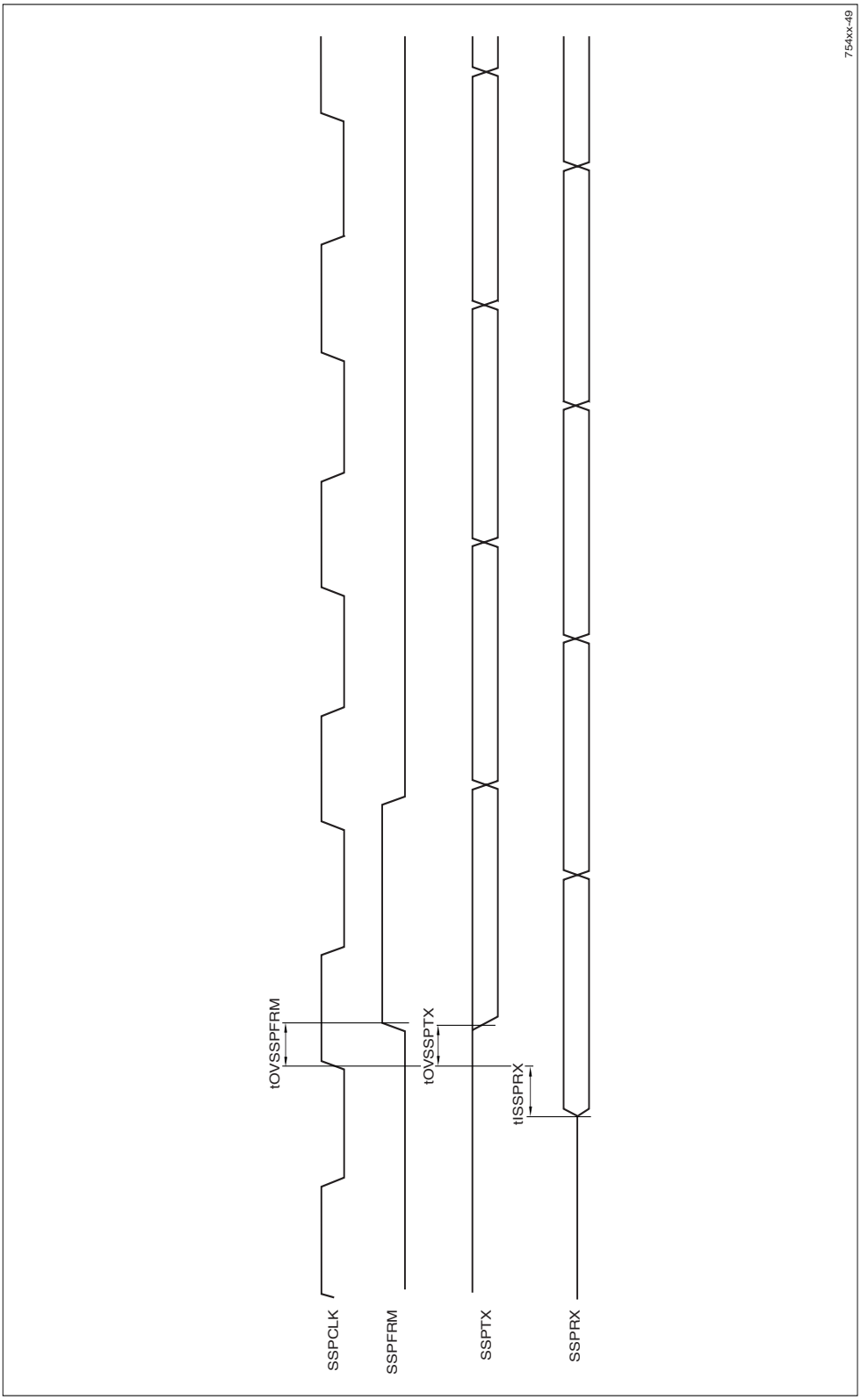
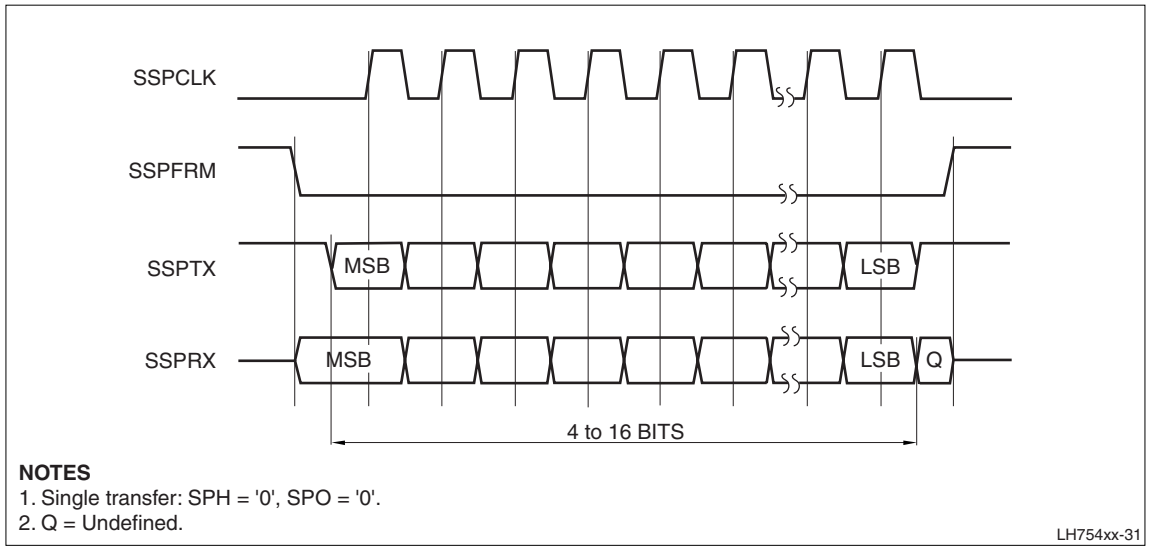


Figure 18-2. SSP Timing Waveform

18.3.1 Motorola SPI Frame Format

For the Motorola SPI format, the serial frame pin (SSPFRM) is active LOW. The SPO and SPH bits in SSP Control Register 0 influence SSPCLK and SSPFRM operation in Single and Continuous Modes.

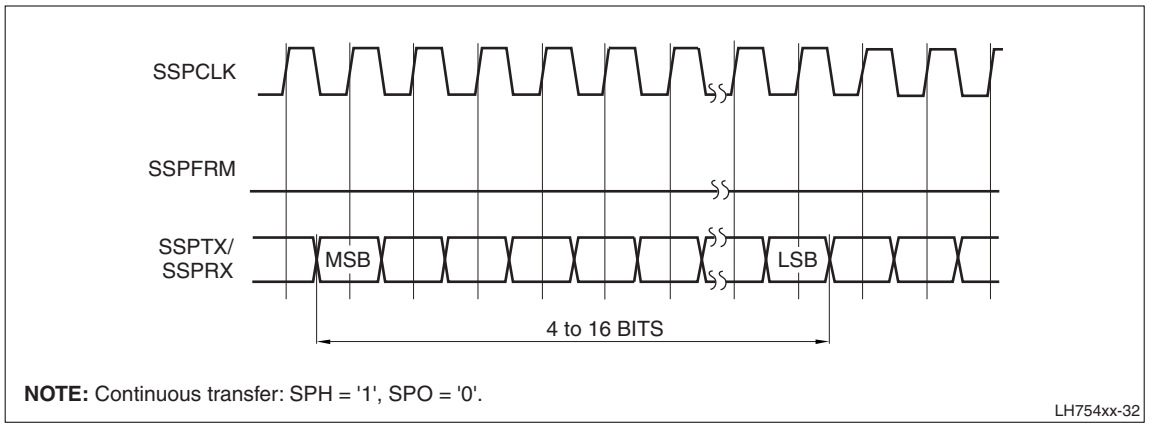
Figure 18-3 through Figure 18-6 show the Motorola SPI frame format for single data transfers, continuous data transfers, and when SPH equals 0 or 1.



NOTES
1. Single transfer: SPH = '0', SPO = '0'.
2. Q = Undefined.

LH754xx-31

Figure 18-3. Motorola SPI Frame Format (Single Transfer)



NOTE: Continuous transfer: SPH = '1', SPO = '0'.

LH754xx-32

Figure 18-4. Motorola SPI Frame Format (Continuous Transfer)

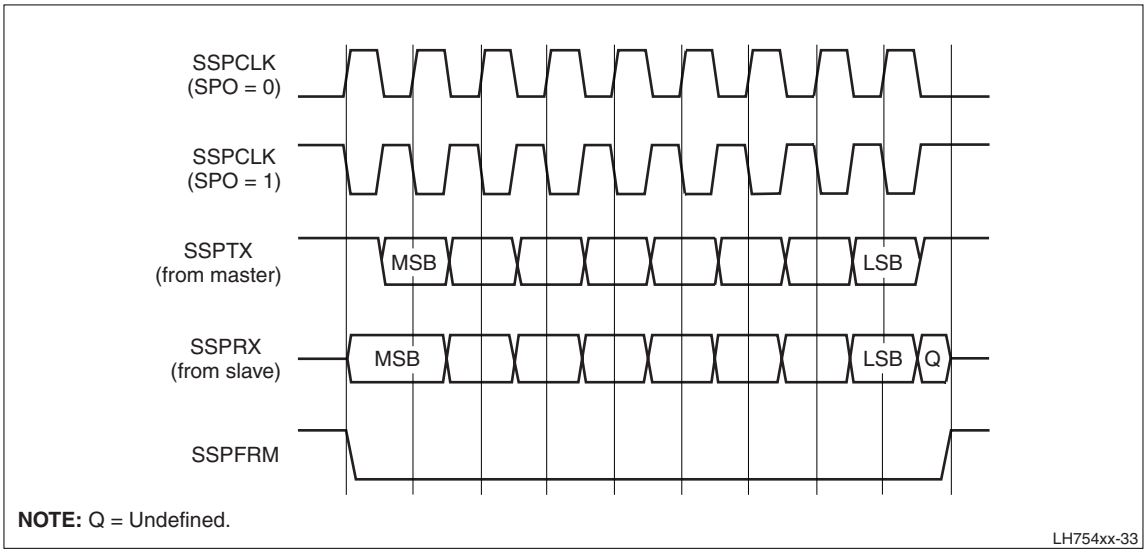


Figure 18-5. Motorola SPI Frame Format with SPH = 0

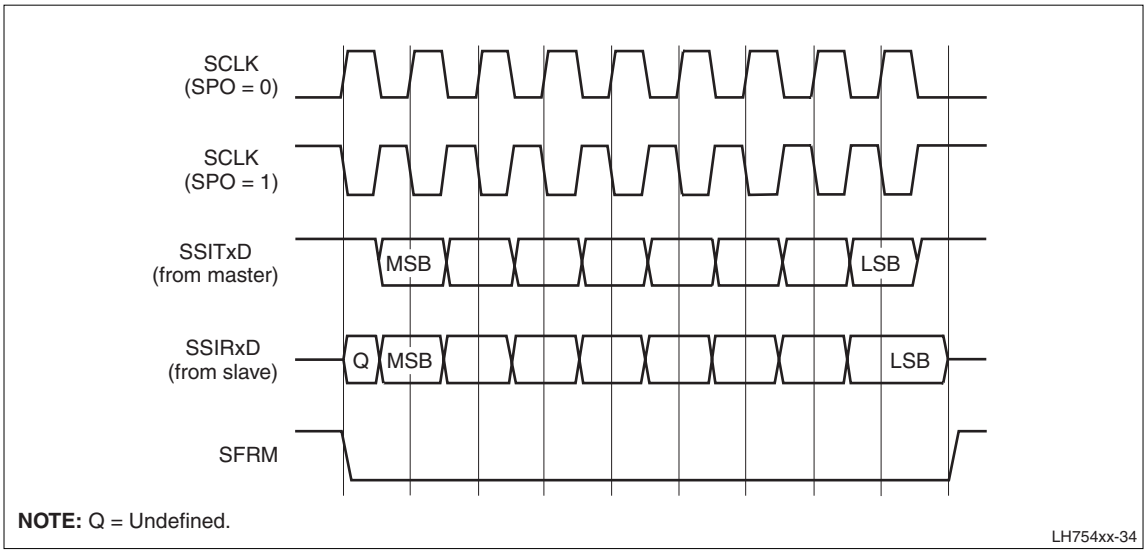


Figure 18-6. Motorola SPI Frame Format with SPH = 1

18.3.2 Texas Instruments Frame Format

For the Texas Instruments DSP-compatible synchronous serial interface frame format, the SSPFRM pin is pulsed for one serial clock period stating at its rising edge, prior to each frame's transmission. For this frame format, the SSP outputs data on the rising edge of the clock and latches input data on the rising edge of the clock.

Figure 18-7 shows the Texas Instruments DSP format for a single transfer. Figure 18-8 shows the same format for continuous transfers.

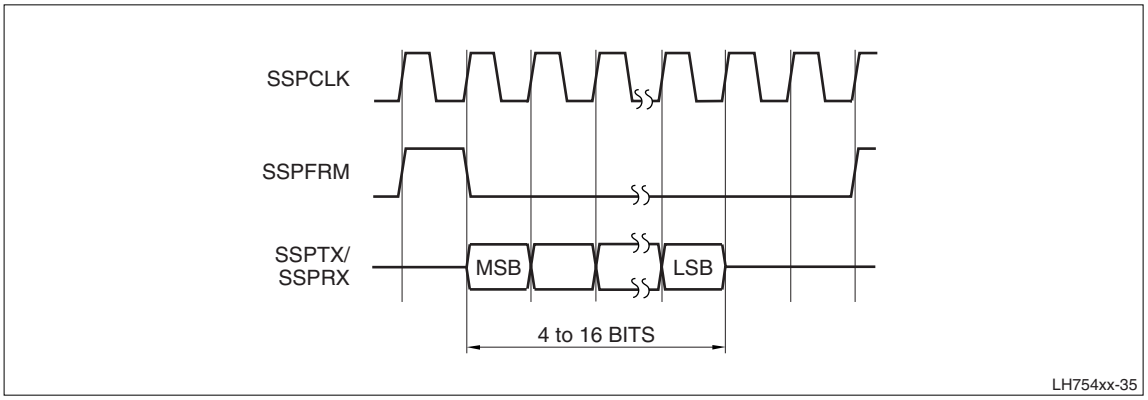


Figure 18-7. Texas Instruments Synchronous Serial Frame Format (Single Transfer)

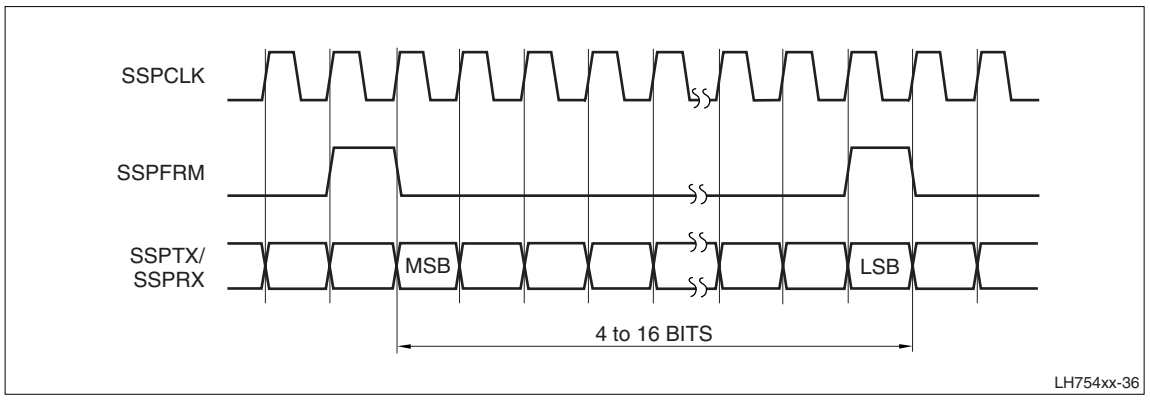


Figure 18-8. Texas Instruments Synchronous Serial Frame Format (Continuous Transfers)

18.3.3 National Semiconductor Frame Format

Unlike the full-duplex transmission capabilities that the other two frame formats support, the National Semiconductor Microwire format uses a special half-duplex, master-slave messaging technique. In this mode:

1. When a frame begins, an 8-bit control message is transmitted to the off-chip slave. During this transmission, the SSP does not receive incoming data.
2. After the message is sent, the off-chip slave decodes it, waits one serial clock after the last bit of the 8-bit control message has been sent, and responds with the requested data. The returned data can be 4 to 16 bits long, making the total frame 13 to 25 bits long.

During reception, data goes through a serial-to-parallel conversion before being placed into the receive FIFO. The data is then read out via the AMBA APB interface.

Figure 18-9 shows the National Semiconductor Microwire format for a single transfer. Figure 18-10 shows this format for continuous transfers.

18.4 Clock Generation

The serial bit rate is derived by dividing down the system clock, HCLK. The clock is first divided by an even prescale value, CPSDVSR, from 2 to 254, which is programmed in the CPSR Register. The clock is further divided by a value from 1 to 256, which is SCR + 1 (where SCR is the value programmed in the CTRL0 Register). The frequency of the output clock SSPCLK is defined as fSSPCLK ÷ (CPSDVSR × (1 + SCR)).

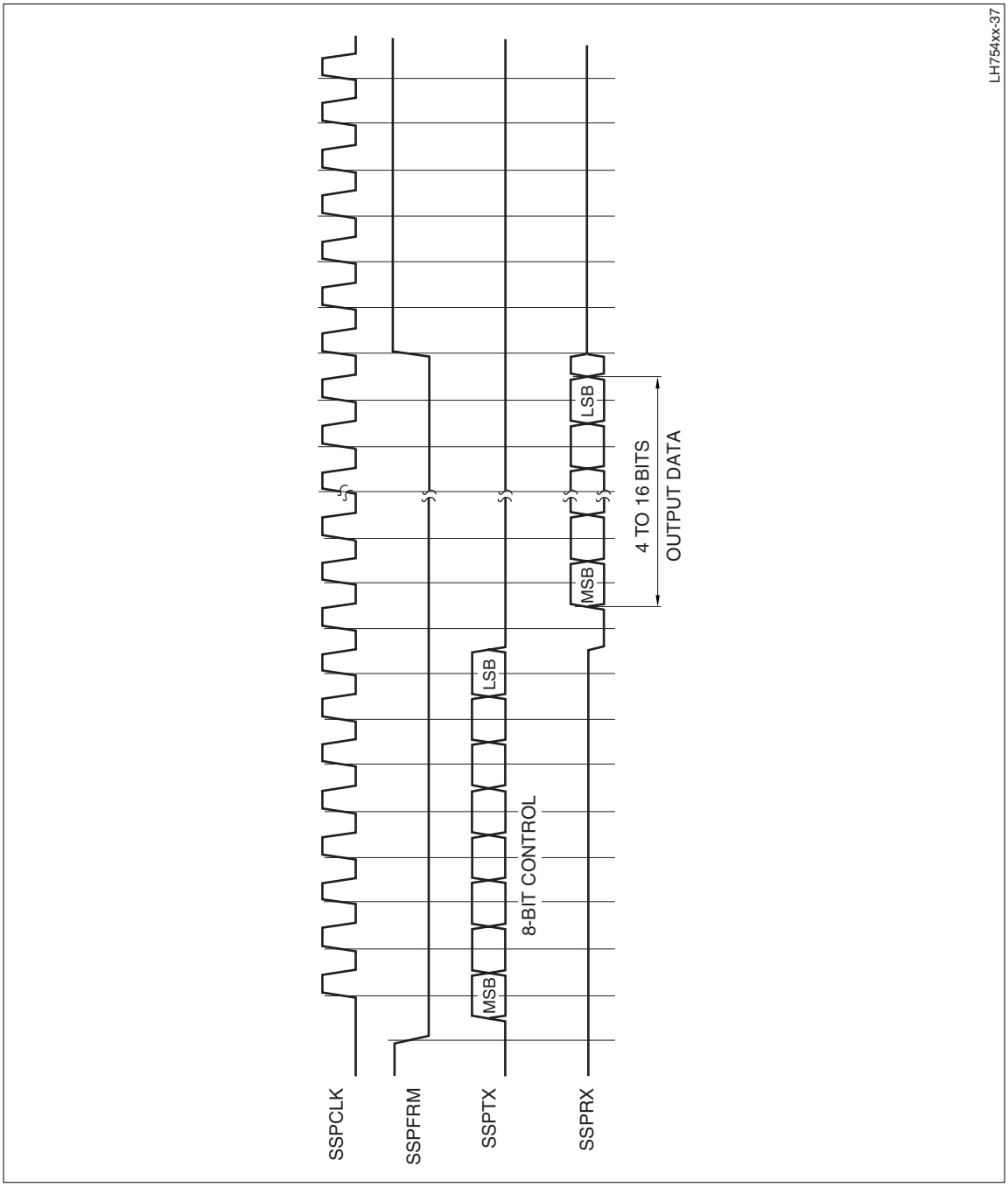


Figure 18-9. Microwire Frame Format (Single Transfer)

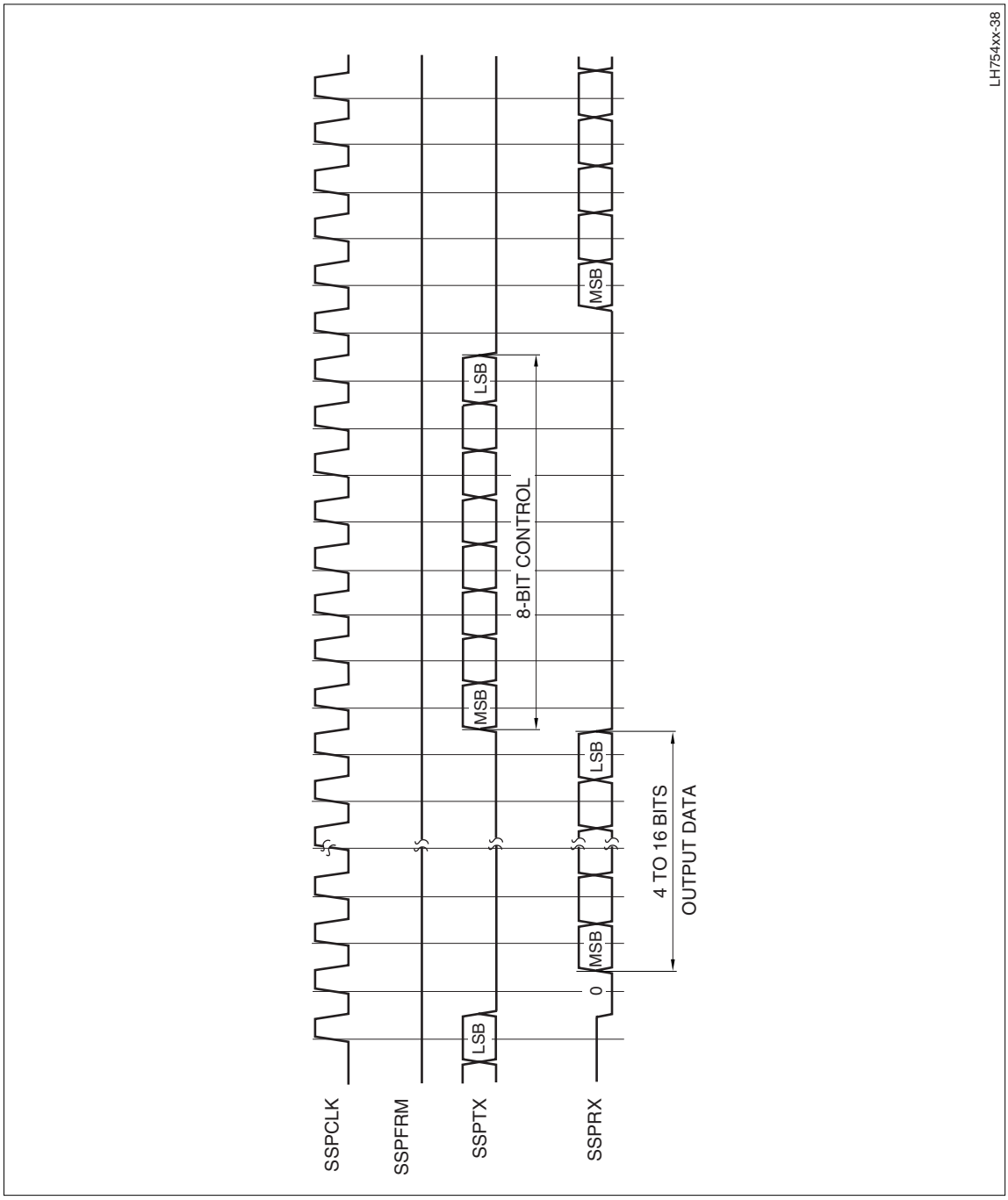


Figure 18-10. Microwire Frame Format (Continuous Transfers)

18.5 SSP Programmer's Model

The base address for the SSP is:

SSP Base Address: 0xFFFC6000

Locations at offsets 0x01C through 0xFFF are reserved and must not be used during normal operation.

18.5.1 SSP Register Summary

Table 18-2. Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL0	0x000	RW	0x0000	Control Register 0
CTRL1	0x004	RW	0x0000	Control Register 1
DR	0x008	RW	0x0000	Receive FIFO (Read)/Transmit FIFO (Write)
SR	0x00C	R	0x03	Status Register
CPSR	0x010	RW	0x00	Clock Prescale Register
IIR/ ICR	0x014	RW	0x0	Interrupt Identification Register (read) / Interrupt Clear Register (write)
RXTO	0x018	RW	0x000	Receive Timeout Register
///	0x01C - 0xFFF			Reserved

18.5.2 SSP Register Definitions

18.5.2.1 Control Register 0

CTRL0 is Control Register 0. CTRL0 contains five bit fields that control various SSP functions. The active bits used in this register are Read/Write.

Table 18-3. CTRL0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SCR								SPH	SPO	FRF		DSS			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC6000 + 0x000															

Table 18-4. CTRL0 Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:8	SCR	**Serial Clock Rate** SCR generates the SSP's transmit and receive bit rates. Bit rate = fSSPCLK ÷ (CPSDVSR × (1 + SCR)) CPSDVSR is an even value from 2 to 254 and SCR is a value from 0 - 255.
7	SPH	**SSPCLK Phase** Applicable to Motorola SPI frame format only.
6	SPO	**SSPCLK Polarity** Applicable to Motorola SPI frame format only.
5:4	FRF	**Frame Format** 00 = Motorola SPI frame format 01 = TI synchronous serial frame format 10 = National Microwire frame format 11 = Reserved, undefined operation
3:0	DSS	**Data Size Select** 0000 = Reserved, undefined operation 0001 = Reserved, undefined operation 0010 = Reserved, undefined operation 0011 = 4-bit data 0100 = 5-bit data 0101 = 6-bit data 0110 = 7-bit data 0111 = 8-bit data 1000 = 9-bit data 1001 = 10-bit data 1010 = 11-bit data 1011 = 12-bit data 1100 = 13-bit data 1101 = 14-bit data 1110 = 15-bit data 1111 = 16 bit data

18.5.2.2 Control Register 1

CTRL1 is the Control Register 1. CTRL1 contains five bit fields that control various SSP functions. The active bits used in this register are Read/Write.

Table 18-5. CTRL1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											SSE	LBM	RORIE	TIE	RIE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC6000 + 0x004															

Table 18-6. CTRL1 Register Definitions

BITS	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	SSE	**Synchronous Serial Port Enable** 0 = SSP operation is disabled. 1 = SSP operation is enabled.
3	LBM	**Loopback Mode** 0 = Normal serial port operation enabled. 1 = Output of transmit serial shifter is connected to input of receive serial shifter internally.
2	RORIE	**Receive FIFO Overrun Interrupt Enable** 0 = Overrun detection is disabled. Overrun condition does not generate the SSPRORINTR interrupt. Clearing this bit to zero also clears the SSPROINTER if it is already asserted. 1 = Overrun detection is enabled. Overrun condition generates the SSPRORINTR interrupt.
1	TIE	**Transmit FIFO Interrupt Enable** 0 = Transmit FIFO half-full or less condition does not generate the SSPTXINTR interrupt. 1 = Transmit FIFO half-full or less condition generates the SSPTXINTR interrupt.
0	RIE	**Receive FIFO Interrupt Enable** 0 = Receive FIFO half-full-or-more condition does not generate the SSPRXINTR interrupt. 1 = Receive FIFO half-full-or-more condition generates the SSPRXINTR interrupt.

18.5.2.3 Receive / Transmit FIFO Register

DR is the 16-bit-wide Receive / Transmit FIFO register. The active The active bits used in this register are Read/Write.

- When DR is read, the entry in the receive FIFO (pointed to by the current FIFO read pointer) is accessed. As data values are removed by the SSP's receive logic from the incoming data frame, they are placed into the entry in the receive FIFO (pointed to by the current FIFO write pointer).
- When DR is written to, the entry in the transmit FIFO (pointed to by the write pointer), is written to. Data values are removed from the transmit FIFO one value at a time by the transmit logic. Each value is loaded into the transmit serial shifter, then serially shifted out onto the SSPTX pin at the programmed bit rate.

When a data size of less than 16 bits is selected, programmers must right-justify data written to the transmit FIFO. The transmit logic ignores the unused bits. Received data less than 16 bits is automatically right-justified in the receive buffer.

When the SSP is programmed for National Microwire frame format, the default size for transmit data is eight bits (the most-significant byte is ignored). The receive data size is controlled by the programmer. The transmit FIFO and the receive FIFO are not cleared, even when the SSE bit in Control Register 1 is set to zero (refer to Section 18.5.2.2). This allows the software to fill the transmit FIFO before enabling the SSP.

When using Texas Instruments mode with data lengths smaller than 16 bits, the most-significant bits (MSB[s]) that exceed the data length are undefined. As a result, all bits greater than the length of the data should be masked.

Table 18-7. DR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	DATA															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC6000 + 0x008															

Table 18-8. DR Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	DATA	**Transmit/Receive FIFO** Read = Receive FIFO Write = Transmit FIFO Right-justify data when the SSP is programmed for a data size that is less than 16 bits. Unused bits at the top are ignored by transmit logic. The receive logic automatically right-justifies.

18.5.2.4 Status Register

SR is the Status Register. This register contains bits that indicate the FIFO fill status and the SSP busy status.

Table 18-9. SR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											BSY	RFF	RNE	TNF	TFE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC6000 + 0x00C															

Table 18-10. SR Register Definitions

BITS	NAME	DESCRIPTION
31:5	///	**Reserved** Reads as 0.
4	BSY	**SSP Busy Flag** 0 = SSP is idle. 1 = SSP is transmitting and/or receiving a frame or the transmit FIFO is non-empty.
3	RFF	**Receive FIFO Full** 0 = Receive FIFO is not full. 1 = Receive FIFO is full.
2	RNE	**Receive FIFO Not Empty** 0 = Receive FIFO is empty. 1 = Receive FIFO is not empty.
1	TNF	**Transmit FIFO Not Full** 0 = Transmit FIFO is full. 1 = Transmit FIFO is not full.
0	TFE	**Transmit FIFO Empty** 0 = Transmit FIFO is not empty. 1 = Transmit FIFO is empty.

18.5.2.5 Clock Prescale Register

CPSR is the Clock Prescale Register. The CPSR Register specifies the division factor by which the input HCLK should be internally divided before further use. The active bits used in this register are Read/Write.

The value programmed into this register is a value from 2 to 254. This register defaults to zero after Reset.

Table 18-11. CPSR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								CPSDVSR							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	RW	RW	RW	RW	RW	RW	RW	R
ADDR	0xFFFC6000 + 0x010															

Table 18-12. CPSR Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:8	///	**Reserved** Reads undefined. Write as 0.
7:0	CPSDVSR	**Clock Prescale Divisor** An even number from 2 to 254, depending on the frequency of SSPCLK. The least significant bit is hard-coded to zero. It may be written, but always will return zero.

18.5.2.6 Interrupt Identification/Clear Register

IIR is the Interrupt Identification Register. The interrupt status is read from this register.

ICR is the Interrupt Clear Register. A write of any value to this register clears the SSP receive FIFO Overrun Interrupt. This interrupt-clearing mechanism is in addition to the other mechanism in the CTRL1 Register. Therefore, clearing the RORIE bit in the CTRL1 Register also clears the overrun condition if already asserted. All the bits clear to zero after a System Reset.

Table 18-13. IIR/ICR Register (Read Characteristic)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///													RORIS	TIS	RIS
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC6000 + 0x014															

Table 18-14. IIR/ICR Register Definitions (Read Operation)

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Reads 0.
15:3	///	**Reserved** Reads undefined.
2	RORIS	**Read SSP Receive FIFO Overrun Interrupt Status** 0 = SSPRORINTR is not asserted. 1 = SSPRORINTR is asserted.
1	TIS	**Read SSP Transmit FIFO Service Request Interrupt Status** 0 = SSPTXINTR is not asserted. 1 = SSPTXINTR is asserted.
0	RIS	**Read SSP Receive FIFO Service Request Interrupt Status** 0 = SSPRXINTR is not asserted. 1 = SSPRXINTR is asserted.

Table 18-15. IIR/ICR Register (Write Characteristic)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFC6000 + 0x014															

Table 18-16. IIR/ICR Register Definitions (Write Characteristic)

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	///	**Clear Receive Overrun Interrupt** A write to these bits clears the Receive Overrun Interrupt, regardless of the data value written.

18.5.2.7 Receive Timeout Register

RXTO is the Receive Timeout Register. This register specifies the number of system clock cycles that must elapse before the receive timeout. The active bits used in this register are Read/Write.

Table 18-17. RXTO Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RXTO															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC6000 + 0x018															

Table 18-18. RXTO Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	RXTO	**HCLK Wait Periods** If 0x0001 ≤ RXTO ≤ 0xFFFF, then RXTO contains the number of HCLK periods that must elapse from the time the SSP Receive Interrupt has been cleared after a previous data transfer without another SSP Receive Interrupt before the Receive Timeout Interrupt is asserted. RXTO = 0x0000; disables the Rx Timeout function. Writing any value to this register clears the current Receive Overrun Interrupt.

18.5.3 SSP Interrupts

The SSP can assert four interrupts:

- SSPRXINTR — SSP Receive FIFO Service Interrupt request, locally maskable
- SSPTXINTR — SSP Transmit FIFO Service Interrupt request, locally maskable
- SSPRORINTR — SSP Receive Overrun Interrupt request, locally maskable
- SSPRXTOINTR — SSP Receive FIFO Timeout Interrupt request.

The first three interrupts can be enabled or disabled by changing the mask bits in Control Register 1. Setting the appropriate mask bit HIGH enables the interrupt. The fourth interrupt, SSPRXTOINTR SSP, is not locally maskable and should be masked using the Vectored Interrupt Controller IntEnable Register (see Section 10.2.2.5). All four interrupts are combined into a single interrupt: SSPINTR.

If the Receive Timeout function is not required, disable it by writing the value 0x0000 to the RXTO Register. Writing any value to this register clears the current SSPRXTOINTR interrupt.

Provision of the individual outputs as well as a combined interrupt output, allows use of either a global ISR or modular device drivers to handle interrupts from the SSP.

The status of the four individual interrupt sources can be read from the IIR Register.

18.5.3.1 Receive Interrupt

SSPRXINTR is the Receive Interrupt. This interrupt is asserted when there are four or more valid entries in the receive FIFO. The interrupt is cleared by reading the receive FIFO until there are three or fewer entries.

18.5.3.2 Transmit Interrupt

SSPTXINTR is the Transmit Interrupt. This interrupt is asserted when the FIFO is less than or equal to half full (when there is space for four or more entries). The interrupt is cleared when there are five or more entries in the transmit FIFO.

This interrupt is not qualified with the Synchronous Serial Port Enable bit (bit [4]) in Control Register 1, 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 ISR. For more information about Control Register 1, see Section 18.5.2.2.

NOTE: The SSPTXINTR interrupt is always set if the Synchronous Serial Port Enable bit in Control Register 1 is not set.

18.5.3.3 Receive Overrun Interrupt

SSPRORINTR is the Receive Overrun Interrupt. This interrupt is asserted when the FIFO is already full and an additional data frame is received, causing an overrun of the FIFO. Data is over-written in the Shift Register, but not the FIFO.

18.5.3.4 Receive Timeout Interrupt

SSPRXTOINTR is the Receive Timeout Interrupt. This interrupt is asserted if the receive FIFO does not generate a further service request interrupt (SSPRXINTR) within the number of HCLK periods programmed in the RXTO Register.

18.5.3.5 SSPINTR

The SSPRXINTR, SSPTXINTR, SSPRORINTR, and SSPRXTOINTR interrupts are also combined into the single output SSPINTR. This interrupt is an OR function of the individual interrupt sources. This output can be connected to the system interrupt controller to provide another level of masking on a per-peripheral basis.

The combined SSP Interrupt is asserted if any of the four individual interrupts is asserted and enabled.

Chapter 19
UART0 and UART1

UART0 and UART1 offer similar internal functionality to the industry-standard 16C550. They perform serial-to-parallel conversion on data received from a peripheral device and parallel-to-serial conversion on data transmitted to the UART. The CPU reads and writes data and control/status information through the AMBA APB interface. The transmit and receive paths are buffered with internal FIFO memories that support programmable-service 'trigger levels' and overrun protection. These FIFO memories enable up to 16 bytes to be stored independently in both transmit and receive modes. FIFO depth is 16. If a FIFO is disabled, a 1-byte holding register is used.

Figure 19-1 shows a block diagram of UART0 and UART1.

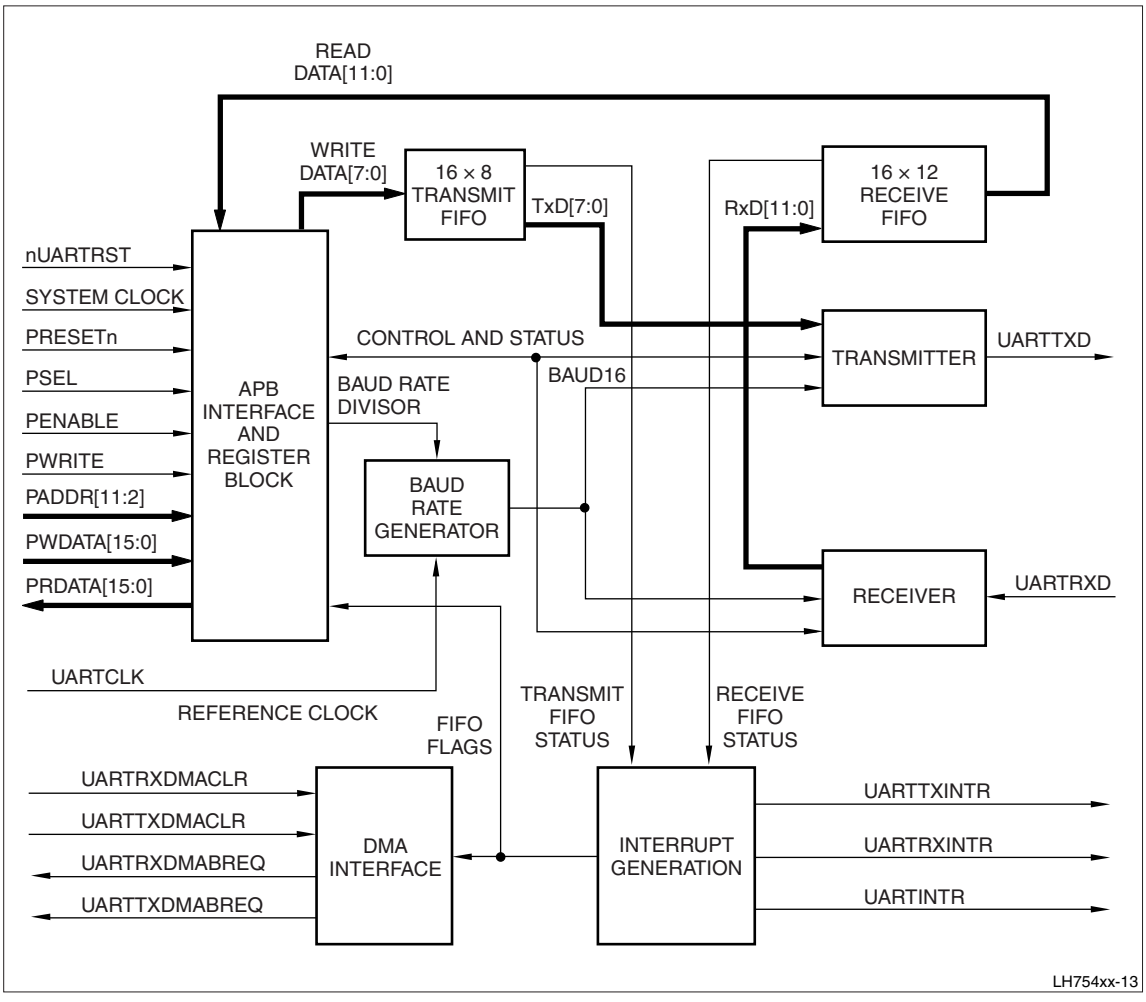


Figure 19-1. UART0 and UART1 Block Diagram

19.1 UART0 and UART1 Features

UART0 and UART1 provide the following features:

- Supports baud rates up to 921,600 baud (given an external crystal frequency of 14.756 MHz)
- Support for 5, 6, 7, or 8 data bits per character
- Even, odd, stick, or no-parity bit generation and detection
- 1 or 2 stop-bit generation
- Full-duplex operation
- Separate transmit and receive FIFOs, with programmable-service 'trigger levels' (1/8, 1/4, 1/2, 3/4, and 7/8) and overrun protection
- Programmable baud-rate generator that enables the UART input clock to be divided by 16 to 65535x16. It also generates an internal clock that is common to both transmit and receive portions of the UART. The divisor can be a fractional number.
- Support for Direct Memory Access (DMA)
- Support for generating and detecting breaks during UART transactions
- Loopback testing.

A UART clock that can operate asynchronously or synchronously with the system clock.

19.2 UART0 and UART1 Theory of Operation

All UART Control and Status Registers can be accessed through the APB. During transmission, data writes to the transmit FIFO through the APB interface. When data writes to the transmit FIFO:

- The UART causes the data frame to start transmitting with the parameters indicated in register LCR_H if the UART is enabled. Data continues to transmit until no data remains in the transmit FIFO.
- The BUSY bit in the FR Register is asserted. This bit remains HIGH until the transmit FIFO is empty and the last character, including the stop bits, has been sent.

19.2.1 UART0 and UART1 Receiver Data Frame

A UART receiver data frame has the following structure:

- A start bit that indicates the beginning of the frame. This bit consists of a 0 on the receiver input for one bit period.
- Data, which consists of five to eight data bits.
- An optional parity bit, which can be used with available hardware for parity-error checking.
- One or more stop bits, which indicate the end of the frame. Stop bits consist of a 1 on the receiver input for as many bit periods as the number of stop bits specified when programming the receiver.

The UART receiver is in the marking state (i.e., the input is 1) from the time a stop bit is sent until the time the next start bit is received. When the receiver receives an entire frame, the UART transfers the received data and the frame status to the receiver FIFO. The receive FIFO is a 12-bit-wide, 16-word-deep FIFO memory buffer, or a 1-byte-deep holding register.

The start bit works with the UART bit clock to synchronize the receiver with the source driving the receiver. When the source drives the receiver input from the idle state to 0, the receiver waits 7/16ths of a bit period, then samples the input three times:

- Once at 7/16ths of the bit period
- Once at 8/16ths of the bit period
- Once at 9/16ths of the bit period.

If the input is 0 for at least two of the three samples, the UART recognizes a start bit. After recognizing the start bit, the receiver repeats the following sequence until all data bits, any parity bit, and all stop bits are detected:

1. Wait 14/16ths of a bit period, then sample the input.
2. Wait 1/16th of a bit period, then sample the input.
3. Wait 1/16th of a bit period, then sample the input.
4. Choose the majority value of the three samples as the input value for that bit period.

After recognizing the final stop bit, the UART stores the received data frames and associated status bits in the receiver FIFO.

19.2.2 Status Conditions

UARTs 0 and 1 adhere to the following status conditions:

- If a UART fails to detect a 1 for all programmed stop-bit periods following a data frame, the UART sets the framing-error status for that frame.
- Enabling parity-error detection causes the UART to compare the parity bit in each frame with the parity required for the hardware. The UART sets the parity-error status for each frame containing a parity error.
- A line break is 0 for all bits (start bit, data bits, parity bit, and stop bits). The UART sets the line-break status for each frame containing a line break.
- The overrun-status bit indicates some frames might have been lost immediately preceding the frame with the overrun status.

The overrun status is announced by both the overrun-error bit in the RSR Register and an overrun-status bit in a frame in the receiver FIFO. If the receiver FIFO is full and another frame is received, the receiver enters the overrun state and the UART sets the overrun-error bit in the RSR. The overrun-error bit remains set until a 1 writes to the overrun-error bit in the ECR Register.

While the receiver FIFO remains full, additional data frames at the receiver unit are lost and are not stored in the receiver FIFO. When the UART can resume storing frames in the receiver FIFO, the receiver exits the overrun state. The overrun-status bit in the receiver FIFO is set in the first frame stored after the overrun.

19.2.3 On-Chip DMA Capabilities

UARTs 0 and 1 can be programmed to utilize the on-chip DMA to reduce processor bandwidth required to service UART activities. DMA functions support burst transfers on the receive channel, transmission channel, or both.

When UART FIFOs are enabled, the UARTs will generate a DMA request to the DMA controller when the number of characters in the FIFO crosses the watermark programmed in the UART. If the number of characters to be transmitted or received is lesser than the currently programmed watermark, DMA transfers are not initiated and those characters are not transmitted or copied into the receive buffer in memory. Therefore, it is highly recommended that while performing DMA transfers, FIFOs be turned off and the UART be operated in single character mode.

- When DMA is enabled on the receive channel, a DMA request is issued when the receive FIFO reaches its programmed high water mark. Once the DMA block services the request, a new one is issued when the FIFO fills above its high water mark.
- When DMA is enabled on the transmit channel, a request is issued when the transmit channel FIFO falls below its low water mark. The request is reissued if the FIFO remains below that level when the DMA request has been serviced, or the next time that the FIFO falls below that level.

DMA requests are masked when the UART issues an error interrupt. If the UART is in the FIFO Disabled Mode, only the DMA Single Transfer Mode can operate, since only one character can be transferred to or from the FIFOs at any time. As a result, the programmed water mark level is not relevant in the FIFO Disabled Mode.

19.2.4 Programming Control Registers

The UART must be disabled before any of the Control Registers are programmed. When the UART is disabled in the middle of transmission or reception, it completes the operation on the current character before stopping.

19.3 UART0 and UART1 Programmer's Model

While UART0 and UART1 offer similar internal functionality to the industry standard 16C550, the programmer's interface differs. That information is covered here.

The base address for UART0 is:

UART0 Base Address: 0xFFFC0000

The base address for UART1 is:

UART1 Base Address: 0xFFFC1000

The following locations are reserved and must not be used during normal operation:

- 0xFFFC0008 through 0xFFFC0014
- 0xFFFC001C through 0xFFFC0020
- 0xFFFC004C through 0xFFFC007C
- 0xFFFC0080 through 0xFFFC008C
- 0xFFFC0090 through 0xFFFC0FFC

19.3.0.1 UART0 and UART1 Register Summary

Table 19-1. UART0 and UART1 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
DR	0x000	RW	0x---	Data read or written from the interface. It is 12 bits wide on a read and 8 on a write.
RSR/ECR	0x004	RW	0x0	Receive Status Register (read)/Error Clear Register (write)
///	0x008 - 0x014			Reserved
FR	0x018	R	0x00000090	Flag Register (read only)
///	0x01C - 0x020			Reserved
IBRD	0x024	R	0x0000	Integer Baud Rate Divisor Register
FBRD	0x028	RW	0x00	Fractional Baud Rate Divisor Register
LCTRL_H	0x02C	RW	0x00	Line Control Register, HIGH byte
CTRL	0x030	RW	0x0300	UART Control Register
IFLS	0x034	RW	0x12	Interrupt FIFO Level Select Register
IMSC	0x038	RW	0x000	Interrupt Mask Set/Clear Register
RIS	0x03C	R	0x00-	Raw Interrupt Status Register
MIS	0x040	R	0x00-	Masked Interrupt Status Register
ICR	0x044	W		Interrupt Clear Register
DMACTRL	0x048	RW	0x00	DMA Control Register
///	0x04C - 0x07C			Reserved
///	0x080 - 0x08C			Reserved
///	0x090 - 0xFFC			Reserved

19.3.1 UART0 and UART1 Register Definitions

19.3.1.1 Data Register

DR is the Data Register for words that are to be transmitted or have been received over the serial interface. Writing to this register initiates transmission from the UART.

- If the FIFOs are enabled, data written to this location is pushed onto the transmit FIFO.
- If the FIFOs are not enabled, data is stored in the Transmitter Holding Register (the bottom word of the transmit FIFO).

A read to this register pops the first word from the receive FIFO. This word consists of the received character and the associated error bits.

Table 19-2. DR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///				OVERRUN ERROR	BREAK ERROR	PARITY ERROR	FRAMING ERROR	DATA							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	UART0: 0xFFFC0000 + 0x000 UART1: 0xFFFC1000 + 0x000															

Table 19-3. DR Register Definitions

BIT	NAME	DESCRIPTION
31:12	///	**Reserved** Write the reset value.
11	OVERRUN ERROR	**Receive FIFO Full/Empty** 0 = There is an empty space in the FIFO and a new character can be written to it. 1 = Data is received and the receive FIFO is already full.
10	BREAK ERROR	**Break Error** 1 = A break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). This bit is cleared to 0 after a write to ECR. In FIFO Mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state) and the next valid start bit is received.
9	PARITY ERROR	**Parity Error** 1 = The parity of the received data character does not match the parity selected as defined by bits [2] and [7] of the LCR_H Register (see Section 19.3.1.9). In FIFO Mode, this error is associated with the character at the top of the FIFO.
8	FRAMING ERROR	**Framing Error** 1 = The received character did not have a valid stop bit (a valid stop bit is 1). In FIFO Mode, this error is associated with the character at the top of the FIFO. Framing errors are not generated for the number of stop bits; e.g.: when one is expected and two are recieved, or two are expected and one is recieved.
7:0	DATA	**Receive/Transmit Data Character** Read = Receive data character. Write = Transmit data character.

19.3.1.2 Receive Status/Error Clear Register

RSR/ECR is the Receive Status Register/ Error Clear Register.

If the status is read from this register, the status bits in this register correspond to the status bits of the last word read from the DR Register. The status information for overrun is set immediately when an overrun condition occurs.

A write to the ECR Register clears the framing, parity, break, and overrun errors. All bits clear to '0' on System Reset.

NOTE: The received data character must be read first from the DR Register before reading the error status associated with that data character from the RSR Register. This read sequence cannot be reversed because the RSR Register is updated only when a read occurs from the DR Register. However, the status information can also be obtained by reading the DR Register.

Table 19-4 and Table 19-5 describe the RSR/ECR Register for write operations.

Table 19-4. RSR/ECR Register (Write Operations)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								ERROR CLEAR							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W
ADDR	UART0: 0xFFFC0000 + 0x004 UART1: 0xFFFC1000 + 0x004															

Table 19-5. RSR/ECR Register Definitions (Write Operations)

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	ERROR CLEAR	**Error Clear1** A write to this register clears the framing, parity, break, and overrun errors. The data value is not important.

Table 19-6 and Table 19-7 describe the RSR/ECR Register for read operations.

Table 19-6. RSR/ECR Register (Read Operations)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///												OVERRUN ERROR	BREAK ERROR	PARITY ERROR	FRAMING ERROR
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	UART0: 0xFFFC0000 + 0x004 UART1: 0xFFFC1000 + 0x004															

Table 19-7. RSR/ECR Register Definitions (Read Operations)

BITS	NAME	DESCRIPTION
31:4	///	**Reserved** Write the reset value.
3	OVERRUN ERROR	**Data Overrun Error** 1 = Data is received and the FIFO is already full. This bit is cleared to 0 by a write to ECR. The FIFO contents remain valid since no further data is written when the FIFO is full, only the contents of the shift register are overwritten. The CPU must now read the data in order to empty the FIFO.
2	BREAK ERROR	**Break Error** 1 = A break condition was detected, indicating that the received data input was held LOW for longer than a full-word transmission time (defined as start, data, parity, and stop bits). This bit is cleared to 0 after a write to ECR. In FIFO Mode, this error is associated with the character at the top of the FIFO. When a break occurs, only one 0 character is loaded into the FIFO. The next character is only enabled after the receive data input goes to a 1 (marking state) and the next valid start bit is received.
1	PARITY ERROR	**Parity Error** 1 = The parity of the received data character does not match the parity selected as defined by bits [2] and [7] of the LCR_H Register (see Section 19.3.1.7). This bit is cleared to 0 by a write to ECR.
0	FRAMING ERROR	**Framing Error** 1 = The received character did not have a valid stop bit (a valid stop bit is 1). This bit is cleared to 0 by a write to ECR.

19.3.1.3 Flag Register

FR is the Flag Register. After System Reset, TXFF, RXFF, and BUSY are '0', and TXFE and RXFE are '1'.

Table 19-8. FR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///				///	///	///		///							
RESET	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								TRANSMIT FIFO EMPTY	RECIEVE FIFO FULL	TRANSMIT FIFO FULL	RECIEVE FIFO EMPTY	UART BUSY	///		
RESET	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	UART0: 0xFFFC0000 + 0x018 UART1: 0xFFFC1000 + 0x018															

Table 19-9. FR Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	TRANSMIT FIFO EMPTY	**Transmit FIFO Empty** The meaning of this bit depends on the state of the FEN bit in the LCR_H Register (see Section 19-16). FIFO disabled = This bit is set when the Transmit Holding Register is empty. FIFO enabled = The TXFE bit is set when the transmit FIFO is empty.
6	RECEIVE FIFO FULL	**Receive FIFO Full** The meaning of this bit depends on the state of the FEN bit in the LCR_H Register (see Section 19-16). FIFO disabled = This bit is set when the Receive Holding Register is full. FIFO enabled = RXFF bit is set when the receive FIFO is full.
5	TRANSMIT FIFO FULL	**Transmit FIFO Full** The meaning of this bit depends on the state of the FEN bit in the LCR_H Register (see Section 19.3.1.9). FIFO disabled = This bit is set when the Transmit Holding Register is full. FIFO enabled = The TXFF bit is set when the transmit FIFO is full.
4	RECEIVE FIFO EMPTY	**Receive FIFO Empty** The meaning of this bit depends on the state of the FEN bit in the LCR_H Register (see Section 19.3.1.9). FIFO disabled = This bit is set when the Receive Holding Register is empty. FIFO enabled = The RXFE bit is set when the receive FIFO is empty
3	UART BUSY	**UART Busy** 1 = UART is busy transmitting data. This bit remains set until the complete byte, including all stop bits, has been sent from the Shift Register. This bit is set as soon as the transmit FIFO becomes non-empty (regardless of whether the UART is enabled).
2:0	///	**Reserved** Write the reset value.

19.3.1.4 UART Line Control Register

The LCTRL Register is a single 30-bit register formed from three registers in the address map:

- LCTRL_H
- IBRD
- FBRD.

The 30 bits of LCR are updated when LCR_H is written. Table 19-10 shows how to update the contents of the register.

Table 19-10. Updating Register Contents

TO UPDATE	PERFORM
The IBRD or FBRD Register	A LCR_H write operation at the end.
Both the IBRD and FBRD Registers	Write to each register's location in either order, then write to LCR_H.
Either the IBRD or FBRD Register	Write only to the desired register, then write to LCR_H.

The following UART clock frequency must be selected:

$f_{UARTCLK}$ (MIN.) $\geq 16 \times$ baud_rate (MAX.)

$f_{UARTCLK}$ (MAX.) $\leq 16 \times 65535 \times$ baud_rate (MIN.)

This frequency must be within the required error limits for all baud rates to be used and must not be more than 5/3 times faster than the frequency of the system clock.

19.3.1.5 Integer Baud Rate Divisor Register

IBRD is the integer part of the baud rate divisor value. The active bits used in this register are Read/Write. All of the bits in this register clear to '0' on System Reset.

Table 19-11. IBRD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	BAUD RATE INTEGER															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	UART0: 0xFFFC0000 + 0x024 UART1: 0xFFFC1000 + 0x024															

Table 19-12. IBRD Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:0	BAUD RATE INTEGER	**Integer Baud Rate Divisor** This value is used with the Fractional Baud Rate Divisor to ascertain the baud rate for the UART. For information about calculating this divisor value, see Section 19.3.1.7. For information about the Fractional Baud Rate Divisor, see Section 19.3.1.6. These bits are cleared to '0' on Reset.

19.3.1.6 Fractional Baud Rate Divisor Register

FBRD is the fractional part of the baud rate divisor value. The active bits used in this register are Read/Write. All the bits are cleared to '0' on System Reset. All the bits are cleared to '0' on System Reset.

Table 19-13. FBRD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										BAUD RATE FUNCTION					
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	UART0: 0xFFFC0000 + 0x028 UART1: 0xFFFC1000 + 0x028															

Table 19-14. FBRD Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
5:0	BAUD RATE FUNCTION	**Fractional Baud Rate Divisor** This value is used with the Integer Baud Rate Divisor to ascertain the baud rate for the UART. For information about calculating this divisor value, see Section 19.3.1.7. For information about the Integer Baud Rate Divisor, see Section 19.3.1.5. These bits are cleared to '0' on Reset.

19.3.1.7 Calculating the Divisor Value

The following example shows how to clear a divisor value. This example assumes that the required baud rate is 230,400 and the UARTCLK = 4 MHz.

1. Baud Rate Divisor = $(4 \times 10^6) \div (16 \times 230{,}400) = 1.085$
2. BRDt and BRDf = 0.085
3. Fractional part, m = integer $((0.085 \times 64) + 0.5) = 5$
4. Generated baud rate divider = 1 + 5/64 = 1.078
5. Generated baud rate = $(4 \times 10^6) \div (16 \times 1.078) = 231{,}911$
6. Error = $((231{,}911 - 230{,}400) \div 230{,}400) \times 100 = 0.656\%$

The maximum error using a 6-bit FBRD Register = 1/64 × 100 = 1.56%. This occurs when m = 1 and the error is cumulative over 64 clock ticks.

19.3.1.8 Typical Bit Rates and Their Corresponding Divisor

Table 19-15 shows some typical bit rates and their corresponding divisor, given the UART Clock Frequency of 14.7456 MHz.

Table 19-15. Bit Rates and Their Corresponding Divisors

UART CLK (MHz)	BAUD RATE	INTEGER DIVISOR (IBRD)	FRACTIONAL DIVISOR (FBRD)
14.7456	921600	1	0
14.7456	460800	2	0
14.7456	230400	4	0
14.7456	153600	6	0
14.7456	115200	8	0
14.7456	76800	12	0

19.3.1.9 Line Control Register

LCTRL_H is the Line Control Register. The active bits used in this register are Read/Write.

This register accesses bits [29:22] of the UART LCTRL Register (see Section 19.3.1.4). The contents of the LCTRL_H Register are not updated until transmission or reception of the current character is complete. Table 19-16 is a truth table for the SPS, EPS, and PEN bits of the LCTRL_H Register.

Table 19-16. LCTRL_H Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								STICK PARITY SELECT	WORDLENGTH		ENABLE FIFOS	TWO STOP BITS SELECt	EVEN PARITY SELECT	PARITY ENABLE	SEND BREAK
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	UART0: 0xFFFC0000 + 0x02C UART1: 0xFFFC1000 + 0x02C															

Table 19-17. LCTRL_H Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	STICK PARITY SELECT	**Stick Parity Select** Bits [7], [2], and [1] work together to set up the parity. See Table 19-18.
6:5	WORD LENGTH	**Word Length** Indicates the number of data bits transmitted or received in a frame. 00 = 5 bits 01 = 6 bits 10 = 7 bits 11 = 8 bits
4	ENABLE FIFOS	**FIFO Enable Buffers** 1 = Enables transmit and receive FIFO buffers (FIFO Mode). When cleared to 0, the FIFOs are disabled (Character Mode); that is, the FIFOs become 1-byte-deep holding registers.
3	TWO STOP BITS SELECT	**Frame Stop Bits** 1 = Two stop bits are transmitted at the end of the frame. The receive logic always checks for received stop bits, regardless of whether there are one or two.
2	EVEN PARITY SELECT	**Even Parity Select** Bits [7], [2], and [1] work together to set up the parity. See Table 19-18.
1	PARITY ENABLE	**Parity Enable** Bits [7], [2], and [1] work together to set up the parity. See Table 19-18.
0	SEND BREAK	1 = A LOW level is continuously output on the UARTTXD output, after completing transmission of the current character. This bit must be asserted for at least one complete frame transmission time to generate a break condition. The transmit FIFO contents remain unaffected during a break condition. For normal use, this bit must be cleared to zero.

Table 19-18. Truth Table for bits [7], [2], and [1]

PARITY ENABLE (PEN)	EVEN PARITY SELECT (EPS)	STICK PARITY SELECT (SPS)	RESULTANT PARITY BIT (TRANSMITTED OR CHECKED)
0	X	X	Not transmitted or checked
1	1	0	Even parity
1	0	0	Odd parity
1	0	1	1
1	1	1	0

19.3.1.10 UART Control Register

CTRL is the UART Control Register. To enable transmission, bit [8] and bit [0] must be set. Similarly, to enable reception, bit [9] and bit [0] must be set. The active bits used in this register are Read/Write.

Table 19-19. CTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						RECEIVE ENABLE	TRANSMIT ENABLE	LC	///						UART ENABLE
RESET	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	UART0: 0xFFFC0000 + 0x030 UART1: 0xFFFC1000 + 0x030															

Table 19-20. CTRL Register Definitions

BIT	NAME	DESCRIPTION
31:10	///	**Reserved** Write the reset value.
9	RECEIVE ENABLE	**Receive Section** 1 = Enables the receive section of the UART. When the UART is disabled in the middle of reception, it completes the current character before stopping.
8	TRANSMIT ENABLE	**Transmit Section** 1 = Enables the transmit section of the UART. When the UART is disabled in the middle of transmission, it completes the current character before stopping.
7	LC	**Loopback Control Bit** Places UART0 and UART1 into Loopback Mode. 0 = Loopback is disabled. 1 = Loopback is enabled.
6:1	///	**Reserved** When writing to this register, perform a read-modified-write operation to this field.
0	UART ENABLE	**UART Enable** 1 = Enables the UART. When the UART is disabled in the middle of transmission or reception, it completes the current character before stopping.

19.3.1.11 Interrupt FIFO Level Select Register

IFLS is the Interrupt FIFO Level Select Register. The active bits used in this register are Read/Write.

The IFLS Register defines the FIFO level at which interrupts are generated to request service for the receive and transmit FIFOs. The interrupts are generated based on a transition through a level rather than being based on the level. That is, the design is such that the interrupts are generated when the fill level progresses through the trigger level. The bits are reset so that the trigger level is when the FIFOs are at the half-way mark.

Table 19-21. IFLS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										RECEIVE INTERRUPT FIFO LEVEL SELECT			TRANSMIT INTERRUPT FIFO LEVEL SELECT		
RESET	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW
ADDR	UART0: 0xFFFC0000 + 0x034 UART1: 0xFFFC1000 + 0x034															

Table 19-22. IFLS Register Definitions

BIT	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5:3	RECEIVE INTERRUPT FIFO LEVEL SELECT	**Trigger Points for the Receive Interrupt** 000 = Receive FIFO becomes ≥ 1/8 full. 001 = Receive FIFO becomes ≥ 1/4 full. 010 = Receive FIFO becomes ≥ 1/2 full. 011 = Receive FIFO becomes ≥ 3/4 full. 100 = Receive FIFO becomes ≥ 7/8 full. 101:111 = Reserved.
2:0	TRANSMIT INTERRUPT FIFO LEVEL SELECT	**Trigger Points for the Transmit Interrupt** 000 = Transmit FIFO becomes ≤ 1/8 full. 001 = Transmit FIFO becomes ≤ 1/4 full. 010 = Transmit FIFO becomes ≤ 1/2 full. 011 = Transmit FIFO becomes ≤ 3/4 full. 100 = Transmit FIFO becomes ≤ 7/8 full. 101:111 = Reserved.

19.3.1.12 Interrupt Mask Set/Clear Register

IMSC is the Interrupt Mask Set/Clear Register. The active bits used in this register are Read/Write.

On a read, this register returns the current value of the mask on the relevant interrupt. On a write of '1' to the particular bit, it enables the interrupt. A write of '0' masks the corresponding interrupt. All the bits are cleared to 0 following a System Reset.

Table 19-23. IMSC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///					OEIM	BEIM	PEIM	FEIM	RTIM	TXIM	RXIM	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	R	R	R	R
ADDR	UART0: 0xFFFC0000 + 0x038 UART1: 0xFFFC1000 + 0x038															

Table 19-24. IMSC Register Definitions

BITS	NAME	DESCRIPTION
31:11	///	**Reserved** Write the reset value.
10	OVERRUN ERROR INTERRUPT MASK	**Overrun Error Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the OEIM interrupt. When Read, returns the current mask for the OEIM interrupt.
9	BREAK ERROR INTERRUPT MASK	**Break Error Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the BEIM interrupt. When Read, returns the current mask for the BEIM interrupt.
8	PARITY ERROR INTERRUPT MASK	**Parity Error Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the PEIM interrupt. When Read, returns the current mask for the PEIM interrupt.
7	FRAMING ERROR INTERRUPT MASK	**Framing Error Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the FEIM interrupt. When Read, returns the current mask for the FEIM interrupt.

Table 19-24. IMSC Register Definitions (Cont'd)

BITS	NAME	DESCRIPTION
6	RECEIVE TIMEOUT INTERRUPT MASK	**Receive Timeout Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the RTIM interrupt. When Read, returns the current mask for the RTIM interrupt.
5	TRANSMIT INTERRUPT MASK	**Transmit Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the TXIM interrupt. When Read, returns the current mask for the TXIM interrupt.
4	RECEIVE INTERRUPT MASK	**Receive Interrupt Mask** Write values: 0 = Clears the mask. 1 = Sets the mask of the RXIM interrupt. When Read, returns the current mask for the RXIM interrupt.
3:0	///	**Reserved** Write the reset value.

19.3.1.13 Raw Interrupt Status Register

RIS is the Raw Interrupt Status Register. On a read, this register returns the current raw status value of the corresponding interrupt. A write has no effect.

Table 19-25. RIS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///					OVERRUN ERROR INTERRUPT STATUS	BREAK ERROR INTERRUPT STATUS	PARITY ERROR INTERRUPT STATUS	FRAMING ERROR INTERRUPTSTATUS	RECEIVE TIMEOUT INTERRUPT STATUS	TRANSMIT INTERRUPT STATUS	RECEIVE INTERRUPT STATUS	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	UART0: 0xFFFC0000 + 0x03C UART1: 0xFFFC1000 + 0x03C															

Table 19-26. RIS Register Definitions

BITS	NAME	DESCRIPTION
31:11	///	**Reserved** Writes have no effect.
10	OVERRUN ERROR INTERRUPT STATUS	**Raw Interrupt State** Specifies the raw interrupt state (prior to masking) of the UARTOEINTR interrupt.
9	BREAK INTERRUPT STATUS	**Break Interrupt Status** Specifies the raw interrupt state (prior to masking) of the UARTBEINTR interrupt.
8	PARITY ERROR INTERRUPT STATUS	**Parity Error Interrupt Status** Specifies the raw interrupt state (prior to masking) of the UARTPEINTR interrupt.
7	FRAMING ERROR INTERRUPT STATUS	**Framing Error Interrupt Status** Specifies the raw interrupt state (prior to masking) of the UARTFEINTR interrupt.
6	RECEIVE TIMEOUT INTERRUPT STATUS	**Receive Timeout Interrupt Status** Specifies the raw interrupt state (prior to masking) of the UARTRTINTR interrupt.
5	TRANSMIT INTERRUPT STATUS	**Transmit Interrupt Status** Specifies the raw interrupt state (prior to masking) of the UARTTXINTR interrupt.
4	RECEIVE INTERRUPT STATUS	**Receive Interrupt Status** Specifies the raw interrupt state (prior to masking) of the UARTRXINTR interrupt.
3:0	///	**Reserved** Writes have no effect.

19.3.1.14 Masked Interrupt Status Register

MIS is the Masked Interrupt Status Register. On a read, this register returns the current masked status value of the corresponding interrupt. A write has no effect.

Table 19-27. MIS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///					OVERRUN ERROR MASKED INTERRUPT STATUS	BREAK ERROR MASKED INTERRUPT STATUS	PARITY ERROR MASKED INTERRUPT STATUS	FRAMING ERROR MASKED INTERRUPT STATUS	RECEIVE TIMEOUT MASKED INTERRUPT STATUS	TRANSMIT MASKED INTERRUPT STATUS	RECEIVE MASKED INTERRUPT STATUS	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	UART0: 0xFFFC0000 + 0x040 UART1: 0xFFFC1000 + 0x040															

Table 19-28. MIS Register Definitions

BIT	NAME	DESCRIPTION
31:11	///	**Reserved** Writes have no effect.
10	OVERRUN ERROR MASKED INTERRUPT STATUS	**Overrun Error Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTOEINTR interrupt.
9	BREAK ERROR MASKED INTERRUPT STATUS	**Break Error Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTBEINTR interrupt.
8	PARITY ERROR MASKED INTERRUPT STATUS	**Parity Error Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTPEINTR interrupt.
7	FRAMING ERROR MASKED INTERRUPT STATUS	**Framing Error Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTFEINTR interrupt.
6	RECEIVE TIMEOUT MASKED INTERRUPT STATUS	**Receive Timeout Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTRTINTR interrupt.
5	TRANSMIT MASKED INTERRUPT STATUS	**Transmit Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTTXINTR interrupt.
4	RECEIVE MASKED INTERRUPT STATUS	**Receive Masked Interrupt Status** Specifies the masked interrupt state (after masking) of the UARTRXINTR interrupt.
3:0	///	**Reserved** Writes have no effect.

19.3.1.15 ICR

ICR is the Interrupt Clear Register. The active bits used in this register are Write Only. On a write of '1', the corresponding interrupt is cleared. A write of '0' has no effect.

Table 19-29. ICR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///					OVERRUN ERROR INTERRUPT CLEAR	BREAK ERROR INTERRUPT CLEAR	PARITY ERROR INTERRUPT CLEAR	FRAMING ERROR INTERRUPT CLEAR	RECEIVE TIMEOUT INTERRUPT CLEAR	TRANSMIT INTERRUPT CLEAR	RECEIVE INTERRUPT CLEAR	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	W	W	W	W	W	W	W	R	R	R	R
ADDR	UART0: 0xFFFC0000 + 0x044 UART1: 0xFFFC1000 + 0x044															

Table 19-30. ICR Register Definitions

BIT	NAME	DESCRIPTION
31:15	///	**Reserved** Write the reset value.
10	OVERRUN ERROR INTERRUPT CLEAR	**Overrun Error Interrupt Clear** Clears the UARTOEINTR interrupt.
9	BREAK ERROR INTERRUPT CLEAR	**Break Error Interrupt Clear** Clears the UARTBEINTR interrupt.
8	PARITY ERROR INTERRUPT CLEAR	**Parity Error Interrupt Clear** Clears the UARTPEINTR interrupt.
7	FRAMING ERROR INTERRUPT CLEAR	**Framing Error Interrupt Clear** Clears the UARTFEINTR interrupt.
6	RECEIVE TIMEOUT INTERRUPT CLEAR	**Receive Timeout Interrupt Clear** Clears the UARTRTINTR interrupt.
5	TRANSMIT INTERRUPT CLEAR	**Transmit Interrupt Clear** Clears the UARTTXINTR interrupt.
4	RECEIVE INTERRUPT CLEAR	**Receive Interrupt Clear** Clears the UARTRXINTR interrupt.
3:0	///	**Reserved** Write the reset value.

19.3.1.16 DMACTRL

DMACTRL is the DMA Control Register. The active bits used in this register are Read/ Write. All the bits are cleared to '0' on System Reset.

Table 19-31. DMACTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///													DMA ON ERROR	TRANSMIT DMA ENABLE	RECEIVE DMA ENABLE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W
ADDR	0xFFFC1000 + 0x048															

Table 19-32. DMACTRL Register Definitions

BIT	NAME	DESCRIPTION
31:3	///	**Reserved** Write the reset value.
2	DMA ON ERROR	**DMA on Error** 1 = Disables the DMA receive request output, UARTRXDMABREQ, when the UART Error Interrupt is asserted.
1	TRANSMIT DMA ENABLE	**Transmit DMA Enable** 1 = Enables the DMA for the transmit FIFO.
0	RECEIVE DMA ENABLE	**Receive DMA Enable** 1 = Enables the DMA for the receive FIFO.

19.3.2 UART0 and UART1 Interrupts

Both UART0 and UART1 have a combined interrupt. Only UART1 has separate UARTRXINTR and UARTTXINTR. The individual UART interrupt outputs are OR'd together to produce the combined interrupt for UART0 and UART1. However, UART1 has separate UARTRXINTR and UARTTXINTR interrupts. The individual UART interrupt outputs are OR'd together to produce the combined interrupt for that UART. Interrupt conditions within the combined interrupt are individually maskable.

The combined interrupt for UART1 has UARTRXINTR and UARTTXINTR, even though they are connected to the VIC. If using all three interrupts, exercise care when assigning the priorities in the VIC.

19.3.2.1 UARTRXINTR

UARTRXINTR is the receive interrupt. This interrupt changes state when one of the FIFO events in Table 19-33 occurs:

Table 19-33. UARTRXINTR State

FIFO EVENT	RECEIVE INTERRUPT STATUS
FIFOs are enabled and the receive FIFO reaches the programmed trigger level.	Receive interrupt is cleared by either: • Reading data from the receive FIFO until it becomes less than the trigger level, or • Clearing the interrupt.
FIFOs are disabled (have a depth of one location) and data is received, filling the location.	Receive interrupt is cleared by either: • Performing a single read of the receive FIFO, or • Clearing the interrupt.

19.3.2.2 UARTTXINTR

UARTTXINTR is the transmit interrupt. The transmit interrupt is based on a transition through a level, rather than on the level itself. When the interrupt and the UART are enabled before any data writes to the transmit FIFO, the interrupt is not set. The interrupt is only set after written data exits the single location of the transmit FIFO, leaving the FIFO empty.

Table 19-34. UARTTXINTR State

FIFO EVENT	TRANSMIT INTERRUPT STATUS
FIFOs are enabled and the transmit FIFO reaches the programmed trigger level.	The transmit interrupt is cleared by either: • Writing data to the transmit FIFO until it becomes greater than the trigger level, or • Clearing the interrupt.
FIFOs are disabled (have a depth of one location) and no data is present in the transmitter's single location.	The transmit interrupt is cleared by either: • Performing a single write to the transmit FIFO, or • Clearing the interrupt.

19.3.2.3 UARTINTR

The UARTINTR interrupt is the combined interrupt for UART0 and UART1. It is asserted if one or more of the other interrupts are asserted.

Chapter 20
UART2

The UART2 peripheral offers similar functionality to the industry standard 82510. It performs serial-to-parallel conversion on data received from a peripheral device and parallel-to-serial conversion on data transmitted to the peripheral device. The CPU reads and writes data and control/status information through the AMBA APB interface. The transmit and receive paths are buffered with internal FIFO memories that support a programmable depth from 1 to 4.

Figure 20-1 shows a block diagram of the UART.

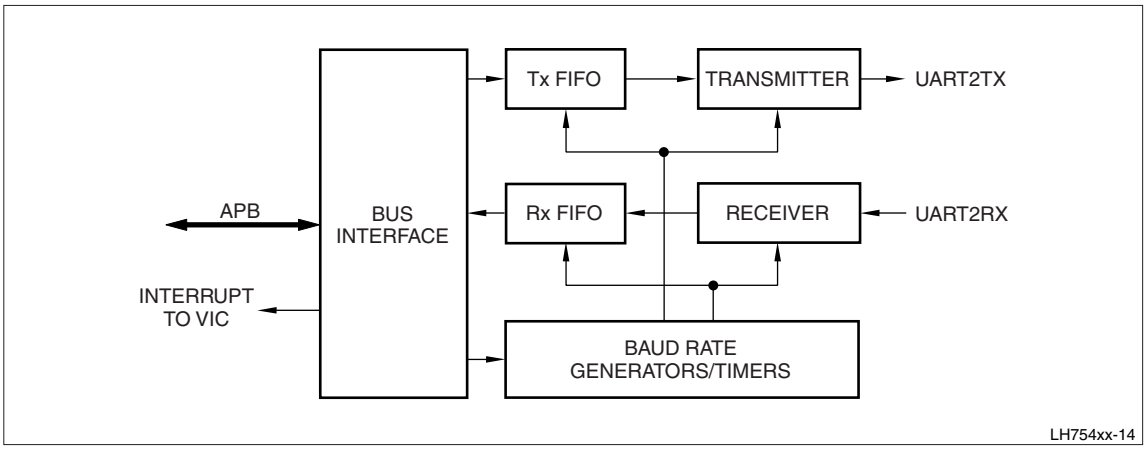


Figure 20-1. UART2 Block Diagram

20.1 UART2 Features

UART2 provides the following features:

- Supports baud rates up to 3,225,600 baud (given a system clock of 70 MHz)
- Support for 5, 6, 7, 8, or 9 data bits per character
- Even, odd, HIGH, LOW, software, or no parity-bit generation and detection
- 3/4, 1, 1-1/4, 1-1/2, 1-3/4, or 2 stop-bit generation
- μLAN address flag
- Full-duplex operation
- Separate transmit and receive FIFOs, with programmable depth (1 or 4). Each FIFO has the following programmable-service 'trigger levels' and overrun protection:
  - Receive trigger levels: 1/4, 1/2, 3/4, or full
  - Transmit trigger levels: empty, 1/4, 1/2, 3/4.
- Two 16-bit baud-rate generators. Each baud-rate generator can be configured as a timer and is completely independent of the other. The mode, output, and source of each baud-rate generator are configurable.
- A single interrupt that can be triggered by transmit and receive FIFO thresholds, receive errors, control character or address marker reception, or timer timeout
- Generation and detection of breaks during UART transactions
- Support for local loopback, remote loopback, and auto-echo modes
- μLAN Address Mode.

20.2 UART2 Theory of Operation

All Control and Status Registers for the UART can be accessed through the APB.

During transmission, data is written into the transmit FIFO through the APB interface. If the UART is enabled, it causes a data frame to start transmitting with the parameters indicated in the LCR Register. Data continues to be transmitted until there is no data left in the transmit FIFO. The TxSt bit in the LSR Register is set to '0' as soon as data is written to the transmit FIFO. The TxSt bit remains LOW until the FIFO is empty and the last character, including the stop bits, has been sent.

NOTE: Once data transmission begins, wait at least one baud cycle before turning off the transmission.

20.2.1 UART Receiver Data Frame

A UART receiver data frame has the following structure:

- A start bit that indicates the beginning of the frame. The start bit consists of a '0' on the receiver input for one bit period.
- Data, which consists of at least five and no more than nine data bits
- An optional parity bit, which can be used with available hardware for parity error checking
- One or more stop bits, which indicate the end of the frame. Stop bits consist of a '1' on the receiver input for as many bit periods as the number of stop bits previously specified when programming the receiver.

The UART receiver is in the marking state (i.e., the input is 1) from the time when a stop bit is sent until the time the next start bit is received. When the receiver receives an entire frame, the UART transfers the received data and the frame status to the receiver FIFO. This buffer can have a depth of either four words (FIFO Mode) or one word (Character Mode).

The UART receiver:

- Synchronizes the incoming data
- Passes it through a digital filter to filter out the spikes
- Samples the UART2RX pin either three or seven times at a frequency of 16X the bit rate to generate the data bit. The number of sampling times is programmable.

Bit-polarity decisions are made by majority voting. If all samples do not agree, the bit is reported as a noisy bit in the RXF Register. To compensate for such environments, the 3/16 Sampling Mode can be used for high-frequency transmissions or when serious resistance-capacitance (RC) delays are expected on the channel. Similarly, the 7/16 Sampling Mode can be used for noisy media.

The UART receiver also uses a Digital Phase Lock Loop (DPLL) to overcome frequency-shift problems. However, using the DPLL in a very noisy environment can increase errors. For this reason, the DPLL can be disabled with the RMD Register.

The start bit works with the system clock to synchronize the receiver with the source driving the receiver. The start-bit verification can be performed through a majority-voting system or an absolute voting system.

- With absolute voting, all samples must agree. Otherwise, a false start bit is determined and the receiver returns to the Start Bit Search Mode.
- With majority voting, start-bit verification works according to the Sampling Mode in effect:
  - For 3/16 Sampling Mode, the UART recognizes a start bit if the input is 0 for at least two of the three samples.
  - For 7/16 Sampling Mode, the UART recognizes a start bit if the input is 0 for at least four of the seven samples.

After recognizing the start bit, the receiver repeats one of the following sequences until all data bits, any parity bit, and all stop bits are detected:

- 3/16 Sampling Mode
  1. Wait 14/16ths of a bit period, then sample the input.
  2. Wait 1/16th of a bit period, then sample the input.
  3. Wait 1/16th of a bit period, then sample the input.
  4. Choose the majority value of the three samples as its input value for that bit period.

- 7/16 Sampling Mode
  1. Wait 10/16ths of a bit period, then sample the input.
  2. Wait 1/16th of a bit period, then sample the input.
  3. Wait 1/16th of a bit period, then sample the input.
  4. Wait 1/16th of a bit period, then sample the input.
  5. Wait 1/16th of a bit period, then sample the input.
  6. Wait 1/16th of a bit period, then sample the input.
  7. Wait 1/16th of a bit period, then sample the input.
  8. Choose the majority value of the seven samples as its input value for that bit period.

After recognizing the final stop bit, the UART stores the received data frames and associated status bits in the receiver FIFO.

20.2.2 Status Conditions

UART2 adheres to the following status conditions:

- If the UART fails to detect a 1 for all programmed stop-bit periods following a data frame, the UART sets the framing-error status for that frame.
- A line break is 0 for all bits (start bit, data bits, parity bit, and stop bits). The UART sets the line-break status for each frame containing a line break.
- An address/control character marker bit indicates that the character is either a control character when in Normal Mode or an address character when in µLAN Mode.
- Enabling parity-error detection causes the UART to compare the parity bit in each frame with the parity required for the hardware. The UART sets the parity-error status for each frame containing a parity error.
- If a received character has non-identical samples for at least one of its bits, the Received Character Noisy bit is set.
- If a received character has no parity or framing error, the Received Character OK bit is set.

20.2.3 Disabling the Loading of Incoming Characters

UART2 provides the option of disabling the loading of incoming characters into the receiver FIFO by using the UNLOCK/LOCK FIFO commands. When the receiver FIFO is locked, received characters do not load into the FIFO and can be lost if another character is received. The UNLOCK/LOCK FIFO commands are useful when the CPU is not willing to receive characters or is waiting for specific control/address characters.

µLAN Mode provides three address-recognition options. Each option varies in the amount of CPU offload and degree of FIFO control.

- Automatic Mode — the receiver unlocks the FIFO whenever an address match occurs.
- Semi-Automatic Mode — the receiver unlocks the FIFO when an address character is received, whether it matches or not.
- Manual Mode — the receiver does not control the FIFO unlocking.

The receiver can be configured to be in Control Character Recognition Mode instead of µLAN Mode. There is no FIFO control in this mode, but the receiver can generate an interrupt when it receives standard ASCII or EBCDIC control characters.

The receiver can also generate an interrupt upon a character match, with either of two user-defined characters.

20.2.4 Baud Rate Generators

UART2 has two 16-bit baud-rate generators. These baud-rate generators are completely independent of each other and can be separately configured as timers.

Dividing the system clock with the divisor count generates the baud rate. Either baud-rate generator can clock either serial machine (transmitter or receiver). Alternatively, the two baud-rate generators can be cascaded to provide a larger divisor that generates a single baud rate for clocking both serial machines. The divisor for a single baud-rate generator can operate with any divisor between 1 and 65,535.

Use the following formula to calculate the UART2 baud rate for a single-baud rate generator:

$$f\text{baud} = \frac{f\text{HCLK}}{16 \times divisor}$$

If two baud-rate generators are cascaded, use either of the following formulas to calculate the single baud rate:

$$f\text{baud} = \frac{f\text{HCLK}}{16 \times divisorA \times divisorB}$$

or

$$f\text{baud} = \frac{f\text{HCLK}}{divisorA \times 16 \times divisorB}$$

In these formulas:

- divisorA is defined by the BAH and BAL Registers.
- divisorB is defined by the BBH and BBL Registers.
- The divisor range for the cascaded baud-rate generators is 1 to 4,294,836,225.

When a baud-rate generator is configured as a timer, it counts down from its divisor/count value (BAH/BAL or BBH/BBL) to 1 once it is enabled. A maskable Timer Expired interrupt is generated upon terminal count. To start (or restart) counting, the software issues a Start Timer command (see Section 20.3.2.13). The delay between the trigger and the terminal count is determined by the following equation:

Delay = (count value) × (HCLK period)

The divisor (or count, depending on the mode) can be updated during operation, unless the particular baud-rate generator is being used as a clock source by one of the serial machines and the particular serial machine is operating at the time. Do not load the count registers (BAL or BBL) with 0 at any time. Do not load the count registers with 1 in Timer Mode.

20.3 UART2 Programmer's Model

The base address for UART2 is:

UART2 Base Address: 0xFFFC2000

20.3.1 UART2 Register Summary

The Configuration, Status, and Control Registers are contained in one of four banks. Selection of banks 0 to 3 is accomplished by writing to the GIR Register bits 6 and 5. The GIR Register is accessible at the same address for all register banks. See Table 20-1 for more details.

The registers are considered static, except for changes in status due to incoming and outgoing characters, changes due to interrupt generation, and changes in status due to register access via the programming interface.

20.3.1.1 Register Bank 0

DLAB (Divisor Latch Access Bit) is a bit in the Line Configure Register (LCR). See Section 20.3.2.7.

Table 20-1. Register Bank 0 (Default On Reset)

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
TXD	0x00	0	W	—	Transmit Buffered Data Register
RXD	0x00	0	R	0x00	Receive Buffered Data Register
BAL	0x00	1	RW	0x02	BRGA Divisor Least Significant Byte Register. The DLAB bit in the LCR Register must be set to access this register.
BAH	0x04	1	RW	0x00	BRGA Divisor Most Significant Byte Register. The DLAB bit in the LCR Register must be set to access this register.
GER	0x04	0	RW	0x00	General Enable Register
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register
LCR	0x0C		RW	0x00	Line Control Register
MCTRL	0x10		RW	0x00	Loopback Control Register
LSR	0x14		RW	0x60	Line Status Register
///	0x18				Reserved
ACTRL0	0x1C		RW	0x00	Address/Control Character Register 0

20.3.1.2 Register Bank 1

Table 20-2. Register Bank 1

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
TXD	0x00		W		Transmit Buffered Data Register
RXD	0x00		R	0x00	Receive Buffered Data Register
TXF	0x04		W		Transmit Character Flag Register
RXF	0x04		R	0x40	Receive Character Flag Register
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register (same register as in bank 0)
TMCTRL	0x0C		W		Timer Control Register
TMST	0x0C		R	0x30	Timer Status Register
MCTRL	0x10		W		Loopback Control Register
FLR	0x10		R	0x00	FIFO Level Register
RCM	0x14		W		Receive Command Register
RST	0x14		R	0x00	Receive Machine Status Register
TCM	0x18		W		Transmit Command Register
///	0x18		R		Reserved
ICM	0x1C		W		Internal Command Register
GSR	0x1C		R	0x12	General Status Register

20.3.1.3 Register Bank 2

Table 20-3. Register Bank 2

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
///	0x00				Reserved
FMD	0x04		RW	0x00	FIFO Mode Register
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register (same register as in bank 0)
TMD	0x0C		RW	0x00	Transmit Machine Mode Register
IMD	0x10		RW	0x0C	Internal Mode Register
ACTRL1	0x14		RW	0x00	Address/Control Character Register 1
RIE	0x18		RW	0x1E	Interrupt Enable Register
RMD	0x1C		RW	0x00	Receive Machine Mode Register

20.3.1.4 Register Bank 3

Table 20-4. Register Bank 3

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
CLCF	0x00	0	RW	0x00	Clocks Configure Register
BACF	0x04	0	RW	0x04	BRGA Configuration Register
BBL	0x00	1	RW	0x05	BRGB Divisor LSB Register. The DLAB bit in the LCR Register needs to be set to access this register.
BBH	0x04	1	RW	0x00	BRGB Divisor MSB Register. The DLAB bit in the LCR Register needs to be set to access this register.
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register (same register as in bank 0)
BBCF	0x0C		RW	0x84	BRGB Configuration Register
///	0x10				Reserved
///	0x14				Reserved
TMIE	0x18		RW	0x00	Timer Interrupt Enable Register

20.3.2 UART2 Register Definitions

20.3.2.1 Transmit Buffered Data Register

Register Banks: 0 and 1

TXD is the Transmit Buffered Data Register. The active bits used in this register are Write Only. The TXD Register holds the next data byte to be pushed into the Transmit FIFO.

Table 20-5. TXD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W
ADDR	0xFFFC2000 + 0x00															

NOTE: The reset value of this register's bits is indeterminate.

Table 20-6. TXD Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads indeterminate. Write 0.
7:0	D7:D0	**Transmitted Data** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.2 Receive Buffered Data Register

Register Banks: 0 and 1

RXD is the Receive Buffered Data Register. The RXD Register holds the earliest received character in the Rx FIFO After System Reset, this register is undefined.

Table 20-7. RXD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x00															

Table 20-8. RXD Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Read as zero.
7:0	D7:D0	**Received Data** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.3 BRGA Divisor Least Significant Byte Register

Register Bank: 0

BAL is the BRGA Divisor Least Significant Byte Register. The BAL Register holds the least-significant byte of the BRGA divisor/count value. The Divisor Latch Access Bit (DLAB) bit in the LCR Register must be set to access this register (see Section 20.3.2.7). The possible programmed values for this register range from 2 to 65,535.

Table 20-9. BAL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x00															

Table 20-10. BAL Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	D7:D0	**Least-Significant Byte of BRGA Divisor/Count Value** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.4 BRGA Divisor Most Significant Byte Register

Register Bank: 0

BAH is the BRGA Divisor Most Significant Byte Register. The BAH Register holds the most-significant byte of the BRGA divisor/count value. The DLAB bit in the LCR Register must be set to access this register (see Section 20.3.2.7).

Table 20-11. BAH Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x04															

Table 20-12. BAH Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	D7:D0	**Most-Significant Byte of BRGA Divisor/Count Value** Most-significant byte of BRGA divisor/count value. Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.5 General Enable Register

Register Bank: 0

GER is the General Enable Register. The GER Register enables or disables the bits of the GSR Register from being reflected in the GIR Register. GER acts as the Device Enable Register, masking the interrupt requests from the UART blocks.

Table 20-13. GER Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										TIE	TxIE	///	RxIE	TFIE	RFIE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	R	RW	RW	RW
ADDR	0xFFFC2000 + 0x04															

Table 20-14. GER Register Definitions

BITS	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5	TIE	**Timers Interrupt Enable**
4	TxIE	**Transmitter Interrupt Enable**
3	///	**Reserved** Write the reset value.
2	RxIE	**Receiver Interrupt Enable**
1	TFIE	**Transmit FIFO Interrupt Enable**
0	RFIE	**Receive FIFO Interrupt Enable**

20.3.2.6 General Interrupt/Bank Register

Register Banks: 0, 1, 2, and 3

GIR is the General Interrupt/Bank Register. The GIR Register holds the highest priority enabled pending interrupt from the GSR Register. This register also holds a pointer to the current register segment. Writing to this register updates only the Bank bits.

Table 20-15. GIR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									BANK 1	BANK 0	///	BI2	BI1	BI0	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RW	R	R	R	R	R	R	R	R	R	RW	RW	R	R	R	R	R
ADDR	0xFFFC2000 + 0x08															

Table 20-16. GIR Register Definitions

BITS	NAME	DESCRIPTION
31:7	///	**Reserved** Write the reset value.
6	Bank 1	**Bank 1** Works with bit [5] to indicate the use of a specific type of bank. Possibilities are: • 8250A/16450 Compatible Bank (Bank 0) • General Work Bank (Bank 1) • General Configuration Bank (Bank 2) • Baud Rate Generation Configuration Bank (Bank 3) For more information, see Table 20-17.
5	Bank 0	**Bank 0** Works with bit [6] to indicate the use of a specific type of bank. Possibilities are: • 8250A/16450 Compatible Bank (Bank 0) • General Work Bank (Bank 1) • General Configuration Bank (Bank 2) • Baud Rate Generation Configuration Bank (Bank 3) For more information, see Table 20-17.
4	///	**Reserved** Write the reset value.
3:1	BI2, BI1, BI0	**Read Only Bits** Decodes six different pending interrupts (see Table 20-18).
0	///	**Reserved** Write the reset value.

Table 20-17. Bank Select Bits [6:5]

BANK 1	BANK 0	SELECTED BANK
0	0	8250A/16450 Compatible Bank (Bank 0)
0	1	General Work Bank (Bank 1)
1	0	General Configuration Bank (Bank 2)
1	1	Baud Rate Generation Configuration Bank (Bank 3)

Table 20-18. Pending Interrupt Status Bits [3:1]

BI2	BI1	BI0	PENDING INTERRUPT
0	0	0	Not Used
0	0	1	Transmit FIFO Interrupt (lowest priority)
0	1	0	Receive FIFO Interrupt
0	1	1	Receiver Interrupt
1	0	0	Transmitter Interrupt
1	0	1	Timer Interrupt (highest priority)

20.3.2.7 Line Control Register

Register Bank: 0

LCR is the Line Control Register. The LCR Register defines the basic configuration of the serial link.

Table 20-19. LCR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								DLAB	SBK	PM2	PM1	PM0	SBLO	CL1	CL0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x0C															

Table 20-20. LCR Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	DLAB	**BAL and BAH Register Access** 0 = Denies access to the access to the BAL and BAH Registers in Bank 0, and the BBL and BBH Registers in Bank 3. 1 = Allows access to the access to the BAL and BAH Registers in Bank 0, and the BBL and BBH Registers in Bank 3.
6	SBK	**UART2TX Pin** 1 = Forces the UART2TX pin LOW. The UART2TX pin remains LOW, regardless of all activities, until this bit is reset.
5	PM2	**Parity Mode** Works with PM1, PM0, and bit [2] from the Transmit Machine Mode Register to define the supported parity mode. See Table 20-21.
4	PM1	**Parity Mode** Works with PM2, PM0, and bit [2] from the Transmit Machine Mode Register to define the supported parity mode. See Table 20-21.
3	PM0	**Parity Mode** Works with PM2, PM1, and bit [2] from the Transmit Machine Mode Register to define the supported parity mode. See Table 20-21.
2	SBL0	**Stop Bit Length** Works with bits [1:0] from the Transmit Machine Mode Register to define the stop-bit length for transmission. The Receive Machine can identify 3/4 stop bit or more. See Table 20-22.
1	CL1	**Character Bit Length** Works with CL0 and bit [5]) from the Transmit Machine Mode Register to define a character's bit length. See Table 20-23.
0	CL0	**Character Bit Length** Works with CL1 and bit [5]) from the Transmit Machine Mode Register to define a character's bit length. See Table 20-23.

Table 20-21. Parity Modes

PM0	SPF	PM2	PM1	PARITY SELECTED
0	x	x	x	N Parity
1	0	0	0	Odd Parity
1	0	0	1	Even Parity
1	0	1	0	High Parity
1	0	1	1	Low Parity
1	1	0	0	Software Parity

Table 20-22. Stop Bit Lengths

SBL2	SBL1	SBL0	STOP BIT LENGTH
0	0	0	4/4
0	0	1	6/4 or 8/4*
0	1	0	3/4
0	1	1	4/4
1	0	0	5/4
1	0	1	6/4
1	1	0	7/4
1	1	1	8/4

NOTE: * 6/4 if character length is 6 bits; otherwise, 8/4.

Table 20-23. Character Bit Lengths

NBCL	CL1	CL0	CHARACTER LENGTH
0	0	0	5 Bits
0	0	1	6 Bits
0	1	0	7 Bits
0	1	1	8 Bits
1	0	0	9 Bits

20.3.2.8 Loopback Control Register

Register Banks: 0 and 1

MCTRL is the Loopback Control Register. The MCTRL Register places UART 2 into the Loopback Mode selected with the IMD Register (described in Section 20.3.2.23).

For Bank 0, bit [4] is Read/Write and has a reset value of 0x00. For Bank 1, bit [4] is Read Only and its reset bits are indeterminate.

Table 20-24. MCTRL Register (Bank 0)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											LC	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	R
ADDR	0xFFFC2000 + 0x10															

Table 20-25. MCTRL Register (Bank 1)

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											LC	///			
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x10															

Table 20-26. MCTRL Register Definitions

BITS	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	LC	**Loopback Control Bit** Places UART2 into Loopback Mode. 0 = Loopback is disabled. 1 = Loopback is enabled.
3:0	///	**Reserved** Write the reset value.

20.3.2.9 Line Status Register

Register Bank: 0

LSR is the Line Status Register. The LSR Register holds the status of the serial link. It is provided for compatibility with the Intel 8250A UART. This register shares the following five bits with the RST Register (described in Section 20.3.2.17):

- BkD
- FE
- PE
- OE
- RFIR.

When this register is read, the read operation clears bits [7:0] of the RST Register and bits [4:0] of this register. Similarly, these same bits in the RST and LSR registers get cleared when the RST Register is read.

Table 20-27. LSR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									TxST	TFST	BkD	FE	PE	OE	RFIR
RESET	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x14															

Table 20-28. LSR Register Definitions

BITS	NAME	DESCRIPTION
31:7	///	**Reserved** Write the reset value.
6	TxST	**Transmit Machine Status Bit** This bit is functionally identical to the TxIR bit of the GSR Register. 1 = Transmit Machine is in idle state. Note: Idle may indicate that the Transmit Machine (txM) is empty or disabled.
5	TFST	**Transmit FIFO Status** Functionally identical to the TFIR bit of the GSR Register. It indicates that the Transmit FIFO level is equal to or below the Transmit FIFO Threshold. To disable the transmit FIFO status from being reflected in GIR, either: • Write a zero to the TFIE bit of the GER Register, or • Use the Tx FIFO HOLD INTERRUPT logic. When the TxFIFO is in the hold state, no interrupts are generated, regardless of the TFIR and TFIE bit settings. The Transmit FIFO enters the hold state when the DPU reads the GIR Register and the source of the interrupt is TxFIFO. To exit, the CPU must either set the TFIR bit of GSR to zero by writing a character to TxFIFO or setting the TFIE bit of GER to zero to disable the Tx FIFO.
4	BkD	**Break Detected** Functionally equivalent to the BkD bit of the RST Register.
3	FE	**Framing Error Detected** Functionally equivalent to the FE bit of the RST Register.
2	PE	**Parity Error** Functionally equivalent to the PE bit of the RST Register.
1	OE	**Overrun Error** Functionally equivalent to the OE bit of the RST Register.
0	RFIR	**Receive FIFO Interrupt Request** Functionally identical to the RFIR bit of the GSR Register. Indicates that the RX FIFO level is above the Rx FIFO threshold. This bit is forced LOW during any READ from the Rx FIFO. A zero written to this bit acknowledges an Rx FIFO interrupt.

20.3.2.10 Address/Control Character Register0

Register Bank: 0

The ACTRL0 Register contains a byte that is compared to each received character. The exact function depends on the configuration of the IMD Register (see Section 20.3.2.23).

In Normal Mode, this register can be used to program a special control character; in this case, a matched character is reported in the RST Register (see Section 20.3.2.17).

The maximum length of the control characters is eight bits. If the length is less than eight bits, the character must be right-justified, with the leading bits filled with zeros. In μLAN Mode, this register contains the 8-bit station address for recognition. In this mode, only incoming address characters (that is, characters with the address bit set) are compared to these registers. The PCRF bit in the RST Register is not set when an Address or Control Character match occurs.

Table 20-29. ACTRL0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x1C															

Table 20-30. ACTRL0 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	D7:D0	**Data** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.11 Transmit Character Flag Register

Register Banks: 1

TXF is the Transmit Flag Register. The active bits used in this register are Write Only.

The TXF Register holds additional components of the next character to be pushed into the Tx FIFO. The contents of this register are pushed into the Tx FIFO with the transmit Data Register when the CPU writes to the TxD Register.

Table 20-31. TXF Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								μLAN	SP	D8	///				
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W
ADDR	0xFFFC2000 + 0x04															

NOTE: The reset value of this register's bits is indeterminate.

Table 20-32. TXF Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads indeterminate. Write 0.
7	μLAN	**Address Marker Bit** Specifies the address marker bit and is transmitted in μLAN Mode.
6	SP	**Parity Bit** Specifies the parity bit for the character that is transmitted in Software Parity Mode.
5	D8	**9th Bit of Data** Specifies the 9th bit of data in a 9-bit operating mode.
4:0	///	**Reserved** Reads indeterminate. Write 0.

20.3.2.12 Received Character Flags Register

Register Bank: 1

RXF is the Received Character Flags Register. The RXF Register contains additional information about the character in the RXD Register. This register's contents are loaded by the receiver simultaneously with the RXD Register.

Table 20-33. RXF Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									ROK	RXN	RPE	ACR	BKF	RFE	RND
RESET	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x04															

Table 20-34. RXF Register Definitions

BITS	NAME	DESCRIPTION
31:7	///	**Reserved** Write the reset value.
6	ROK	**Received Character OK** Indicates that the character in RxD has no parity or framing error. The parity error is not included in the Software Parity Mode.
5	RXN	**Received Character Noisy** Indicates that the received character was noisy and had no identical samples for at least one of its bits.
4	RPE	**Receive Character Parity Error** Indicates that the RxD character had a parity error. However, in Software Parity Mode, the received parity bit is held as is. For information about the different parity modes, see Table 20-21.
3	ACR	**Address/Control Character Marker** Indicates that the character in RxD is either a control character in Normal Mode or an address character in μLAN Mode.
2	BKF	**Break Flag** Indicates that the character is part of a break sequence.
1	RFE	**Receive Character Framing Error** Indicates that no stop bit was found for the character in RxD. Note that a framing error is generated for the first character of the break sequence.
0	RND	**Ninth Bit of Received Character** The most-significant bit of the character in RxD writes into this bit. This bit is zero for characters with fewer than nine bits.

20.3.2.13 Timer Control Register

Register Bank: 1

TMCTRL is the Timer Control Register. The active bits used in this register are Write Only.

The TMCTRL Register controls the operation of the following UART timers:

- STA and TGA
- STB and TGB.

A timer has no effect when it is configured as a baud-rate generator. TGA and TGB are not reset after command execution.

Table 20-35. TMCTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										TGB	TGA	///		STB	STA
RESET	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
RW	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ADDR	0xFFFC2000 + 0x0C															

NOTE: The reset value of this register's bits is indeterminate.

Table 20-36. TMCTRL Register Definitions

BITS	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5	TGB	**Timer-B Gate** Serves as a gate for Timer B operation. 0 = Disables counting. 1 = Enables counting.
4	TGA	**Timer-A Gate** Serves as a gate for Timer A operation. 0 = Disables counting. 1 = Enables counting.
3:2	///	**Reserved** Write the reset value.
1	STB	**Start Timer B** Loads/reloads Timer B with its count value (BBH/BBL). At terminal count a status bit is set in TMST (TBEx).
0	STA	**Start Timer A** Loads/reloads Timer A with its count value (BAH/BAL). At terminal count, a status bit is set in TMST (TAEx).

20.3.2.14 Timer Status Register

Register Bank: 1

TMST is the Timer Status Register. The TMST Register holds the status of the timers. Bits [1] and [0] of this register generate interrupts that are reflected in the TIR bit (bit [5]) of the GSR Register (see Section 20.3.2.20).

Bits [5] and [4] of this register display the counting status, but do not generate interrupts. This register is not useful when using the timer in Baud Rate Generation Mode.

When this register is read, the read operation clears bits [1:0] of this register.

NOTE: Do not use the TMST Register to check the timer-expire status of UART2. Instead, use the UART2 Timer Interrupts.

Table 20-37. TMST Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										GBS	GAS	///		TBEx	TAEx
RESET	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x0C															

Table 20-38. TMST Register Definitions

BITS	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5	GBS	**Gate B State** Indicates the counting state of the software gate of Timer B, as written through the TMCTRL Register (see Section 20.3.2.13). 0 = Disables counting. 1 = Enables counting. This bit does not generate an interrupt.
4	GAS	**Gate A State** Reflects the state of the software gate of Timer A, as written through the TMCTRL Register (see Section 20.3.2.13). 0 = Disables counting. 1 = Enables counting. This bit does not generate an interrupt.
3:2	///	**Reserved** Write the reset value.
1	TBEx	**Timer B Expired** 1 = Generates an interrupt through the TIR bit of the GSR Register to indicate that Timer B count has expired. This bit is set via the terminal count pulse that the timer generates when it terminates counting.
0	TAEx	**Timer A Expired** 1 = Generates an interrupt through the TIR bit of the GSR Register to indicate that Timer A count has expired. This bit is set via the terminal count pulse that the timer generates when it terminates counting.

20.3.2.15 FIFO Level Register

Register Bank: 1

FLR is the FIFO Level Register. The FLR Register holds the current Receive and Transmit FIFO occupancy levels.

Table 20-39. FLR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									RFL2	RFL1	RFL0	///	TFL2	TFL1	TFL0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x10															

Table 20-40. FLR Register Definitions

BITS	NAME	DESCRIPTION
31:7	///	**Reserved** Write the reset value.
6:4	RFL2, RFL1, RFL0	**Receive FIFO Level of Occupancy** Indicates the number of characters in the Receive FIFO. The valid range is from zero (000) to four (100).
3	///	**Reserved** Write the reset value.
2:0	TFL2, TFL1, TFL0	**Transmit FIFO Level of Occupancy** Indicates the number of characters in the Transmit FIFO. The valid range is from zero (000) to four (100).

20.3.2.16 Receive Command Register

Register Bank: 1

RCM is the Receive Command Register. The RCM Register controls the operation of the receive machine. The active bits used in this register are Write Only.

Table 20-41. RCM Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								RXE	RXDI	FRM	FRF	LRF	ORF	///	
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	W	W	W	W	W	W	R	R
ADDR	0xFFFC2000 + 0x14															

NOTE: The reset value of this register's bits is indeterminate.

Table 20-42. RCM Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Reads indeterminate. Write 0.
7	RXE	**Receive Enable** 1 = Enables the reception of characters.
6	RXDI	**Receive Disable** 1 = Disables the reception of data on RXD pin. RxDI takes priority over RxE in disabling the reception of characters.
5	FRM	**Flush Receive Machine** 1 = Resets the receiver logic, except registers and FIFOs, enables reception, and unlocks the receive FIFO.
4	FRF	**Flush Receive FIFO** Setting this bit clears the Rx FIFO.
3	LFR	**Lock Rx FIFO** 1 = Disables the write mechanism of the Rx FIFO, so that characters subsequently received are lost (not written to the Rx FIFO). Reception is not disabled and complete status/event reporting continues. Use this command in the μLAN Mode to disable loading of characters into the RxFIFO, until an address match is detected. LRF takes priority over ORF in locking Rx FIFO.
2	ORF	**Open (unlock) Rx FIFO** 1 = Enables (unlocks) the Rx FIFO write mechanism.
1:0	///	**Reserved** Reads indeterminate. Write 0.

20.3.2.17 Receive Machine Status Register

Register Bank: 1

RST is the Receive Machine Status Register. The RST Register displays the status of the receive machine. It reports events that occurred since the RST was cleared.

All RST Register contents, except bit [0], are cleared when it is read. Each bit in this register, when set, can cause an interruption. Five bits of this register are shared with the LSR Register.

When this register is read, the read operation clears bits [7:0] of this register and bits [4:0] of the LSR Register. Similarly, these same bits in the RST and LSR Registers are cleared when the LSR Register is read.

Table 20-43. RST Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								CRF	PCRF	BKT	BKD	FE	PE	OE	RFIR
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x14															

Table 20-44. RST Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	CRF	**Control/Address Character Received** 1 = Causes an interrupt if a control character or address character is received. In μLAN Mode, this interrupt indicates that an address character has been received. In Normal Mode, this interrupt indicates that a standard ASCII or EBCDIC control character has been received.
6	PCRF	**Programmed Control/Address Character Received** 1 = Causes an interrupt when an address or control character match occurs. In μLAN Mode, this interrupt indicates that an address character has been received. In Normal Mode, this interrupt indicates that a standard ASCII or EBCDIC control character has been received.
5	BKT	**Break Terminated** Indicates that a break condition has been terminated.
4	BKD	**Break Detected** Indicates that a break condition has been detected.
3	FE	**Framing Error** Indicates that a received character did not have a valid stop bit.
2	PE	**Parity Error** Indicates that a received character had a parity error.
1	OE	**Overrun Error** Indicates that a received character was lost because the Rx FIFO was full.
0	RIFR	**Receive FIFO Interrupt Request** Functionally identical to the RFIR bit of the GSR Register. Indicates that the RX FIFO level is above the Rx FIFO threshold. This bit is forced LOW during any READ from the Rx FIFO. A zero written to this bit acknowledges an Rx FIFO interrupt.

20.3.2.18 Transmit Command Register

Register Bank: 1

TCM is the Transmit Command Register. The TCM Register enables or disables transmission by the transmit machine, and clears the Tx FIFO.

Table 20-45. TCM Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///												FTM	GRG	TxEN	TxDI
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W
ADDR	0xFFFC2000 + 0x18															

NOTE: The reset value of this register's bits is indeterminate.

Table 20-46. TCM Register Definitions

BITS	NAME	DESCRIPTION
31:4	///	**Reserved** Reads indeterminate. Write 0.
3	FTM	**Flush Transmit Machine** 1 = Resets the transmit machine logic, except for the registers and FIFO, and enables transmission.
2	FTF	**Flush Transmit FIFO** 1 = Clears the Tx FIFO. Data remains in the Tx FIFO until this bit is set. The FIFO starts out flushed at reset.
1	TxEN	**Transmit Enable** 1 = Enable transmission by the transmit machine.
0	TxDI	**Transmit Disable** 1 = Disables transmission. If transmission is occurring when this command is issued, the transmitter completes transmission of the current character before disabling transmission TxDI has priority over TxEN when controlling transmission.

20.3.2.19 Internal Command Register

Register Bank: 1

ICM is the Internal Command Register. The active bits used in this register are Write Only.

Table 20-47. ICM Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///												INTA	STC	///	
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFC2000 + 0x1C															

NOTE: The reset value of this register's bits is indeterminate.

Table 20-48. ICM Register Definitions

BITS	NAME	DESCRIPTION
31:4	///	**Reserved** Reads indeterminate. Write 0.
3	INTA	**Interrupt Acknowledge** This bit provides for an explicit acknowledgement of the device interrupt request. This bit is provided for Manual Acknowledge Mode (see Section 20.3.3.1). This bit forces the INT pin inactive for two clock cycles. After two clock cycles have elapsed, the INT pin can go active again if other enabled interrupts are pending.
2	STC	**Status Clear** 1 = Clears the RST, MSR, and TMST Registers.
1:0	///	**Reserved** Reads indeterminate. Write 0.

20.3.2.20 General Status Register

Register Bank: 1

GSR is the General Status Register. The GSR Register reflects all pending block-level interrupt requests. Each bit in the GSR Register reflects the status of a block and can be individually enabled by the GER Register (see Section 20.3.2.5). The GER Register masks-out GIR interrupts; it does not affect the GSR Register bits. To mask-out the bits in GSR so they do not appear in GSR, mask them out at a lower level.

Table 20-49. GSR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										TIR	TXIR	///	RXIR	TFIR	RFIR
RESET	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC2000 + 0x1C															

Table 20-50. GSR Register Definitions

BITS	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5	TIR	**Timers Interrupt Request** Indicates that a timer has expired. See Section 20.3.2.14.
4	TXIR	**Transmit Machine Interrupt Request** Indicates that the transmit machine is either empty or disabled/idle.
3	///	**Reserved** Write the reset value.
2	RXIR	**Receiver Interrupt Request** Generates the receiver interrupt. Servicing of the interrupt is defined by the RST Register (see Section 20.3.2.17).
1	TFIR	**Tx FIFO Interrupt Request** Indicates that FIFO occupancy is equal to or below the threshold.
0	RFIR	**Receive FIFO Interrupt Request** Indicates that Rx FIFO occupancy is above threshold.

20.3.2.21 FIFO Mode Register

Register Bank: 2

FMD is the FIFO Mode Register. The FMD Register configures the Tx and Rx FIFOs threshold levels; the number of characters contained in the FIFOs that can cause an interrupt.

Table 20-51. FMD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										RFT1	RFT0	///		TFT1	TFT0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW
ADDR	0xFFFC2000 + 0x04															

Table 20-52. FMD Register Definitions

BITS	NAME	DESCRIPTION
31:6	///	**Reserved** Write the reset value.
5:4	RFT1, RFT0	**Receive FIFO Threshold** When the number of characters in the Rx FIFO is greater than the number indicated by these bits, the Rx FIFO interrupt is activated.
3:2	///	**Reserved** Write the reset value.
1:0	TFT1, TFT0	**Transmit FIFO Threshold** When the number of characters in the Tx FIFO is less than or equal to the number indicated by these bits, the Tx FIFO interrupt is activated.

20.3.2.22 Transmit Machine Mode Register

Register Bank: 2

TMD is the Transmit Machine Mode Register. The TMD Register, together with the LCR Register, defines the transmitter operating mode.

Table 20-53. TMD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								EED	CED	NBCL	///		SPF	SBL2	SBL1
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	R	R	RW	RW	RW
ADDR	0xFFFC2000 + 0x0C															

Table 20-54. TMD Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	EED	**Error Echo Disable** 1 = Disables echo of characters received with errors (valid in Echo Mode only). Echo Mode select is set in the IMD Register.
6	CED	**Control Character Echo Disable** 1 = Disables echo of characters recognized as control characters or address characters in μLAN Mode (valid in Echo Mode only). The control character or address character is set in the ACR0 Register. Echo Mode select is set in the IMD Register.
5	NBCL	**Nine-bit Length** Works with bits [1:0] of the CLR Register to select transmit/receive character length of nine bits. See Table 20-23.
4:3	///	**Reserved** Write the reset value.
2	SPF	**Software Parity Force** 1 = Defines the parity modes along with bits [5:3] of the LCR Register (see Section 20.3.2.7). When software parity is enabled (see Table 20-21), the software must determine the parity bit through the TxF Register on transmission or check the parity bit in RxF upon reception. See Table 20-23.
1:0	SBL2, SBL1	**Stop Bit Length** Works with bit [2] of the LCR Register to define the stop bit length. For parity modes supported by the LCR Register, see Table 20-21.

20.3.2.23 Internal Mode Register

Register Bank: 2

IMD is the Internal Mode Register. The IMD Register defines the General Device Operating Mode.

Table 20-55. IMD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///												///	FRD	μLM	LEM
RESET	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x10															

Table 20-56. IMD Register Definitions

BITS	NAME	DESCRIPTION
31:3	///	**Reserved** Write the reset value.
2	RFD	**Receive FIFO Depth** Configures the depth of the Rx FIFO. 0 = Four bytes 1 = One byte. The FIFO acts as a 1-byte buffer to emulate the 8250A UART.
1	μLM	**μLAN Mode** Enables the UART to recognize and/or match an address using the 9-bit MCS-51 asynchronous protocol. 0 = Normal Mode 1 = μLAN Mode See the ACTRL0 Register for a complete description of the μLAN Mode.
0	LEM	**Loopback/Echo Mode Select** Selects either loopback or echo operation, depending on the operating mode selected by bit [4] of the MCTRL Register. In Loopback Mode (LC = 1), this bit selects between local and remote loopback: 0 = Local loopback 1 = Remote loopback In Echo Mode (LC = 0), this bit selects between echo or non-echo operation: 0 = No echo 1 = Echo operation

20.3.2.24 Address/Control Character Register 1

Register Bank: 2

Like the ACTRL0 Register, the ACTRL1 Register contains a byte that is compared to each received character. The value in ACTRL1 is usually a value different from the one in ACTRL0. For information about how the Address/Control Character Registers are used, see Section 20.3.2.10.

Table 20-57. ACTRL1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D3	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x14															

Table 20-58. ACTRL1 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	D7:D0	**Data** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.25 Receive Interrupt Enable Register

Register Bank: 2

RIE is the Receive Interrupt Enable Register. The RIE Register enables interrupts from the Rx state machine. It is used to mask out interrupt requests generated by the status bits of the RST Register (described in Section 20.3.2.17).

Table 20-59. RIE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								CRE	PCRE	BKTE	BKDE	FEE	PEE	OEE	///
RESET	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	R
ADDR	0xFFFC2000 + 0x18															

Table 20-60. RIE Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	CRE	**Control/μLAN Address Character Recognition Interrupt Enable** 1 = Enables an interrupt when the CRF bit of the RST Register is set.
6	PCRE	**Programmable Control/Address Character Match Interrupt Enable** 1 = Enables an interrupt on the PCRF bit of the RST Register.
5	BKTE	**Break Termination Interrupt Enable** 1 = Enables an interrupt on the BKT bit of the RST Register.
4	BKDE	**Break Detection Interrupt Enable** 1 = Enables an interrupt on the BKD bit of the RST Register.
3	FEE	**Framing Error Enable** 1 = Enables an interrupt on the FE bit of the RST Register.
2	PEE	**Parity Error Enable** 1= Enables an interrupt on the PE bit of the RST Register.
1	OEE	**Overrun Error Enable** 1 = Enables an interrupt on the OE bit of the RST Register.
0	///	**Reserved** Write the reset value.

20.3.2.26 Receive Machine Mode Register

Register Bank: 2

RMD is the Receive Machine Mode Register. The RMD Register defines the receiver operating mode. For information about manually locking the FIFO, see Section 20.3.2.16.

Table 20-61. RMD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								µCM0	µCM1	DPD	SWM	SSM	///		
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	R	R	R
ADDR	0xFFFC2000 + 0x1C															

Table 20-62. RMD Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:6	µCM0, µCM1	**µLAN/Control Character Recognition Mode** In Normal Mode, these bits define the Control Character Recognition Mode. In µLAN Mode, these bits define modes of address recognition: Settings for Normal Mode: 00 = No standard set Control Characters recognized 01 = ASCII Control Characters (00H through 1 FH + 7FH) 10 = Reserved 11 = EBCDIC Control Character recognized (00H – 3 FH) Settings for µLAN Mode: 00 = Manual Mode. The receiver reports reception of any address character via CRF bit of RST Register and writes it to the Rx FIFO. 01 = Semi-automatic Mode. Same as Manual Mode, but the receiver opens (unlocks) the Rx FIFO upon reception of any address characters. Subsequent received characters are written into the FIFO. You must lock the FIFO if the address character does not match the station's address. 10 = Automatic Mode. The receiver opens (unlocks) the Rx FIFO upon address match. Also, the receiver locks the Rx FIFO upon recognition of address mismatch; that is, it controls the flow of characters into the Rx FIFO, depending on the results of the address comparison. If a match occurs, it allows characters to be sent to the FIFO; if a mismatch occurs, it keeps the characters out of the FIFO by locking the FIFO. 11 = Reserved
5	DPD	**Disable Digital Phase Locked Loop** 1 = Disable the DPLL machine. (Using the DPLL in a very noisy media can increase the error rate.)
4	SWM	**Sampling Window Mode** Controls the mode of data sampling. 0 = Small window, 3/16 sampling. 1 = Large window, 7/16 sampling. Do not set bits [4] and [3] to '1' at the same time; otherwise, data may become corrupted.
3	SSM	**Start Bit Sampling Mode** Controls the mode of start-bit sampling. 0 = Majority voting for start bit: a majority of the samples determines the bit. 1 = If one of the bit samples is not 0, the start bit is not detected. Do not set bits [3] and [4] to '1' at the same time; otherwise, data may become corrupted.
2:0	///	**Reserved** Write the reset value.

20.3.2.27 Clocks Configure Register

Register Bank: 3

CLCF is the Clocks Configure Register. The CLCF Register defines the transmit and receive code modes and sources.

Table 20-63. CLCF Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									RXCS	///	TXCS	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	R	RW	R	R	R	R
ADDR	0xFFFC2000 + 0x00															

Table 20-64. CLCF Register Definitions

BITS	NAME	DESCRIPTION
31:7*	///	**Reserved** Write the reset value.
6	TXCS	**Transmitter Clock Source** Selects the source of the internal transmit clock. 0 = BRGB output 1 = BRGA output
5*	///	**Reserved** Write the reset value.
4	RXCS	**Receiver Clock Source** Selects the source of the internal receive clock. 0 = BRGB output 1 = BRGA output
3:0	///	**Reserved** Write the reset value.

NOTE: *Bits [7] and [5] must always be set to 0 for proper operation.

20.3.2.28 BRGA Configuration Register

Register Bank: 3

BACF is the BRGA Configuration Register. The BACF Register defines the BRGA clock sources and operating mode.

Table 20-65. BACF Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///													BAM	///	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R	R
ADDR	0xFFFC2000 + 0x04															

Table 20-66. BACF Register Definitions

BITS	NAME	DESCRIPTION
31:3	///	**Reserved** Write the reset value.
2	BAM	**BRGA Mode of Operation** Selects between the Timer Mode or the Baud Range Generator Mode. 0 = Timer Mode; input clock source is always the system clock. 1 = Baud Rate Generator Mode.
1:0	///	**Reserved** Write the reset value.

20.3.2.29 BRGB Divisor Least Significant Byte Register

Register Bank: 3

BBL is the BRGB Divisor Least Significant Byte Register. The BBL Register contains the least-significant byte of the BRGB divisor/count value. Acceptable values for this register range from 2 to 65,535. The DLAB bit in the LCR Register must be set to access this register (see Chapter 19, Section 19.3.1.9).

Table 20-67. BBL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x00															

Table 20-68. BBL Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	D7:D0	**Least-Significant Byte of the BRGB Divisor/Count Value** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.30 BRGB Divisor Most Significant Byte Register

Register Bank: 3

BBH is the BRGB Divisor Most Significant Byte Register. The BBH Register contains the most-significant byte of the BRGB divisor/count value. The DLAB bit in the LCR Register must be set to access this register see Chapter 19, Section 19.3.1.9).

Table 20-69. BBH Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								D7	D6	D5	D4	D3	D2	D1	D0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC2000 + 0x04															

Table 20-70. BBH Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	D7:D0	**Most-Significant Byte of the BRGB Divisor/Count Value** Bit [7] holds the most-significant bit. Bit [0] holds the least-significant bit.

20.3.2.31 BRGB Configuration Register

Register Bank: 3

BBCF is the BRGB Configuration Register. The BBCF Register defines the BRGB clock sources and operating mode.

NOTE: BRGB can also obtain its input clock from the BRGA output.

Table 20-71. BBCF Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								BBCS1	BBCS0	///			BBM	///	
RESET	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	R	R	R	RW	R	R
ADDR	0xFFFC2000 + 0x0C															

Table 20-72. BBCF Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:6	BBCS1, BBCS0	Defines the input clock sources for BRGV. 00 = System clock 01 = Reserved 10 = BRGA output 11 = Reserved These bits have no effect in Timer Mode.
5:3	///	**Reserved** Write the reset value.
2	BBM	**BRBG Mode of Operation** 0 = Timer Mode; input clock source is always the system clock. 1 = BRG Mode.
1:0	///	**Reserved** Write the reset value.

20.3.2.32 Timer Interrupt Enable Register

Register Bank: 3

TMIE is the Timer Interrupt Enable Register for the timer block. The TMIE Register masks-out interrupt requests generated by the status bits of the TMST Register.

Table 20-73. TMIE Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///														TBIE	TAIE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW
ADDR	0xFFFC2000 + 0x18															

Table 20-74. TMIE Register Definitions

BITS	NAME	DESCRIPTION
31:3	///	**Reserved** Write the reset value.
2	TBIE	**Timer A Expired Interrupt Enable** Enables Interrupt on TAEx bit of TMST.
1	TAIE	**Timer B Expired Interrupt Enable** Enables Interrupt on TBEx bit of TMST.

20.3.3 UART2 Interrupts

There are two levels of interrupt/status reporting within UART2: block-level interrupts and second-level interrupts.

- Block-level interrupts include Rx FIFO, Tx FIFO, receiver, transmitter, and timer interrupts. The status of these blocks is reported in the General Status and General Interrupt Registers.
- Second-level interrupts are those generated by the receiver and timers blocks.

Interrupt requests are maskable at both the block level and at the individual source level within the module.

Each block has a fixed priority. If more than one unmasked block requests interrupt service, the on-chip interrupt controller resolves the contention on a priority basis.

An interrupt request from a particular block is activated if one of the unmasked status bits within the Status Register for the block is set. A CPU service operation, such as reading the appropriate Status Register, resets the status bits.

20.3.3.1 Acknowledge Modes

The interrupt logic asserts an interrupt signal to the VIC when an interrupt is coded into the General Interrupt Register.

UART2 has two Interrupt Acknowledgment Modes: Automatic Acknowledge and Manual Acknowledge.

- An interrupt-service operation is considered an automatic acknowledgment in situations where the ARM7TDMI-S core must issue an explicit interrupt acknowledge command back to the UART2 peripheral. This causes the interrupt signal from UART2 to the VIC to go inactive for two clock cycles. After two clock cycles elapse, UART2 determines whether it should reassert the interrupt if there are other UART2 interrupts pending.
- In Manual Acknowledge Mode, the ARM7TDMI-S core must issue an explicit Interrupt Acknowledge command via the Interrupt Acknowledge bit of the Internal Command Register. As a result, the INT signal to the VIC is forced LOW for two clocks and then updated.

NOTE: Do not use Automatic Acknowledge Mode. Use Manual Acknowledge Mode instead.

20.3.3.2 Interrupt Service

A service operation is an operation that the CPU performs that resets the particular status bit causing the UART2 interrupt. An interrupt request within UART 2 does not reset the status bit until the interrupt source is serviced.

UART2 interrupt sources can be serviced in various ways. These include:

- Disabling the particular status bit. This method causes the interrupt to occur via the corresponding Block Enable Register.
- Setting the appropriate bit of the Enable Register to zero. This method masks off the corresponding bit in the Status Register, causing an edge on the input line to the interrupt logic.
- Masking off the particular block interrupt request in General Status Register via the General Enable Register. This method achieves the same effect as the Enable Register method above.

Issuing the Status Clear command from the Internal Command Register. This method is applicable to all sources. Table 20-75 lists the detailed service requirements for each source.

Table 20-75. Interrupt Service Requirements

INTERRUPT SOURCE	STATUS REGISTERS (BIT NUMBER)	MARK REGISTER (BIT NUMBER)	SPECIFIC SERVICE
Timers (highest priority)	TMST (1-0) GSR (5)	TMIE (1-0) GER (5)	Read TMST
Transmitters	GSR (4) LSR (4)	GER (4)	Write Character to Tx FIFO
Receiver	LSR (4-1) RST (7-1) GSR (2)	RIE (7-1) GER (2)	Read RST or LSR Write 0 to bit in RST/LSR
Rx FIFO	RST/LSR (0) GSR (0)	GER (0)	Write 0 to LSR/RST Bit zero Read Character
Tx FIFO (lowest priority)	LSR (5) GSR (1)	GER (1)	Write to FIFO Read GIR (1)

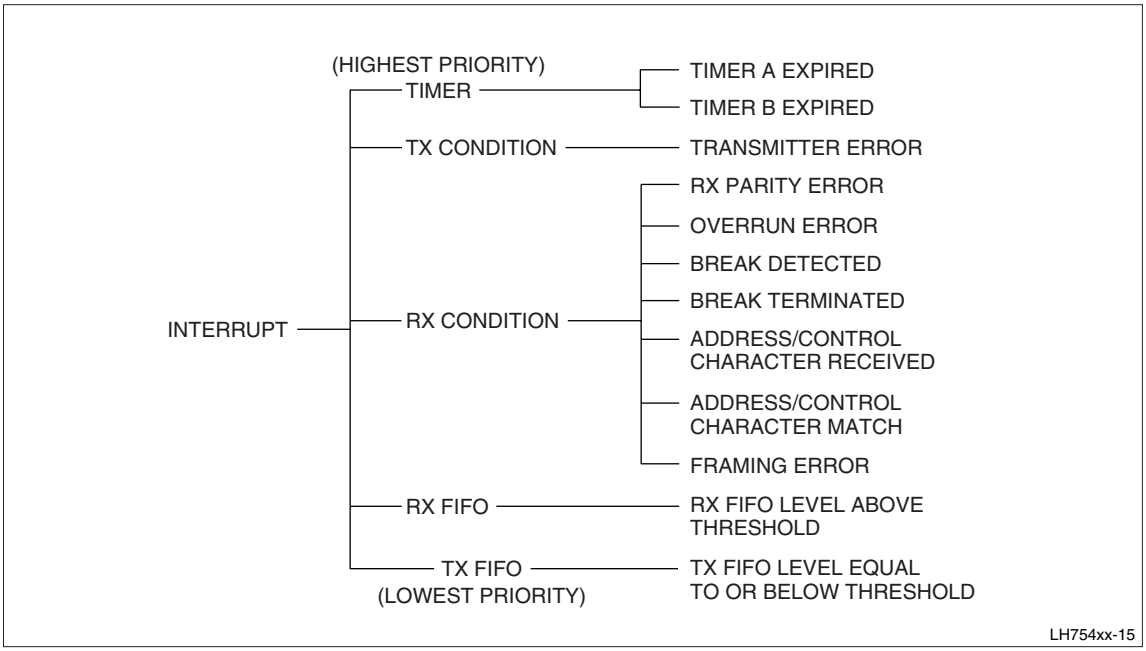


Figure 20-2. Interrupt and Status Reporting Structure

Chapter 21
General Purpose Input/Output

The General Purpose Input/Output (GPIO) is a slave module that connects to the APB. The SoCs use five GPIO modules. Each module has two 8-bit ports, designated A through J, and provides 76 bits of programmable input/output. Figure 21-1 shows a block diagram of the GPIO.

NOTE: Users may want to consult the truth tables in Chapter 24 while reading about the I/O system.

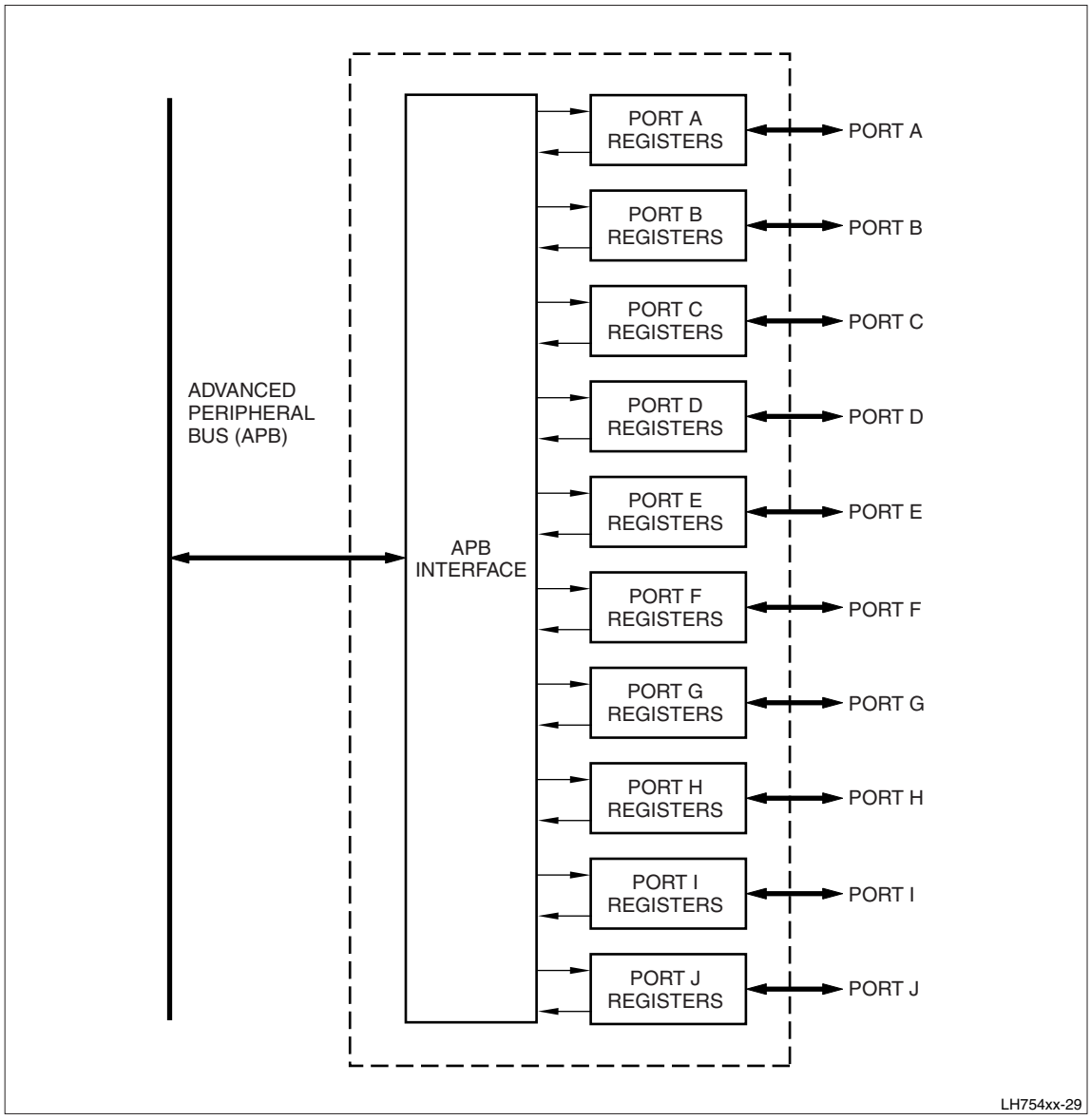


Figure 21-1. GPIO Block Diagram

21.1 GPIO Features

The GPIO supports the following features.

- 10 GPIO ports
- Programmable port direction
- All ports but one default to inputs at System Reset
- Control word read-back capability.

All GPIO pins are multiplexed with other functions. Some GPIO pins have Schmitt Trigger inputs. Some pins can have pull-ups and pull-downs that can be switched on or off under software control.

21.2 GPIO Theory of Operation

There are 10 GPIO ports:

- Seven 8-bit ports
- Two 7-bit ports
- One 6-bit port.

GPIO ports are designated A through J. Pins of all ports, except Port J, can be configured as either inputs or outputs. Port J is input only. Upon System Reset, all ports default to inputs.

Table 21-1. GPIO Ports

PORT	PROGRAMMABLE PINS
A	8 Input/Output Pins
B	6 Input/Output Pins
C	8 Input/Output Pins
D	7 Input/Output Pins
E	8 Input/Output Pins
F	7 Input/Output Pins
G	8 Input/Output Pins
H	8 Input/Output Pins
I	8 Input/Output Pins
J	8 Input Pins

21.2.1 GPIO Programmer's Model

The base addresses for the GPIO modules are:

Ports A and B: 0xFFFDF000
Ports C and D: 0xFFFDE000
Ports E and F: 0xFFFDD000
Ports G and H: 0xFFFDC000
Ports I and J: 0xFFFDB000

The location at offset 0x0C is reserved.

21.2.2 GPIO Registers Summary

Table 21-2. GPIO Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
PIDR	0x00	RW	0x00000000	Port I Data Register
PJDR	0x04	RW	0x00000000	Port J Data Register
PIDDR	0x08	RW	0x00000000	Port I Data Direction Register
	0x0C	Reserved		Reserved
PGDR	0x00	RW	0x00000000	Port G Data Register
PHDR	0x04	RW	0x00000000	Port H Data Register
PGDDR	0x08	RW	0x00000000	Port G Data Direction Register
PHDDR	0x0C	RW	0x00000000	Port H Data Direction Register
PEDR	0x00	RW	0x00000000	Port E Data Register
PFDR	0x04	RW	0x00000000	Port F Data Register
PEDDR	0x08	RW	0x00000000	Port E Data Direction Register
PFDDR	0x0C	RW	0x00000000	Port F Data Direction Register
PCDR	0x00	RW	0x00000000	Port C Data Register
PDDR	0x04	RW	0x00000000	Port D Data Register
PCDDR	0x08	RW	0x00000000	Port C Data Direction Register
PDDDR	0x0C	RW	0x00000000	Port D Data Direction Register
PADR	0x00	RW	0x00000000	Port A Data Register
PBDR	0x04	RW	0x00000000	Port B Data Register
PADDR	0x08	RW	0x00000000	Port A Data Direction Register
PBDDR	0x0C	RW	0x00000000	Port B Data Direction Register

21.2.3 GPIO Register Definitions

21.2.3.1 Port A Data Register

PADR is the Port A Data Register. The active bits used in this register are Read/Write.

Values written to PADR are output on the PA pins if the corresponding PADDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port A Data Direction Register (see Section 21.2.3.3). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-3. PADR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port A Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDF000 + 0x00															

Table 21-4. PADR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port A Data	**Port A Input/Output Data** Specifies Port A input or output data, depending on how the value of the corresponding bit in the PADDR Register is set (see Section 21.2.3.3). PADDR set as output = PADR sets the value on the GPIO Port A pins. PADDR set as input = PADR reads the value on the GPIO Port A pins.

21.2.3.2 Port B Data Register

PBDR is the Port B Data Register. The active bits used in this register are Read/Write.

Values written to PBDR are output on the PB pins if the corresponding PBDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port B Data Direction Register (see Section 21.2.3.4). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-5. PBDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										Port B Data					
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDF000 + 0x04															

Table 21-6. PBDR Register Definitions

BITS	NAME	FUNCTION
31:6	///	**Reserved** Write the reset value.
5:0	Port B Data	**Port B Input/Output Data** Specifies Port B input or output data, depending on how the value of the corresponding bit in the PBDDR Register is set (see Section 21.2.3.4). PBDDR set as output = PBDR sets the value on the GPIO Port B pins. PBDDR set as input = PBDR reads the value on the GPIO Port B pins.

21.2.3.3 Port A Data Direction Register

PADDR is the Port A Data Direction Register. The active bits used in this register are Read/Write.

Bits set in PADDR Register set the corresponding PA pin to be an output.

- Bit [7] controls pin 1 when the pin is configured as PA7. It does not control pin 1 when the pin is configured as D15.
- Bit [6] controls pin 2 when the pin is configured as PA6. It does not control pin 2 when the pin is configured as D14.
- Bit [5] controls pin 4 when the pin is configured as PA5. It does not control pin 4 when the pin is configured as D13.
- Bit [4] controls pin 5 when the pin is configured as PA4. It does not control pin 5 when the pin is configured as D12.
- Bit [3] controls pin 6 when the pin is configured as PA3. It does not control pin 6 when the pin is configured as D11.
- Bit [2] controls pin 7 when the pin is configured as PA2. It does not control pin 7 when the pin is configured as D10.
- Bit [1] controls pin 9 when the pin is configured as PA1. it does not control pin 9 when the pin is configured as D9.
- Bit [0] controls pin 10 when the pin is configured as PA0. It does not control pin 10 when the pin is configured as D8.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

Table 21-7. PADDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port A Data Direction							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDF000 + 0x08															

Table 21-8. PADDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port A Data Direction	**Port A Output/Input** Bits set = Port A output. Bits cleared = Port A input.

21.2.3.4 Port B Data Direction Register

PBDDR is the Port B Data Direction Register. The active bits used in this register are Read/Write.

Bits set in the PBDDR Register set the corresponding PB pin to be an output:

- Bit [5] controls pin 24 when the pin is configured as PB5. It does not control pin 24 when the pin is configured as nWAIT.
- Bit [4] controls pin 25 when the pin is configured as PB4. It does not control pin 25 when the pin is configured as nBLE1.
- Bit [3] controls pin 27 when the pin is configured as PB3. It does not control pin 27 when the pin is configured as nBLE0.
- Bit [2] controls pin 28 when the pin is configured as PB2. It does not control pin 28 when the pin is configured as nCS3.
- Bit [1] controls pin 29 when the pin is configured as PB1. It does not control pin 29 when the pin is configured as nCS2.
- Bit [0] controls pin 30 when the pin is configured as PB0. It does not control pin 30 when the pin is configured as nCS1.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

Table 21-9. PBDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										Port B Data Direction					
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDF000 + 0x0C															

Table 21-10. PBDDR Register Definitions

BITS	NAME	FUNCTION
31:6	///	**Reserved** Write the reset value.
5:0	Port B Data Direction	**Port B Output/Input** Bits set = Port B output. Bits cleared = Port B input.

21.2.3.5 Port C Data Register

PCDR is the Port C Data Register. The active bits used in this register are Read/Write.

Values written to PCDR are output on the PC pins if the corresponding PCDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port C Data Direction Register (see Section 21.2.3.7). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-11. PCDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port C Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDE000 + 0x00															

Table 21-12. PCDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port C Data	**Port C Input/Output Data** Specifies Port C input or output data, depending on how the value of the corresponding bit in the PCDDDR Register is set (see Section 21.2.3.7). PCDDR set as output = PCDR sets the value on the GPIO Port C pins. PCDDR set as input = PCDR reads the value on the GPIO Port C pins.

21.2.3.6 Port D Data Register

PDDR is the Port D Data Register. The active bits used in this register are Read/Write.

Values written to PDDR are output on the PD pins if the corresponding PDDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port D Data Direction Register (see Section 21.2.3.8). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-13. PDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									Port D Data						
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDE000 + 0x04															

Table 21-14. PDDR Register Definitions

BITS	NAME	FUNCTION
31:7	///	**Reserved** Write the reset value.
6:0	Port D Data	**Port D Input/Output Data** Specifies Port D input or output data, depending on how the value of the corresponding bit in the PDDDR Register is set (see Section 21.2.3.8). PDDDR set as output = PDDR sets the value on the GPIO Port D pins. PDDDR set as input = PDDR reads the value on the GPIO Port D pins.

21.2.3.7 Port C Data Direction Register

PCDDR is the Port C Data Direction Register. The active bits used in this register are Read/Write.

Bits set in the PCDDR Register set the corresponding PC pin to be an output:

- Bit [7] controls pin 32 when the pin is configured as PC7. It does not control pin 32 when the pin is configured as A23.
- Bit [6] controls pin 33 when the pin is configured as PC6. It does not control pin 33 when the pin is configured as A22.
- Bit [5] controls pin 35 when the pin is configured as PC5. It does not control pin 35 when the pin is configured as A21.
- Bit [4] controls pin 36 when the pin is configured as PC4. It does not control pin 36 when the pin is configured as A20.
- Bit [3] controls pin 37 when the pin is configured as PC3. It does not control pin 37 when the pin is configured as A19.
- Bit [2] controls pin 38 when the pin is configured as PC2. It does not control pin 38 when the pin is configured as A18.
- Bit [1] controls pin 39 when the pin is configured as PC1. It does not control pin 39 when the pin is configured as A17.
- Bit [0] controls pin 40 when the pin is configured as PC0. It does not control pin 40 when the pin is configured as A16.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

Table 21-15. PCDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port C Data Direction							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDE000 + 0x08															

Table 21-16. PCDDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port C Data Direction	**Port C Output/Input** Bits set = Port C output. Bits cleared = Port C input.

21.2.3.8 Port D Data Direction Register

PDDDR is the Port D Data Direction Register. The active bits used in this register are Read/Write.

Bits set in the PDDDR set the corresponding PD pin to be an output:

- Bit [6] controls pin 72 when the pin is configured as PD6. It does not control pin 72 when the pin is configured as INT6 or DREQ.
- Bit [5] controls pin 73 when the pin is configured as PD5. It does not control pin 73 when the pin is configured as INT5 or DACK.
- Bit [4] controls pin 74 when the pin is configured as PD4. It does not control pin 74 when the pin is configured as INT4 or UARTRX1.
- Bit [3] controls pin 76 when the pin is configured as PD3. It does not control pin 76 when the pin is configured as INT3 or UARTTX1.
- Bit [2] controls pin 77 when the pin is configured as PD2. It does not control pin 77 when the pin is configured as INT2.
- Bit [1] controls pin 78 when the pin is configured as PD1. It does not control pin 78 when the pin is configured as INT1.
- Bit [0] controls pin 79 when the pin is configured as PD0. It does not control pin 79 when the pin is configured as INT0.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

Table 21-17. PDDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									Port D Data Direction						
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDE000 + 0x0C															

Table 21-18. PDDDR Register Definitions

BITS	NAME	FUNCTION
31:7	///	**Reserved** Write the reset value.
6:0	Port D Data Direction	**Port D Output/Input** Bits set = Port D output. Bits cleared = Port D input.

21.2.3.9 Port E Data Register

PEDR is the Port E Data Register. The active bits used in this register are Read/Write.

Values written to PEDR are output on the PE pins if the corresponding PEDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port E Data Direction Register (see Section 21.2.3.11). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-19. PEDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port E Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDD000 + 0x00															

Table 21-20. PEDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port E Data	**Port E Input/Output Data** Specifies Port E input or output data, depending on how the value of the corresponding bit in the PEDDR Register is set (see Section 21.2.3.11). PEDDR set as output = PEDR sets the value on the GPIO Port E pins. PEDDR set as input = PEDR reads the value on the GPIO Port E pins.

21.2.3.10 Port F Data Register

PFDR is the Port F Data Register. The active bits used in this register are Read/Write.

Values written to PFDR will be output on the PF pins if the corresponding PFDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port F Data Direction Register (see Section 21.2.3.12). A read operation from this register returns either:

- The last value written to the Data Register, if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-21. PFDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									Port F Data						
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDD000 + 0x04															

Table 21-22. PFDR Register Definitions

BITS	NAME	FUNCTION
31:7	///	**Reserved** Write the reset value.
6:0	Port F Data	**Port F Input/Output Data** Specifies Port F input or output data, depending on how the value of the corresponding bit in the PFDDR Register is set (see Section 21.2.3.12). PFDDR set as output = PFDR sets the value on the GPIO Port F pins. PFDDR set as input = PFDR reads the value on the GPIO Port F pins.

21.2.3.11 Port E Data Direction Register

PEDDR is the Port E Data Direction Register. The active bits used in this register are Read/Write. Bits set in PEDDR set the corresponding PE pin to be an output:

- Bit [7] controls pin 99 when the pin is configured as PE7. It does not control pin 99 when the pin is configured as SSPFRM.
- Bit [6] controls pin 100 when the pin is configured as PE6. It does not control pin 100 when the pin is configured as SSPCLK.
- Bit [5] controls pin 101 when the pin is configured as PE5. It does not control pin 101 when the pin is configured as SSPRX.
- Bit [4] controls pin 102 when the pin is configured as PE4. It does not control pin 102 when the pin is configured as SSPTX.
- Bit [3] controls pin 103 when the pin is configured as PE3. It does not control pin 103 when the pin is configured as CANTX or UARTTX0.
- Bit [2] controls pin 104 when the pin is configured as PE2. It does not control pin 104 when the pin is configured as CANRX or UARTRX0.
- Bit [1] controls pin 105 when the pin is configured as PE1. It does not control pin 105 when the pin is configured as UARTTX2.
- Bit [0] controls pin 107 when the pin is configured as PE0. It does not control pin 106 when the pin is configured as UARTRX2.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

NOTE: The CANTX and CANRX functions apply to the LH75400 and LH75401 SoC devices only.

Table 21-23. PEDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port E Data Direction							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDD000 + 0x08															

Table 21-24. PEDDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port E Data Direction	**Port E Output/Input** Bits set = Port E output. Bits cleared = Port E input.

21.2.3.12 Port F Data Direction Register

PFDDR is the Port F Data Direction Register. The active bits used in this register are Read/Write.

Bits set in PFDDR set the corresponding PF pin to be an output:

- Bit [6] controls pin 108 when the pin is configured as PF6. it does not control pin 108 when the pin is configured as CTCAP2B or CTCMP2B.
- Bit [5] controls pin 109 when the pin is configured as PF5. It does not control pin 109 when the pin is configured as CTCAP2A or CTCMP2A.
- Bit [4] controls pin 110 when the pin is configured as PF4. It does not control pin 110 when the pin is configured as CTCAP1B or CTCMP1B.
- Bit [3] controls pin 111 when the pin is configured as PF3. It does not control pin 111 when the pin is configured as CTCAP1A or CTCMP1A.
- Bit [2] controls pin 113 when the pin is configured as PF2. It does not control pin 113 when the pin is configured as CTCAP0E.
- Bit [1] controls pin 114 when the pin is configured as PF1. It does not control pin 114 when the pin is configured as CTCAP0D.
- Bit [0] controls pin 115 when the pin is configured as PF0. it does not control pin 115 when the pin is configured as CTCAP0C.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

Table 21-25. PFDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									Port F Data Direction						
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDD000 + 0x0C															

Table 21-26. PFDDR Register Definitions

BITS	NAME	FUNCTION
31:7	///	**Reserved** Write the reset value.
6:0	Port F Data Direction	**Port F Output/Input** Bits set = Port F output. Bits cleared = Port F input.

21.2.3.13 Port G Data Register

PGDR is the Port G Data Register. The active bits used in this register are Read/Write.

Values written to PGDR are output on the PG pins if the corresponding PGDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Data Direction Register (see Section 21.2.3.15). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-27. PGDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port G Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDC000 + 0x00															

Table 21-28. PGDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port G Data	**Port G Input/Output Data** Specifies Port G input or output data, depending on how the value of the corresponding bit in the PGDDR Register is set (see Section 21.2.3.15). PGDDR set as output = PGDR sets the value on the GPIO Port G pins. PGDDR set as input = PGDR reads the value on the GPIO Port G pins.

21.2.3.14 Port H Data Register

PHDR is the Port H Data Register. The active bits used in this register are Read/Write.

Values written to PHDR are output on the PH pins if the corresponding PHDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port H Data Direction Register (see Section 21.2.3.16). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value written if the bit is configured as an input.

A System Reset clears all bits.

Table 21-29. PHDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port H Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDC000 + 0x04															

Table 21-30. PHDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port H Data	**Port H Input/Output Data** Specifies Port H input or output data, depending on how the value of the corresponding bit in the PHDDR Register is set (see Section 21.2.3.16). PHDDR set as output = PHDR sets the value on the GPIO Port H pins. PHDDR set as input = PHDR reads the value on the GPIO Port H pins.

21.2.3.15 Port G Data Direction Register

PGDDR is the Port G Data Direction Register. The active bits used in this register are Read/Write. Bits set in PGDDR set the corresponding PG pin to be an output:

- Bit [7] controls pin 116 when the pin is configured as PG7. It does not control pin 116 when the pin is configured as CTCAP0B or CTCMP0B.
- Bit [6] controls pin 117 when the pin is configured as PG6. It does not control pin 117 when the pin is configured as CTCAP0A or CTCMP0A.
- Bit [5] controls pin 118 when the pin is configured as PG5. It does not control pin 118 when the pin is configured as CTCLK.
- Bit [4] controls pin 120 when the pin is configured as PG4. It does not control pin 120 when the pin is configured for LCDVEEEN or LCDMOD.
- Bit [3] controls pin 121 when the pin is configured as PG3. It does not control pin 121 when the pin is configured as LCDVDDEN.
- Bit [2] controls pin 122 when the pin is configured as PG2. It does not control pin 122 when the pin is configured as LCDDSPLEN or LCDREV.
- Bit [1] controls pin 123 when the pin is configured as PG1. It does not control pin 123 when the pin is configured as LCDCLS.
- Bit [0] controls pin 124 when the pin is configured as PG0. It does not control pin 124 when the pin is configured as LCDPS.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

NOTE: The LCDMOD, LCDREV, LCDCLS, and LCDPS functions apply to the LH75401 and LH75411 SoC devices only.

Table 21-31. PGDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port G Data Direction							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDC000 + 0x08															

Table 21-32. PGDDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port G Data Direction	**Port G Output/Input** Bits set = Port G output. Bits cleared = Port G input.

21.2.3.16 Port H Data Direction Register

PHDDR is the Port H Data Direction Register. The active bits used in this register are Read/Write. Bits set in PHDDR set the corresponding PH pin to be an output:

- Bit [7] controls pin 125 when the pin is configured as PH7. It does not control pin 125 when the pin is configured as LCDDCLK.
- Bit [6] controls pin 128 when the pin is configured as PH6. It does not control pin 128 when the pin is configured as LCDLP or LCDHRLP.
- Bit [5] controls pin 129 when the pin is configured as PH5. It does not control pin 129 when the pin is configured as LCDFP or LCDSPS.
- Bit [4] controls pin 130 when the pin is configured as PH4. It does not control pin 130 when the pin is configured as LCDEN or LCDEN.
- Bit [3] controls pin 131 when the pin is configured as PH3. It does not control pin 131 when the pin is configured as LCDVD11.
- Bit [2] controls pin 132 when the pin is configured as PH2. It does not control pin 132 when the pin is configured as LCDVD10.
- Bit [1] controls pin 133 when the pin is configured as PH1. It does not control pin 133 when the pin is configured as LCDVD9.
- Bit [0] controls pin 135 when the pin is configured as PH0. It does not control pin 135 when the pin is configured as LCDVD8.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

NOTES:
1. The LCDHRLP function applies to the LH75401 and LH75411 SoC devices only.
2. The LH75400 and LH75410 SoC devices support a single LCDEN function on pin 130.

Table 21-33. PHDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port H Data Direction							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDC000 + 0x0C															

Table 21-34. PHDDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port H Data Direction	**Port H Output/Input** Bits set = Port H output. Bits cleared = Port H input.

21.2.3.17 Port I Data Register

PIDR is the Port I Data Register. The active bits used in this register are Read/Write.

Values written to PIDR are output on the PI pins if the corresponding PIDDR Data Direction bits are set HIGH (port output).

The values read from each bit of this register are determined by the value of the corresponding bit in the Port I Data Direction Register (see Section 21.2.3.19). A read from this register returns either:

- The last bit value written if the bit is configured as an output.
- The current value on the corresponding port input if the bit is configured as an input.

A System Reset clears all bits.

Table 21-35. PIDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port I Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDB000 + 0x00															

Table 21-36. PIDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port I Data	**Port I Input Data** Specifies the Port I input data.

21.2.3.18 Port J Data Register

PJDR is the Port J Data Register. The active bits used in this register are Read/Write.

A read from this register returns the current value on the corresponding port input. A System Reset clears all bits.

Table 21-37. PJDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port J Data							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDB000 + 0x04															

Table 21-38. PJDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port J Data	**Port J Input Data** Specifies the Port J input data.

21.2.3.19 Port I Data Direction Register

PIDDR is the Port I Data Direction Register.

Bits set in PIDDR set the corresponding PI pin to be an output:

- Bit [7] controls pin 136 when the pin is configured as PI7. It does not control pin 136 when the pin is configured as LCDVD7.
- Bit [6] controls pin 137 when the pin is configured as PI6. It does not control pin 137 when the pin is configured as LCDVD6.
- Bit [5] controls pin 138 when the pin is configured as PI5. It does not control pin 138 when the pin is configured as LCDVD5.
- Bit [4] controls pin 139 when the pin is configured as PI4. It does not control pin 139 when the pin is configured as LCDVD4.
- Bit [3] controls pin 141 when the pin is configured as PI3. It does not control pin 141 when the pin is configured as LCDVD3.
- Bit [2] controls pin 142 when the pin is configured as PI2. It does not configure pin 142 when the pin is configured as LCDVD2.
- Bit [1] controls pin 143 when the pin is configured as PI1. It does not configure pin 143 when the pin is configured as LCDVD1.
- Bit [0] controls pin 144 when the pin is configured as PI0. It does not control pin 144 when the pin is configured as LCDVD0.

Clearing a bit configures the pin to be an input. A System Reset clears all bits.

Table 21-39. PIDDR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Port I Data Direction							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFDB000 + 0x08															

Table 21-40. PIDDR Register Definitions

BITS	NAME	FUNCTION
31:8	///	**Reserved** Write the reset value.
7:0	Port I Data Direction	**Port I Output/Input** Bits set = Port I output. Bits cleared = Port I input.

Chapter 22
Controller Area Network

The Controller Area Network (CAN) block pertains to the LH75401 and LH75400 SoC devices only.

The CAN 2.0B Controller is an AMBA-compliant peripheral that connects as a slave to the APB. The CAN Controller is located between the ARM processor and a CAN Transceiver, and is accessed through the AMBA port. Figure 22-1 shows a block diagram of the CAN Controller.

CAN communications are performed serially, at a maximum frequency of 1MB/s, using the TX (transmit) and RX (receive) lines. The RX and TX signals for data reception and transmission provide the communications interface between the CAN Controller and the CAN bus. All peripherals share the TX and RX lines and always see the common incoming and outgoing data.

Data to be transmitted by the CAN Controller is placed in the Transmit Buffer and passed to the Bit Processor, which channels the data onto the TX signal. Messages received by the CAN Controller are filtered by the Acceptance Filter and placed in a 64-byte Receive FIFO. The 64-byte Receive FIFO allows up to five Extended Frame Format (EFF) messages. Together, the Receive Buffer and Receive FIFO allow the CAN Controller to process one message while a second message is being received. The bit rate is controlled by the Bit Timing Logic block and is programmable to 1 Mbit/s.

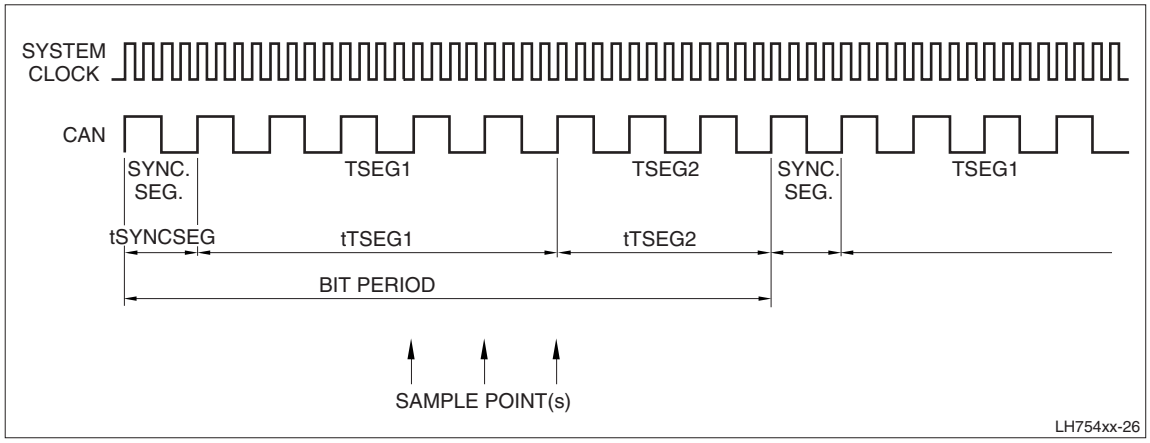


Figure 22-1. CAN Controller Block Diagram

22.1 CAN 2.0B Features

The CAN Controller has the following features:

- Full compliance with 2.0A and 2.0B Bosch specifications
- Support for both 11-bit and 29-bit identifiers
- Support for bit rates up to 1Mbit/s
- 64-byte receive FIFO
- Software-driven bit-rate detection for hot plug-in support
- Single-shot transmission option
- Acceptance filtering
- Listen Only Mode
- Reception of 'own' messages
- Error interrupt generated for each CAN bus error
- Arbitration-lost interrupt with record of bit position
- Read/write error counters
- Last error register
- Programmable error-limit warning.

22.2 CAN Theory of Operation

The CAN Controller has two operating modes:

- Operating Mode, during which data can be transmitted and received.
- Reset Mode, in which special modes of operation, such as changing the values for the Receive and Transmit Error Counters and the Error Warning Limit Register, can be performed.

Reset Mode can be enabled by either setting the Reset Mode in the Mode Register or through a System Reset. The Mode Register also allows special modes to be selected, such as Listen Only Mode and Self Test Mode.

22.2.1 Protocols

The CAN specification consists of a basic protocol and a more advanced 'full' protocol:

- The basic protocol involves a close association between the incoming data and the CPU. With this protocol, the CPU constantly checks incoming data.
- The full protocol supports advanced filtering. Advanced filtering allows the CPU to act upon certain data, freeing the CPU to handle other functions.

'Full' CAN also allows for two different frame formats:

- A standard frame format (referred to as 'CAN A') that is 11 bits long. Each 11-bit identifier can have up to 2,032 unique identifiers, 16 of which are reserved.
- An extended frame format (referred to as 'CAN B') that is 29 bits long. Each 29-bit identifier can have up to 536,870,912 unique identifiers.

Each unique identifier corresponds to a possible peripheral. The identifiers are determined with each design of a network. CAN B networks crash if CAN A devices are used, unless the CAN A devices can ignore the extended format found in CAN B identifiers. Devices that can ignore this extended format known as CAN B passive devices.

22.2.2 Frame Types

The data received and transmitted on a CAN interface are referred to as frames. CAN uses four types of frames, each with its own format.

- The message frame contains data.
- The remote frame requests data from the network.
- The error frame reports errors that occurred at a node.
- The overload frame delays transmission of a CAN frame if the receive node is not ready.

22.2.2.1 Message Frame

The contents of a message frame include:

- Start Bit — The message frame begins with a start bit. As its name implies, this bit indicates the start of the message frame.
- Arbitration Field — The arbitration field follows the start bit. This field is a 29-bit Identifier that consists of two 'recessive' bits, Self-Reception Request (SRR) and IDE, and the Remote Transmission Request (RTR) bit.
- Data Field — The data field follows the arbitration field. The data field contains the actual payload of data, and can range from one to eight bytes of data.
- CRC Field — The CRC field follows the data. This field consists of a 15-bit cyclic-redundancy-code (CRC) that serves as a checksum of the data.
- Acknowledgement Slot — The acknowledgement slot follows the CRC field. The acknowledgement slot verifies that the message frame was correctly received. The transmitter checks for the presence of this acknowledge bit and retransmits the message if no acknowledgement is found.

22.2.2.2 Remote Frame

The Remote frame is a request for information from a particular node. It is a message frame with the RTR bit set, followed by the control field that indicates the number of information bytes (0 to 8) it is requesting. When the node with the information receives the remote frame, it responds by sending the information onto the network.

NOTE: The identifier determines the priority of the node. The lower the identifier, the higher the node priority. Communication onto the bus is non-destructive; this means that if two nodes are requesting the bus at the same time, the node with the higher priority gets access to the bus.

22.2.2.3 Bit Errors

Bit errors occur when either a bit-stuffing violation occurs or when the transmitting node detects that its transmission of data is different than the data it had intended to send. Transmitting nodes normally listen to the outgoing data to ensure that it matches the internal data.

22.2.2.4 Message Errors

Message errors occur if:

- The checksum does not match the data field.
- A predefined bit field is not at its predefined position.

22.2.2.5 Acknowledgement Errors

Acknowledgement errors occur if the transmitter does not receive an acknowledgement after sending a message frame.

Since errors can occur when data is being transmitted or received, the CAN Controller can track these errors and generate an interrupt based on a predetermined value. This type of constant checking for potential errors makes the CAN bus extremely desirable in any electrically hostile environment.

22.2.3 Transmitted and Received Data

Data transmitted to the CAN Controller can follow either the standard (11-bit) or extended (29-bit) frame format. The Transmit Buffer has 13 bytes between addresses 0xFFFC5180 and 0xFFFC51B0 for saving messages consisting of up to eight bits of data. Before writing the data to the buffer, the Transmit Buffer Status (bit 2) should be checked to ensure that the buffer is not busy. If the buffer is busy, the data written is lost and there is no indication that the data has been lost.

Data transmission is initiated by setting the Transmit Request bit or the Self-Reception Request bit in the Command Register. If the Self-Reception bit is used, the Transmit Status bit in the Status Register is set to 1 and the Transmission Request bit is cleared. The output appears on the TX output signal. If bus arbitration is lost or if transmission errors occur while the message is being sent, the CAN Controller tries to send the message again automatically. Data transmission can be aborted by writing to the Command Register, provided the transmission has not yet started. To verify the status of the data transmission, the Status Register can be checked.

Data received by the CAN Controller goes through an Acceptance Filter. The Acceptance Filter passes to the Receive FIFO only those messages that match the ones stored in the Acceptance Filter Registers. Acceptance filtering uses both the Acceptance Code Registers ACCR0-3 and the Acceptance Mask Registers AMR0-3. These two sets of registers can also be used to accept either the standard frame format or the extended frame format.

The Receive FIFO is 64 bytes deep, allowing up to five full extended frame format messages. In addition to the data bytes received, all arbitration bits and the data length code are stored into the corresponding message object. If the FIFO has sufficient space for the data being received, the Data Overrun Interrupt Status bit in the Status Register is set and the data frame being received is discarded. A Data Overrun Interrupt is also generated, if enabled.

22.2.4 Time Delays

The CAN protocol can manage time delays inherent in long bus lengths. It is also efficient at handling differences in clock frequencies for nodes on the bus. Therefore, bit timing is very important.

The CAN standard allows bit timing to be organized into four segments to allow for synchronization:

- A synchronization segment
- The propagation segment
- Phase segment 1
- Phase segment 2.

These segments can be organized into time blocks called 'time quantum'. The time quantum specifies how often the bit timing is sampled to ensure that data is correct. The time quantum is defined as a fixed amount of time that is determined by the pre-scaler and the incoming clock frequency. The CAN Controller can break down the incoming frequency from 1 to 1/64. The segments that define the bit timing can be from 8 to 25 time quanta, according to the CAN specification. The CAN Controller can put these timing segments into a range from 3 to 25 time quanta.

22.2.5 Bus Timing

Because bus timing is critical to the CAN protocol, the CAN Controller has two Bus Timing Registers, BTR0 and BTR1. These registers define the specified time periods used to control the four segments.

- BTR0 defines the CAN bus time quantum.
- BTR1 defines the length of the bit period in terms of CAN bus time quanta and the point at which the incoming data is sampled.

22.2.6 Bus Arbitration

Bus arbitration follows the CAN 2.0A and CAN 2.0B specifications. The bus is always controlled by the node with the highest priority (lowest ID). Only after the bus has been released can the next highest priority node control it.

22.2.7 Error Handling

Errors are handled according to the CAN protocol. There are two types of errors that can occur: transmit errors and receive errors.

The CAN Controller has two counters, one for transmission errors and one for reception errors. These counters automatically increment as errors are encountered.

According to the CAN protocol, once the error count exceeds 255, the bus must be released and the network stopped. This status is reflected in the Status Register. If the error count exceeds 127, the CAN Controller enters an 'Error Passive' state, as defined in the CAN protocol. The Error Warning Limit Register defaults to 96 decimal; if either counter exceeds this value, the Error Status bit in the Status Register is set and an Error Warning Interrupt is generated if enabled.

22.3 CAN Programmer's Model

The base address for the CAN Controller is:

CAN Controller Base Address: 0xFFFC5000

22.3.1 CAN Register Summary

Table 22-1. CAN Register Summary

REGISTER	ADDRESS OFFSET	TYPE		DESCRIPTION	RESET VALUE	NOTES
		OPERATING MODE	RESET MODE			
MOD	0x00	RW	RW	Mode Register	0x01	1
CMR	0x04	W	W	Command Register	0x00	
SR	0x08	R	R	Status Register	0x3C	
IR	0x0C	R	R	Interrupt Register	0x00	
IER	0x10	RW	RW	Interrupt Enable Register	0x00	
///	0x14			Reserved (returns 00h when read)		
BTR0	0x18	R	RW	Bus Timing 0 Register	0x00	
BTR1	0x1C	R	RW	Bus Timing 1 Register	0x00	
///	0x20			Reserved		
///	0x24			Reserved		
///	0x28			Reserved (returns 00h when read)		
ALC	0x02C	R	R	Arbitration Lost Capture Register	0x00	
ECC	0x30	R	R	Error Code Capture Register	0x00	
EWLR	0x34	R	RW	Error Warning Limit Register	0x60	
RXERR	0x38	R	RW	Receive Error Counter Register	0x00	
TXERR	0x3C	R	RW	Transmit Error Counter Register	0x00	
Transmit Buffer	0x40	W	RW	Transmit Frame Information Register (read back from 0x180)		2
	0x44 - 0x70	W	RW	Transmit Data Information (read back from 0x184 - 0x1B0)		2
Receive Window	0x40	R	RW	Receive Frame Information Register		2
	0x44 - 0x70	R	RW	Receive Data Information		2
ACR0	0x40	R	RW	Acceptance Code Register 0	0x00	3
ACR1	0x44	R	RW	Acceptance Code Register 1	0x00	3
ACR2	0x48	R	RW	Acceptance Code Register 2	0x00	3
ACR3	0x4C	R	RW	Acceptance Code Register 3	0x00	3
AMR0	0x50	R	RW	Acceptance Mask Register 0	0x00	3
AMR1	0x54	R	RW	Acceptance Mask Register 1	0x00	3
AMR2	0x58	R	RW	Acceptance Mask Register 2	0x00	3
AMR3	0x5C	R	RW	Acceptance Mask Register 3	0x00	3
RMC	0x74	R	R	Receive Message Counter Register	0x00	
RBSA	0x78	R	RW	Receive Buffer Start Address Register	0x00	
///	0x7C			Reserved	0x7C	
///	0x80 - 0x17C	R	RW	Receive FIFO		
///	0x180 - 0x1B0	R	R	Transmit Buffer		
///	0x1B4 - 0x1FC			Reserved (returns 00h when read)		

NOTES:

1. The Mode Register sets the behavior of the CAN Controller. Bits can be set or reset from the CPU, which sees the Mode Register as part of its Read/Write memory. Reserved bits are read as '0'.
2. Receive data is read from same CAN address where transmit data is written (0x40-0x70). However, transmit data may be read back from 0x180-1B0.
3. The Mode Register sets the behavior of the CAN Controller. Bits can be set or reset from the CPU, which sees the Mode Register as part of its Read/Write memory. Reserved bits are read as '0'.

22.3.2 CAN Register Definitions

22.3.2.1 Mode Register

MOD is the Mode Register. The active bits used in this register are Read/Write.

The MOD Register allows the selection of:

- Acceptance Filter Mode
- Self Test Mode
- Listen Only Mode
- Reset Mode.

Bits [3:1] can be written in both Operating Mode and Reset Mode.

Table 22-2. MOD Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///												AFM	STM	LOM	RM
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
ADDR	0xFFFC5000 + 0x00															

Table 22-3. MOD Register Definitions

BIT	NAME	DESCRIPTION
31:4	///	**Reserved** Write the reset value.
3	AFM	**Acceptance Filter Mode** 0 = Dual filter. Receive data is filtered using two shorter filters. 1 = Single filter. Receive data is filtered using one 4-byte filter.
2	STM	**Self Test Mode** 0 = Normal operation. An acknowledge is required for successful transmission. 1 = Enable Self Test Mode. In this mode, a full node test is possible without any other active node on the bus using the Self-Reception request command.
1	LOM	**Listen Only Mode** 0 = Normal operation. The error counters cannot be changed unless in Reset Mode. 1 = Enable Listen Only Mode. In this mode, the CAN Controller does not send an acknowledge to the CAN bus, even when a message is received successfully.
0	RM	**Reset Mode** 0 = Normal Operation. The CAN Controller returns to Operating Mode on the 1-to-0 transition of this bit. 1 = Reset Mode. Any message currently being transmitted or received is aborted and Reset Mode is entered.

22.3.2.2 Command Register

CMR is the Command Register. The active bits used in this register are Write Only. Setting one or more of the usable bits starts an action in the CAN Controller's transfer layer. Writing a 0 to any bit has no effect.

When using this register:

- All bits return 0 when this register is read.
- At least one system clock cycle is needed between consecutive commands.
- Setting bits [1] and [0] simultaneously results in single-shot transmission of the transmit message without re-transmission in case an error or loss of arbitration occurs.
- Setting bits [4] and [1] simultaneously results in a single-shot transmission of the Transmit message using the Self-Reception feature, again without re-transmission in case of an error or arbitration loss.
- If bits [4] and [0] are set simultaneously, bit [4] is ignored.
- A transmission request made in a previous command cannot be cancelled by setting bit [0] to 0. The requested transmission can only be cancelled by setting bit [1] to 1.

Table 22-4. CMR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											SRR	CDO	RRB	AT	TR
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W
ADDR	0xFFFC5000 + 0x04															

Table 22-5. CMR Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4	SRR	**Self-Reception Request** 0 = No message is transmitted or received simultaneously. 1 = A message is transmitted and received simultaneously.
3	CDO	**Clear Data Overrun** 0 = Do not clear the data overrun condition signaled by the Data Overrun Status bit in the Status Register. 1 = Clear the data overrun condition signaled by the Data Overrun Status bit (bit [1] of the Status Register, described in Section 22.3.2.3). Note that no further data overrun interrupt is generated while the Data Overrun Status bit remains set.
2	RRB	**Release Receive Buffer** 0 = Do not release the Receive Buffer. 1 = Release the Receive Buffer.
1	AT	**Abort Transmission** 0 = Do not cancel the next transmission request. 1 = Cancel the next transmission request, provided it is not already in progress.
0	TR	**Transmission Request** 0 = No message is to be transmitted. 1 = A message is to be transmitted.

22.3.2.3 Status Register

SR is the Status Register. The SR Register reflects the status of the CAN Controller. It appears to the CPU as Read Only memory.

If bits [5] and [4] are both 0, the CAN bus is idle. If both bits are 1, the Controller is waiting to become idle again. After a System Reset, Idle state is entered once the Bus Free sequence (11 consecutive recessive bits) is detected. After a Bus Off event, 128 Bus Free sequences must be received before Idle state is entered.

When using this register:

- If both the Receive Status and the Transmit Status bits are '0', the CAN bus is idle. If both bits are '1', the CAN Controller is waiting to become idle again. After a System Reset, Idle state is entered once the Bus Free sequence (11 consecutive recessive bits) is detected. After a Bus Off event, 128 Bus Free sequences must be received before Idle state is entered.
- For bit [1], the overrun condition is only indicated if the entire message was received. No overrun condition is shown if the message did not complete (e.g., due to an error).

Table 22-6. SR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								BS	ES	TS	RS	TCS	TBS	DOS	RBS
RESET	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x08															

Table 22-7. SR Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	BS	**Bus Status** 0 = CAN Controller is involved in bus activities. 1 = CAN Controller is in the Bus Off state and not involved in bus activities.
6	ES	**Error Status** 0 = Both error counters are below the warning limit. 1 = At least one error counter has reached or exceeded the CPU warning limit defined by the Error Warning Limit Register.
5	TS	**Transmit Status** 0 = No message is being transmitted. 1 = CAN Controller is in the process of transmitting a message.
4	RS	**Receive Status** 0 = Nothing is being received. 1 = CAN Controller is receiving a message.
3	TCS	**Transmission Complete Status** 0 = Last requested transmission has not completed. 1 = Last requested transmission has successfully completed.
2	TBS	**Transmit Buffer Status** 0 = Transmit buffer locked. The CPU cannot access the transmit buffer because a message is either waiting for transmission or is being transmitted. 1 = Transmit buffer released. The CPU may write a message to the transmit buffer.
1	DOS	**Data Overrun Status** 0 = No data overrun has occurred since the last Clear Data Overrun command was given. 1 = Data overrun. A message has been lost because there was not enough space for that message in the receive FIFO. The overrun condition is only indicated if the entire message was received. No overrun condition is shown if the message did not complete — for example, as a result of an error.
0	RBS	**Receive Buffer Status** 0 = Receive buffer empty. No message currently available to be read. 1 = Receive buffer full. One or more complete messages are available to be read from the receive FIFO via the receive buffer.

22.3.2.4 Interrupt Register

IR is the Interrupt Register. The IR Register allows the source of an interrupt to be identified. When one or more bits of this register are set, the CAN Controller sends an interrupt to the CPU.

The IR Register appears to the CPU as Read Only memory. After the register has been read by the CPU, all bits except Receive Interrupt are reset.

Table 22-8. IR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								BEI	ALI	EPI	WUI	DOI	EI	TI	RI
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x0C															

Table 22-9. IR Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	BEI	**Bus Interrupt Error** 1 = CAN Controller detects an error on the CAN bus, provided bit [7] of the Interrupt Enable Register is set (see Section 22.3.2.5).
6	ALI	**Arbitration Lost Interrupt** 1 = CAN Controller loses arbitration and becomes a receiver, provided bit [6] of the Interrupt Enable Register is set (see Section 22.3.2.5).
5	EPI	**Error Passive Interrupt** 1 = CAN Controller re-enters Error Active state after being in Error Passive state or when at least one error counter exceeds the protocol-defined level of 127, provided bit [5] of the Interrupt Enable Register is set (see Section 22.3.2.5).
4	WUI	**Wake-Up Interrupt** 1 = Bus activity is detected, provided bit [4] of the Interrupt Enable Register is set (see Section 22.3.2.5). A wake-up interrupt is also generated if the CPU tries to set bit [4] of the MOD Register while the CAN Controller is involved in bus activities or a CAN interrupt is pending.
3	DOI	**Data Overrun Interrupt** Set on a 0-to-1 transition of bit [1] of the CAN Status Register, provided bit [3] of the Interrupt Enable Register is set (see Section 22.3.2.5).
2	EI	**Error Warning Interrupt** Set on every change (set or clear) of either bit [7] or bit [6] of the Status Register, provided bit [2] of the Interrupt Enable Register is set (see Section 22.3.2.5).
1	TI	**Transmit Interrupt** Set when bit [2] of the Status Register changes from 0 to 1 (released), provided bit [1] of the Interrupt Enable Register is set (see Section 22.3.2.3 and Section 22.3.2.5).
0	RI	**Receive Interrupt** Set when the receive buffer contains one or more messages, provided bit [0] of the Interrupt Enable Register is set (see Section 22.3.2.5). Cleared when the Release Receive Buffer command (bit [2] of the Command Register, described in Section 22.3.2.2) is issued, provided there is no further data to read in the receive buffer. The RI bit, when enabled, mirrors bit [0] of the Status Register (described in Section 22.3.2.3). Consequently, it is not cleared automatically when the Interrupt Register is read.

22.3.2.5 Interrupt Enable Register

IER is the Interrupt Enable Register. This register selects the events that are indicated to the CPU through an interrupt being generated. It appears to the CPU as Read/Write memory.

Table 22-10. IER Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								BEIE	ALIE	EPIE	///	DOIE	EIE	TIE	RIE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC5000 + 0x10															

Table 22-11. IER Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	BEIE	**Bus Interrupt Error Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when a bus error is detected.
6	ALIE	**Arbitration Lost Interrupt Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when the CAN Controller loses arbitration.
5	EPIE	**Error Passive Interrupt Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when the error status of the CAN Controller changes from error active to error passive or vice versa.
4	///	**Reserved** Write the reset value.
3	DOIE	**Data Overrun Interrupt Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when bit [1] of the Status Register is set (see Section 22.3.2.3).
2	EIE	**Error Warning Interrupt Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when the Bus Status bit ([bit [7]) or the Error Status bit (bit [6]) of the Status Register change (see Section 22.3.2.3).
1	TIE	**Transmit Interrupt Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when a message has been successfully transmitted or when the transmit buffer is accessible.
0	RIE	**Receive Interrupt Enable** 0 = Interrupt is disabled. 1 = An interrupt is generated when bit [0] of the Status Register is full (see Section 22.3.2.3). This bit influences bit [0] of the Interrupt Register (described in Section 22.3.2.4) and the external interrupt output, NINT. If RIE clears, NINT becomes inactive (HIGH) immediately if no other interrupt is pending.

22.3.2.6 Bus Timing Register 0

BTR0 is one of two CAN Timing Registers (BTR1 is the other). Together, these two registers define the structure of the bit period.

The BTR0 Register defines the values of the Synchronization Jump Width (SJW) and the Bit Rate Prescaler (BRP). This register can only be written to in Reset Mode. In Operating Mode, it is Read Only.

Table 22-12. BTR0 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								SJW	SJW	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x18															

Table 22-13. BTR0 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:6	SJW	**Synchronization Jump Width** Defines the maximum number of clock cycles by which a bit period can be shortened or lengthened in attempting to re-synchronize on the relevant signal edge (recessive to dominant) of the current transmission. The number of clock cycles range from 0 to 3.
5:0	BRP4 - BRP0	**Bit Rate Prescaler** Defines the period (time quantum) of the CAN clock tSCL as a multiple of the system clock period. The time quantum of the CAN clock is given by: tSCL =2 × tCLK × ((32 × BRP.5) + (16 × BRP.4) + (8 × BRP.3) + (4 × BRP.2) + (2 × BRP.1) + BRP.0 + 1) where tCLK = time period of the system clock frequency = 1/*f* SYSTEM CLK

22.3.2.7 Bus Timing Register 1

BTR1 is one of two CAN timing registers (BTR0 is the other). Together, these two registers define the structure of the bit period.

The BTR1 Register defines the length of the bit period, the location of the sample point and the number of samples to be taken at each sample point. This register can only be written to in Reset Mode. In Operating Mode, it is Read Only.

Table 22-14. BTR1 Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								SAM	TSEG2.2	TSEG2.1	TSEG2.0	TSEG1.3	TSEG1.2	TSEG1.1	TSEG1.0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x1C															

Table 22-15. BTR1 Register Definitions

BITS	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7	SAM	**Sampling** 0 = Bus will be sampled once. This is recommended for high-speed buses (SAE class C). 1 = Bus will be sampled three times. This is recommended for low- and medium-speed buses (class A or B).
6:0	TSEG2.2 - TSEG1.0	**TSEG1 and TSEG2** Determines the number of CAN clock cycles (time quanta) per bit period and the location of the sample point, as given by the parameters tSYNCSEG, tTSEG1, and tTSEG2 in Figure 22-2. Program this value according to the number of CAN clock cycles minus 1. For example, if the number of CAN clock cycles for TSEG1 is 6, program TSEG1 with a value of 5.

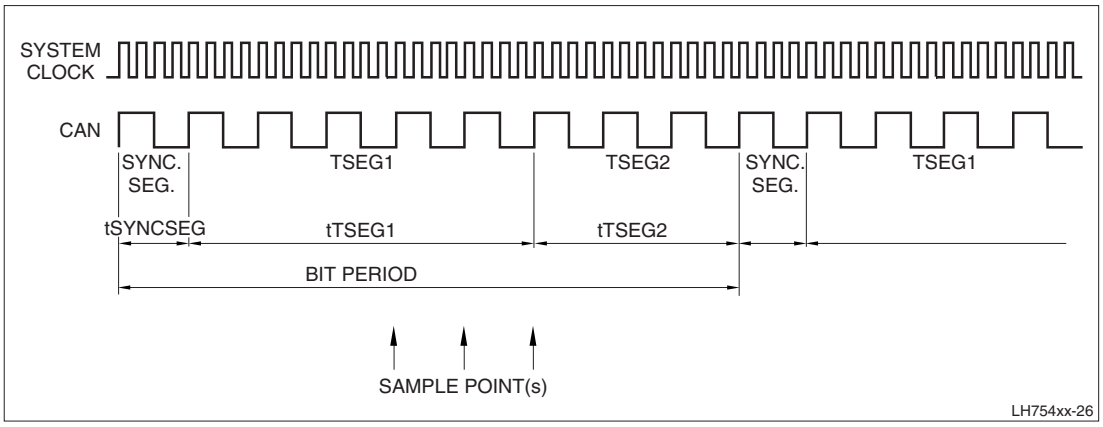


Figure 22-2. General Structure of a Bit Period

22.3.2.8 Arbitration Lost Capture Register

ALC is the Arbitration Lost Capture Register. This register records the bit position at which arbitration was lost.

When bus arbitration is lost:

- An Arbitration Lost Interrupt is generated (if enabled) and the current bit position of the Bit Processor is captured into this Arbitration Lost Capture Register.
- The contents of this register are maintained until the register has been read by user software. Then the capture mechanism is activated again.

The Arbitration Lost Capture Register appears to the CPU as Read Only memory. The Reserved bits always return '0'.

Table 22-16. ALC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											Arbitration Loss				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x2C															

Table 22-17. ALC Register Definitions

BITS	NAME	DESCRIPTION
31:5	///	**Reserved** Write the reset value.
4:0	Arbitration Loss	**Arbitration Loss** Specifies a value from Table 22-18, indicating the cause for losing arbitration.

Table 22-18. Arbitration Losses

ALC[4:0]	DECIMAL VALUE	DESCRIPTION	NOTES
0 0 0 0 0	00	Arbitration lost in 1st bit of identifier (ID.28)	
0 0 0 0 1	01	Arbitration lost in 2nd bit of identifier (ID.27)	
0 0 0 1 0	02	Arbitration lost in 3rd bit of identifier (ID.26)	
0 0 0 1 1	03	Arbitration lost in 4th bit of identifier (ID.25)	
0 0 1 0 0	04	Arbitration lost in 5th bit of identifier (ID.24)	
0 0 1 0 1	05	Arbitration lost in 6th bit of identifier (ID.23)	
0 0 1 1 0	06	Arbitration lost in 7th bit of identifier (ID.22)	
0 0 1 1 1	07	Arbitration lost in 8th bit of identifier (ID.21)	
0 1 0 0 0	08	Arbitration lost in 9th bit of identifier (ID.20)	
0 1 0 0 1	09	Arbitration lost in 10th bit of identifier (ID.19)	
0 1 0 1 0	10	Arbitration lost in 11th bit of identifier (ID.18)	
0 1 0 1 1	11	Arbitration lost in SRTR bit	1
0 1 1 0 0	12	Arbitration lost in IDE bit	
0 1 1 0 1	13	Arbitration lost in 12th bit of identifier (ID.17)	2
0 1 1 1 0	14	Arbitration lost in 13th bit of identifier (ID.16)	2
0 1 1 1 1	15	Arbitration lost in 14th bit of identifier (ID.15)	2
1 0 0 0 0	16	Arbitration lost in 15th bit of identifier (ID.14)	2
1 0 0 0 1	17	Arbitration lost in 16th bit of identifier (ID.13)	2
1 0 0 1 0	18	Arbitration lost in 17th bit of identifier (ID.12)	2
1 0 0 1 1	19	Arbitration lost in 18th bit of identifier (ID.11)	2
1 0 1 0 0	20	Arbitration lost in 19th bit of identifier (ID.10)	2
1 0 1 0 1	21	Arbitration lost in 20th bit of identifier (ID.9)	2
1 0 1 1 0	22	Arbitration lost in 21st bit of identifier (ID.8)	2
1 0 1 1 1	23	Arbitration lost in 22nd bit of identifier (ID.7)	2
1 1 0 0 0	24	Arbitration lost in 23rd bit of identifier (ID.6)	2
1 1 0 0 1	25	Arbitration lost in 24th bit of identifier (ID.5)	2
1 1 0 1 0	26	Arbitration lost in 25th bit of identifier (ID.4)	2
1 1 0 1 1	27	Arbitration lost in 26th bit of identifier (ID.3)	2
1 1 1 0 0	28	Arbitration lost in 27th bit of identifier (ID.2)	2
1 1 1 0 1	29	Arbitration lost in 28th bit of identifier (ID.1)	2
1 1 1 1 0	30	Arbitration lost in 29th bit of identifier (ID.0)	2
1 1 1 1 1	31	Arbitration lost in RTR bit	2

NOTES:
1. RTR Bit in Standard Frame Format (SFF) messages.
2. Applies to Extended Frame Format messages only.

22.3.2.9 Error Code Capture Register

ECC is the Error Code Capture Register. The ECC Register contains information about the type and location of errors on the bus.

When a bus error occurs:

- A Bus Error Interrupt is generated (if enabled) and the current bit position of the Bit Processor is captured into this Error Code Capture Register.
- The contents of this register are maintained until the register has been read by user software. Then the capture mechanism is activated again.

The Error Code Capture Register appears to the CPU as Read Only memory.

Table 22-19. ECC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								Error Code		Direction	Segment Code				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R		R	R				
ADDR	0xFFFC5000 + 0x30															

Table 22-20. ECC Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:6	Error Code	**Error Code** Shows the error code that occurred (see Table 22-21).
5	Direction	**Error Direction** 0 = Error occurred during transmission. 1 = Error occurred during reception.
4:0	Segment Code	**Segment Code** Shows the segment code (see Table 22-22).

Table 22-21. Error Code

ERROR CODE BITS [7:6]	DESCRIPTION
0 0	Bit error
0 1	Form error
1 0	Stuff error
1 1	Other types of errors

Table 22-22. Segment Code

SEGMENT CODE BITS [4:0]	DESCRIPTION
0 0 0 1 1	Start of frame
0 0 0 1 0	ID.28 to ID.21
0 0 1 1 0	ID.20 to ID.18
0 0 1 0 0	SRTR bit
0 0 1 0 1	IDE bit
0 0 1 1 1	ID.17 to ID.13
0 1 1 1 1	ID.12 to ID.5
0 1 1 1 0	ID.4 to ID.0
0 1 1 0 0	RTR bit
0 1 1 0 1	Reserved bit 1
0 1 0 0 1	Reserved bit 0
0 1 0 1 1	Data Length code
0 1 0 1 0	Data field
0 1 0 0 0	CRC sequence
1 1 0 0 0	CRC delimiter
1 1 0 0 1	Acknowledge
1 1 0 1 1	Acknowledge delimiter
1 1 0 1 0	End of frame
1 0 0 1 0	Intermission
1 0 0 0 1	Active error flag
1 0 1 1 0	Passive error flag
1 0 0 1 1	Tolerate dominant bits
1 0 1 1 1	Error delimiter
1 1 1 0 0	Overload flag

22.3.2.10 Error Warning Limit Register

ELWR is the Error Warning Limit Register. The ELWR Register defines the number of errors after which an Error Warning Interrupt is generated, if enabled. The EWLR Register can only be written to in Reset Mode. In Operating Mode, it is Read Only.

Changes made during Reset Mode take effect when the CAN Controller returns to Operating Mode. The default value of this register after System Reset is 0110000 (i.e., 96).

Table 22-23. EWLR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								EWL.7	EWL.6	EWL.5	EWL.4	EWL.3	EWL.2	EWL.1	EWL.0
RESET	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x34															

Table 22-24. EWLR Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	EWL.7 - EWL.0	**Errors Before Interrupt** Specifies the number of errors after which an error-warning interrupt should be generated.

22.3.2.11 Receive Error Counter Register

RXERR is the Receive Error Counter Register. The RXERR Register records the current value of the Receive Error Counter. After a System Reset or when a Bus Off event occurs, this register is automatically set to '0'. This register works with the Transmit Error Counter (described in Section 22.3.2.12) to create a metric for communication quality based on historic performance.

This register can only be written to in Reset Mode. Changes made during Reset Mode only go into effect when the CAN Controller returns to Operating Mode.

In Operating Mode, this register is Read Only. Writing to this register has no effect when the CAN Controller is in Bus Off state.

Table 22-25. RXERR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///								RXERR.7	RXERR.6	RXERR.5	RXERR.4	RXERR.3	RXERR.2	RXERR.1	RXERR.0
RESET	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x38															

Table 22-26. RXERR Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	RXERR 7 - RXERR.0	**Receive Error Counter Value** Specifies the current value of the Receive Error Counter.

22.3.2.12 Transmit Error Counter Register

TXERR is the Transmit Error Counter Register. The TXERR Register records the current value of the Transmit Error Counter. This register works with the Receive Error Counter (described in Section 22.3.2.11) to create a metric for communication quality based on historic performance.

After a System Reset, the Transmit Error Counter is automatically set to '0'. However, if the CAN Controller enters Reset Mode as a result of a Bus Off event, the register is initialized instead to 127 to count the minimum protocol-defined time before the CAN Controller can take part in further transmission on the CAN bus (128 occurrences of the Bus Free sequence of 11 consecutive recessive bits). Reading the Transmit Error Counter during this time indicates the status of the Bus Off recovery.

The TXERR Register can only be written to in Reset Mode. In Operating Mode, this register appears to the CPU as Read Only memory.

While in Bus Off state, writing a value in the range from 0 to 254 to the TXERR Register clears the Bus Off flag. The CAN Controller then waits one Bus Free sequence after the Reset Mode clears.

Writing 255 to the TXERR Register initiates a CPU-driven bus-off event. Note that a CPU-forced content change of the transmit error counter is only possible if Reset Mode was entered previously. An error or bus status change (see Section 22.3.2.3), an error warning, or an error passive interrupt forced by the new register content will not occur until the Reset Mode is canceled again. After leaving Reset Mode, the new transmit counter content is interpreted and the Bus Off event is performed as if it was forced by a bus error event. This means that:

- Reset Mode is entered again.
- The transmit error counter is initialized to 127.
- The receive counter is cleared.
- The relevant status and interrupt register bits are set.

Clearing of Reset Mode now performs the protocol-defined Bus Off recovery sequence (waiting for 128 occurrences of the Bus Free signal).

If Reset Mode is entered again before the Bus Off recovery completes (TXERR > 0), Bus Off stays active, with TXERR frozen until the CAN Controller returns to Operating Mode.

22.3.2.13 Transmit Buffer

The CAN Transmit Buffer has a length of 13 bytes. It accommodates one Transmit message of up to eight data bytes.

Write Only access to the Transmit Buffer is provided in Operating Mode using CAN offsets 0x40 - 0x70.

Table 22-27 shows the global layout of the Transmit Buffer. It is essential to distinguish between SFF and EFF messages.

The CAN transmit buffer is divided into descriptor and data fields. The first byte of the descriptor field holds frame information. It describes the frame format (SFF or EFF), remote or data frame, and data length. This is then followed by either two identifier bytes for SFF messages or four bytes for EFF messages. The data field contains up to eight data bytes.

NOTE: Direct read-only access to the Transmit Buffer is possible using the CAN offset space from 0x180 to 0x1B0.

Table 22-27. CAN Transmit Buffer

STANDARD FRAME FORMAT		EXTENDED FRAME FORMAT	
CAN OFFSET	**FIELD**	**CAN OFFSET**	**FIELD**
0x40	TX Frame Information	0x40	TX Frame Information
0x44	TX Identifier 1	0x44	TX Identifier 1
0x48	TX Identifier 2	0x48	TX Identifier 2
0x4C	TX Data Byte 1	0x4C	TX Identifier 3
0x50	TX Data Byte 2	0x50	TX Identifier 4
0x54	TX Data Byte 3	0x54	TX Data Byte 1
0x58	TX Data Byte 4	0x58	TX Data Byte 2
0x5C	TX Data Byte 5	0x5C	TX Data Byte 3
0x60	TX Data Byte 6	0x60	TX Data Byte 4
0x64	TX Data Byte 7	0x64	TX Data Byte 5
0x68	TX Data Byte 8	0x68	TX Data Byte 6
0x6C	(Unused)	0x6C	TX Data Byte 7
0x70	(Unused)	0x70	TX Data Byte 8

22.3.2.14 Transmit Buffer Descriptor Field

Table 22-28 and Table 22-29 show the bit layout of the Descriptor Field of the Transmit Buffer. Table 22-28 shows the SFF message format, while Table 22-29 shows the EFF message format.

Table 22-28. Transmit Frame (SFF)

CAN OFFSET	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0X40	FF	RTR	x[1]	x[1]	DLC.3	DLC.2	DLC.1	DLC.0
0X44	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0X48	ID.20	ID.19	ID.18	x[2]	x[1]	x[1]	x[1]	x[1]

NOTES:
1. Don't care, but '0' is recommended for compatibility with the Receive Buffer in case the Self-Reception or the Self Test option is used.
2. Don't care, but matching the RTR bit in the Receive Buffer is recommended in case the Self-Reception or the Self Test option is used.

Table 22-29. Transmit Frame (EFF)

CAN OFFSET	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0x40	FF	RTR	x[1]	x[1]	DLC.3	DLC.2	DLC.1	DLC.0
0x44	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	ID.20	ID.19	ID.18	ID.17	ID.16	ID.15	ID.14	ID.13
0x4C	ID.12	ID.11	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5
0x50	ID.4	ID.3	ID.2	ID.1	ID.0	x[2]	x[1]	x[1]

NOTES:
1. Don't care, but '0' is recommended for compatibility with the Receive Buffer in case the Self-Reception or the Self Test option is used.
2. Don't care, but matching the RTR bit in the Receive Buffer is recommended in case the Self-Reception or the Self Test option is used.

Table 22-30. Transmit Frame/Receive Frame Definitions

NAME	DESCRIPTION
///	**Reserved**
FF	**Frame Format** Selects the type of frame format to be used for transmission. 0 = Uses Standard Frame Format. 1 = Uses Extended Frame Format.
RTR	**Remote Transmission Request** 0 = Indicates a data frame, as defined in the CAN protocol. 1 = Indicates a remote frame.
x[1]	Don't care, but 0 is recommended for compatibility with the Receive Buffer, in case the Self-Reception or Self Test Option is used.
DLC.3 - DLC.0	**Data Length Code** Specifies the number of data bytes included in message being sent. The maximum number of data bytes that can be included in a frame is eight; therefore, values greater than eight are interpreted as eight automatically. Consult the CAN 2.0B specification for a complete description about data-length encoding. Note, too, that although no data bytes are transmitted from the local host in the case of a remote frame transmission, the data length of the remote frame should still be specified to avoid bus errors if two CAN Controllers start a remote frame transmission with the same identifier simultaneously.
ID	**Identifier** Acts as the message's name, is used in a receiver for acceptance filtering, and determines the bus access priority. The lower the binary value of the identifier the higher the priority. • In SFF, the identifier consists of 11 bits (ID.28 to ID.18). • In EFF, messages the identifier consists of 29 bits (ID.28 to ID.0). ID.28 is the most-significant bit and is transmitted first on the bus.
Data Field	**Data Field** Comprises the number of data bytes defined by the data length code. The most-significant bit of data byte 1 at CAN address 0xFFFC5064 (SFF) or CAN address 0xFFFC5084 (EFF) is transmitted first.

22.3.2.15 CAN Receive Buffer

The Receive Buffer provides the window through which the CPU accesses the Receive FIFO. Like the Transmit Buffer, the Receive Buffer has a length of 13 bytes (enough to accommodate one Receive message of up to eight data bytes).

Read Only access to the Receive Buffer is provided in Operating Mode using CAN offsets 0x40 - 0x70.

The layout of the Receive Buffer is similar to, and compatible with, the Transmit Buffer. It is essential to distinguish between SFF and EFF messages.

The Receive Buffer is subdivided into descriptor and data fields. The first byte of the descriptor field holds frame information. It describes the frame format (SFF or EFF), specifies remote or data frame and gives the data length. This is then followed by either two identifier bytes for SFF messages or four bytes for EFF messages. The data field contains up to eight data bytes.

Table 22-31. CAN Receive Buffer

STANDARD FRAME FORMAT		EXTENDED FRAME FORMAT	
CAN OFFSET	**FIELD**	**CAN OFFSET**	**FIELD**
0x40	RX Frame Information	0x40	RX Frame Information
0x44	RX Identifier 1	0x44	RX Identifier 1
0x48	RX Identifier 2	0x48	RX Identifier 2
0x4C	RX Data Byte 1	0x4C	RX Identifier 3
0x50	RX Data Byte 2	0x50	RX Identifier 4
0x54	RX Data Byte 3	0x54	RX Data Byte 1
0x58	RX Data Byte 4	0x58	RX Data Byte 2
0x5C	RX Data Byte 5	0x5C	RX Data Byte 3
0x60	RX Data Byte 6	0x60	RX Data Byte 4
0x64	RX Data Byte 7	0x64	RX Data Byte 5
0x68	RX Data Byte 8	0x68	RX Data Byte 6
0x6C	(Unused)	0x6C	RX Data Byte 7
0x70	(Unused)	0x70	RX Data Byte 8

22.3.2.16 Receive Buffer Descriptor Field

Table 22-32 and Table 22-33 show the bit layout of the Descriptor Field of the Receive Buffer. Table 22-32 shows the SFF message format, while Table 22-33 shows the EFF message format.

Table 22-32. Receive Frame (SFF)

CAN OFFSET	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0x40	FF	RTR	0	0	DLC.3	DLC.2	DLC.1	DLC.0
0x44	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	ID.20	ID.19	ID.18	RTR	0	0	0	0

Table 22-33. Receive Frame (EFF)

CAN OFFSET	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0x40	FF	RTR	0	0	DLC.3	DLC.2	DLC.1	DLC.0
0x44	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	ID.20	ID.19	ID.18	ID.17	ID.16	ID.15	ID.14	ID.13
0x4C	ID.12	ID.11	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5
0x50	ID.4	ID.3	ID.2	ID.1	ID.0	RTR	0	0

NOTE: The received data length code in the frame information byte (CAN Offset 0x40) represents the length of the data sent, which may be greater than eight bytes. However, the maximum number of data bytes received will be eight.

22.3.2.17 Acceptance Code Registers

Registers ACR0, ACR1, ACR2, and ACR3 are the Acceptance Code Registers. These 8-bit registers record the bit patterns used by the Acceptance Filter, along with the masks provided by the Acceptance Mask Registers, to filter received data.

The way in which these bit patterns are applied depends on whether a single filter or dual filters are used and on whether the data is in SFF or EFF format (see Section 22.3.4).

The registers are only accessible for Read/Write access in Reset Mode.

22.3.2.18 Acceptance Mask Registers (AMR0 - AMR3)

Registers AMR0, AMR1, AMR2, and AMR3 are the Acceptance Mask Registers. These 8-bit registers record the mask patterns applied by the Acceptance Filter when filtering received data. Register values of zero identify the bits of the incoming data bytes that are required to match the bit values in the corresponding Acceptance Code Registers. Register values of 1 mark the bits as 'don't care'.

The bits of incoming data identified by these masks is based on whether:

- A single filter or dual filters are being used
- The data is in SFF or EFF format (see Section 22.3.4).

The registers are only accessible for Read/Write access in Reset Mode.

22.3.2.19 Receive Message Counter Register

RMC is the Receive Message Counter Register. The RMC Register records the number of messages currently available in the Receive FIFO. It increments automatically with each Receive event and decrements with each Release Receive Buffer command. It is available for Read Only access in both Operating Mode and Reset Mode.

The register is reset to 00h by either a System Reset or a Software Reset.

Table 22-34. RMC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											RMC.4	RMC.3	RMC.2	RMC.1	RMC.0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC5000 + 0x74															

Table 22-35. RMC Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	RMC.4 - RMC.0	**Receive FIFO Messages** Specifies the number of messages currently available in the Receive FIFO.

22.3.2.20 Receive Buffer Start Address Register

RBSA is the Receive Buffer Start Address Register. The active bits used in this register are Read/Write.

This register records the current location of the RX FIFO Read Pointer within the 64-byte RX FIFO as a value between 0 and 63.

- Location 0 corresponds to CAN Offset 0x80.
- Location 63 corresponds to CAN Offset 0x17C.

This register is reset to 00h by a System Reset, but is left unchanged by a Software Reset (which also does not change the FIFO contents). However, the RX FIFO Write Pointer is set by a Software Reset to the value of the RX FIFO Read Pointer. As a result, the data accessed by the Receive Buffer following a Software Reset is overwritten by the next message to be recorded in the RX FIFO.

NOTE: It is only possible to write to this register in Reset Mode.

Table 22-36. RBSA Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///										RBSA					
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC5000 + 0x78															

Table 22-37. RBSA Register Definitions

BIT	NAME	DESCRIPTION
31:8	///	**Reserved** Write the reset value.
7:0	RMC.4 - RMC.0	**RX FIFO Read Pointer Location** Specifies the current location of the RX FIFO Read Pointer within the 64-byte RX FIFO.

22.3.3 CAN Reset Mode

There are two ways for the CAN Controller to enter Reset Mode:

- Setting the Reset Mode bit in the Mode Register (see Section 22.3.2.1) so that the current transmission/reception of any message is aborted. This causes the CAN Controller to enter Reset Mode on the next positive edge of the system clock.
- Going to Bus Off state (bit [7] of the Status Register set to 1) following a System Reset. This situation occurs, for example, when the transmit error counter exceeds its maximum count of 255.

To return the CAN Controller to Operating Mode, clear the Reset Mode bit (bit [0] of the Mode Register. The CAN Controller does not send or receive any data until it detects either:

- A Bus Free sequence of 11 recessive bits on the bus, if the reset was triggered by a System Reset.
- 128 such sequences, if the reset was caused by the bus status going to Bus Off.

Table 22-38 describes the effect of the reset on the CAN Controller registers. In this table, 'x' means the reset has no effect on the value of these registers or bits.

Table 22-38. Effect of Reset on CAN Controller Registers

REGISTER	BIT	SYMBOL	NAME	RESET VALUE		NOTES
				SYSTEM	SOFTWARE	
Mode	7:5		Reserved	0 (reserved)	0 (reserved)	
	4		Reserved	0 (reserved)	0 (reserved)	
	3	AFM	Acceptance Filter Mode	0 (dual filters)	x	
	2	STM	Self Test Mode	0 (normal)	x	
	1	LOM	Listen Only Mode	0 (normal)	x	
	0	RM	Reset Mode	1 (present)	1 (present)	1
Command	7:5		Reserved	0 (reserved)	0 (reserved)	
	4	SRR	Self-Reception Request	0 (absent)	0 (absent)	
	3	CDO	Clear Data Overrun	0 (no action)	0 (no action)	
	2	RRB	Release Receive Buffer	0 (no action)	0 (no action)	
	1	AT	Abort Transmission	0 (absent)	0 (absent)	
	0	TR	Transmission Request	0 (absent)	0 (absent)	
Status	7	BS	Bus Status	0 (bus-on)	x	
	6	ES	Error Status	0 (ok)	x	
	5	TS	Transmit Status	1 (wait idle)	1 (wait idle)	
	4	RS	Receive Status	1 (wait idle)	1 (wait idle)	
	3	TCS	Transmission Complete Status	1 (complete)	X	
	2	TBS	Transmit Buffer Status	1 (released)	1 (released)	
	1	DOS	Data Overrun Status	0 (absent)	0 (absent)	
	0	RBS	Receive Buffer Status	0 (empty)	0 (empty)	

Table 22-38. Effect of Reset on CAN Controller Registers (Cont'd)

REGISTER	BIT	SYMBOL	NAME	RESET VALUE		NOTES
				SYSTEM	SOFTWARE	
Interrupt	7	BEI	Bus Error Interrupt	0 (reset)	0 (reset)	
	6	ALI	Arbitration Lost Interrupt	0 (reset)	0 (reset)	
	5	EPI	Error Passive Interrupt	0 (reset)	0 (reset)	
	4	WUI	Wake-Up Interrupt	0 (reset)	0 (reset)	
	3	DOI	Data Overrun Interrupt	0 (reset)	0 (reset)	
	2	EI	Error Warning Interrupt	0 (reset)	x	2
	1	TI	Transmit Interrupt	0 (reset)	0 (reset)	
	0	RI	Receive Interrupt	0 (reset)	0 (reset)	
Interrupt Enable	7	BEIE	Bus Error Interrupt Enable	0	x	
	6	ALIE	Arbitration Lost Interrupt Enable	0	x	
	5	EPIE	Error Passive Interrupt Enable	0	x	
	4	WUIE	Wake-Up Interrupt Enable	0	x	
	3	DOIE	Data Overrun Interrupt Enable	0	x	
	2	EIE	Error Warning Interrupt Enable	0	x	
	1	TIE	Transmit Interrupt Enable	0	x	
	0	RIE	Receive Interrupt Enable	0	x	
Bus Timing 0	7	SJW.1	Synchronization Jump Width 1	0	x	
	6	SJW.0	Synchronization Jump Width 0	0	x	
	5	BRP.5	Bit Rate Prescaler 5	0	x	
	4	BRP.4	Bit Rate Prescaler 4	0	x	
	3	BRP.3	Bit Rate Prescaler 3	0	x	
	2	BRP.2	Bit Rate Prescaler 2	0	x	
	1	BRP.1	Bit Rate Prescaler 1	0	x	
	0	BRP.0	Bit Rate Prescaler 0	0	x	
Bus Timing 1	7	SAM	Sampling	0	x	
	6	TSEG2.2	Time Segment 2.2	0	x	
	5	TSEG2.1	Time Segment 2.1	0	x	
	4	TSEG2.0	Time Segment 2.0	0	x	
	3	TSEG1.3	Time Segment 1.3	0	x	
	2	TSEG1.2	Time Segment 1.2	0	x	
	1	TSEG1.1	Time Segment 1.1	0	x	
	0	TSEG1.0	Time Segment 1.0	0	x	
Arbitration Lost Capture		ALC	Arbitration Lost Capture	00h	x	
Error Code Capture		ECC	Error Code Capture	00h	x	
Error Warning Limit		EWLR	Error Warning Limit Register	96 (decimal)	x	
Receive Error Counter		RXERR	Receive Error Counter	0 (reset)	x	3
Transmit Error Counter		TXERR	Transmit Error Counter	0 (reset)	x	3

Table 22-38. Effect of Reset on CAN Controller Registers (Cont'd)

REGISTER	BIT	SYMBOL	NAME	RESET VALUE		NOTES
				SYSTEM	SOFTWARE	
Transmit Buffer		TXB	Transmit Buffer	x	x	
Receive Buffer		RXB	Receive Buffer	x	x	
Acceptance Code Registers 0-3		ACR0 - ACR3	Acceptance Code Registers 0-3	00h	x	
Acceptance Mask Registers 0-3	—	AMR0 - AMR3	Acceptance Mask Registers 0-3	00h	x	
Receive Message Count	—	RMC	Receive Message Count	0	0	
Receive Buffer Start Address	—	RBSA	Receive Buffer Start Address	00h	x	
Receive FIFO	—	—	Receive FIFO	x	x	

NOTES
1. Software Reset (MOD.0) or Bus Off.
2. If the Reset Mode was entered due to a Bus Off condition, the Error Warning Interrupt will be set (if enabled).
3. If the Reset Mode was entered due to a Bus Off condition, the Receive Error Counter is cleared and the Transmit Error Counter is initialized to 127 to count-down the CAN-defined Bus Off recovery time consisting of 128 occurrences of 11 consecutive recessive bits.

22.3.4 CAN Acceptance Filtering

The CAN Controller filters the incoming data stream, discarding any message that does not have the required bit pattern in its identifier.

The bit pattern against which the message identifier is recorded in the Acceptance Code Registers (ACR0 through ACR3), masked by the values recorded in the Acceptance Mask Registers (AMR0 through AMR3).

- A value of 0 in AMR0 through AMR3 identifies the bits at the corresponding positions in ACR0 through ACR3, which must be matched in the message identifier.
- A value 1 identifies the corresponding bits as 'don't care'.

The bit patterns recorded in the ACR0 - ACR3 Registers can be used as either a single 4-byte filter or as two shorter filters. The selection is made through the AFM bit (bit [3]) of the Mode Register (see Section 22.3.2.1).

- If AFM = 1, a single filter is applied.
- if AFM = 0, two filters are applied. When two filters are used, the incoming message is accepted if its identifier matches either filter.

The way in which the bit patterns defined by ACR0 through ACR3 are applied further depend on whether the incoming message is in Standard Frame Format (SFF) or Extended Frame Format (EFF) as shown in Table 22-39, Table 22-40, Table 22-41, and Table 22-42.

Table 22-39. Standard Frame Format, Single Filter: Receive Buffer and Filter

DATA (WITH OFFSET)					
0x44		**0x48**		**0x4C**	**0x50**
ID.28 … ID.21	ID.20 … ID18	RTR	x x x x *(not matched)*	Data Byte 1	Data Byte 2
SINGLE FILTER					
ACR0[7:0]	ACR1[7:4]		(ACR1[3:0] unused)	ACR2[7:0]	ACR3[7:0]
AMR0[7:0]	AMR1[7:4]		(AMR1[3:0] unused)	AMR2[7:0]	AMR3[7:0]

NOTE: If data bytes do not have to be matched, AMR2 and AMR3 should be set to FFh.

Table 22-40. Standard Frame Format, Dual Filters: Receive Buffer and Filters

DATA (WITH OFFSET)						
0x44		**0x48**		**0x4C**		**0x50**
ID.28 … ID.21	ID.20 … ID18	RTR	x x x x (not matched)	Data Byte 1 [7:4]	Data Byte 1 [3:0]	Data Byte 2 *(not matched)*
DUAL FILTERS: FILTER 1						
ACR0[7:0]	ACR1[7:4]			ACR1[3:0]	ACR3[3:0]	
AMR0[7:0]	AMR1[7:4]			AMR1[3:0]	AMR3[3:0]	
DUAL FILTERS: FILTER 2						
ACR2[7:0]	ACR3[7:4]					
AMR2[7:0]	AMR3[7:4]					

Table 22-41. Extended Frame Format, Single Filter: Receive Buffer and Filter

DATA (WITH OFFSET)					
0x44	**0x48**	**0x4C**		**0x50**	
ID.28 … ID.21	ID.20 … ID.13	ID.12 … ID.5	ID.4 … ID0	RTR	x x *(not matched)*
SINGLE FILTER					
ACR0[7:0]	ACR1[7:0]	ACR2[7:0]	ACR3[7:2]		(ACR3[1:0] unused)
AMR0[7:0]	AMR1[7:0]	AMR2[7:0]	AMR3[7:2]		(AMR3[1:0] unused)

Table 22-42. Extended Frame Format, Dual Filters: Receive Buffer

DATA (WITH OFFSET)					
0x44	**0x48**	**0x4C**		**0x50**	
ID.28 … ID.21	ID.20 … ID.13	ID.12 … ID.5 *(not matched)*	ID.4 … ID0 *(not matched)*	RTR *(not matched)*	x x *(not matched)*
DUAL FILTERS: FILTER 1					
ACR0[7:0]	ACR1[7:0]				
AMR0[7:0]	AMR1[7:0]				
DUAL FILTERS: FILTER 2					
ACR2[7:0]	ACR3[7:0]				
AMR2[7:0]	AMR3[7:0]				

Chapter 23
Analog-to-Digital Converter/ Brownout Detector

The Analog to Digital Converter (ADC) is an AMBA-compliant SoC peripheral that connects as a slave to the APB. The ADC block consists of an 8-channel, 10-bit Analog-to-Digital Converter with integrated Touch Screen Controller. The complete touch screen interface is achieved by combining the front-end biasing, control circuitry with analog-to-digital conversion, reference generation, and digital control.

The ADC has a bias-and-control network that allows correct operation with both 4- and 5-wire touch panels. A 16-entry × 16-bit wide FIFO holds a 10-bit ADC output and a 4-bit tag number. When the screen is touched, it pushes the conductive coating on the coversheet against the coating on the glass, making electrical contact. The voltages produced are the analog representation of the position touched. The voltage level of the coversheet is converted continuously by the ADC and monitored by the system.

Figure 23-1 shows a block diagram of the ADC.

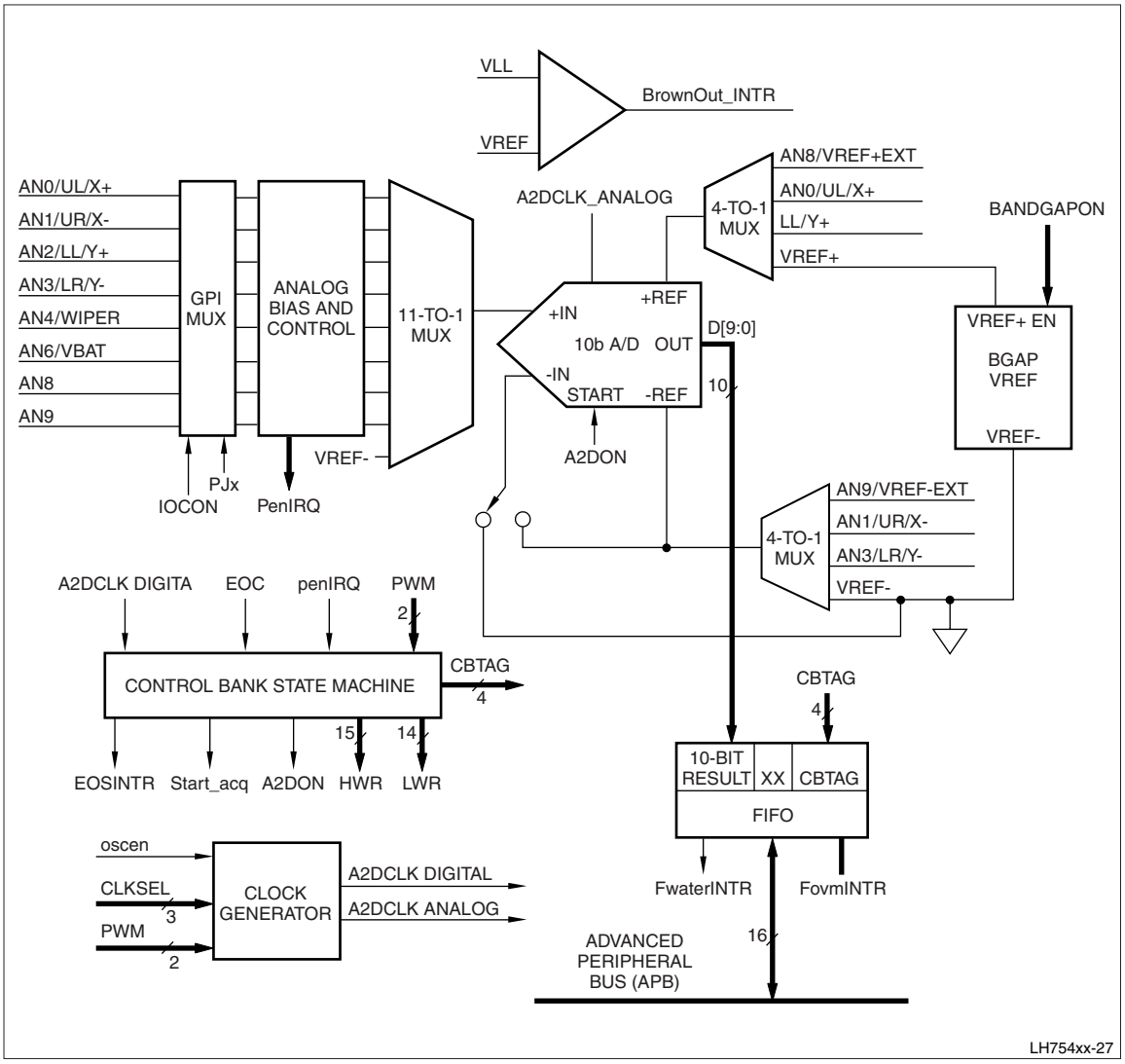


Figure 23-1. ADC Block Diagram

23.1 ADC Features

The ADC has the following features:

- A 10-bit fully differential Successive Approximation Register (SAR) with integrated sample/hold (see Section 23.1.4 for an explanation of the SAR architecture)
- An 8-channel multiplexer that routes user-selected inputs to the ADC in single-ended and differential modes
- A 16-entry × 16-bit wide FIFO that holds the 10-bit ADC output
- Front bias-and-control network for touch screen interface and support functions, which are compatible with industry-standard 4- and 5-wire touch-sensitive panels
- Touch-pressure sensing circuits
- Pen-down sensing circuit and interrupt generator
- Independently controlled voltage reference generator
- Conversion automation function to minimize controller interrupt overhead
- Brownout Detector.

23.1.1 Bias-and-Control Network

The bias-and-control network supports both 4- and 5-wire touch panels. Multiplexers on the reference inputs enable connection in both single-ended and differential modes.

- In 4-wire operation, connection is to inputs X+, X-, Y+, and Y-. Pull HIGH and pull LOW switches allow X and Y coordinate measurement in addition to pen-pressure sensing. The Pen Interrupt line is also available via the Interrupt Masking/Enabling Register (see Section 23.3.2.4).
- In 5-wire operation, panel connections are to UL, UR, LL, and LR inputs, and the sense input is connected to WIPER. The Pen Interrupt line is also available in this mode.

NOTE: For pen-triggered interrupts, use the following procedure instead of using the WIPER's Pen Interrupt (PENIRQ) pull-up. Before checking the Pen Down state, use bias-and-control network bit [2] to short the AN0 pin to VDDA_ADC. This discharges the capacitor formed by the Touch Screen and any capacitance added to the AN0 pin. To generate a Pen Down Interrupt, connect AN4 to VSSA_ADC using bit [8] of the bias-and-control network. Then connect the PENIRQ detector to AN0 using bit [12] of the bias-and-control network.

23.1.2 Clock Generator

The ADC has a programmable measurement clock that is derived from the crystal oscillator, (nominally 14.7456 MHz). The clock drives the measurement sequencer and the successive-approximation circuitry. Higher clock frequencies can allow faster measurement throughput. Slower clock frequencies, on the other hand, can allow fewer clocks to settle for a measurement and can reduce ADC power consumption. If the clock is too slow, the sample-and-hold amplifier on the ADC input may drop before the measurement is complete.

See Section 23.3.2.5 for clock-gating options and for information about programming the available clock frequencies. See Chapter 24 for the maximum ADC clock frequency and sample-and-hold amplifier time constant.

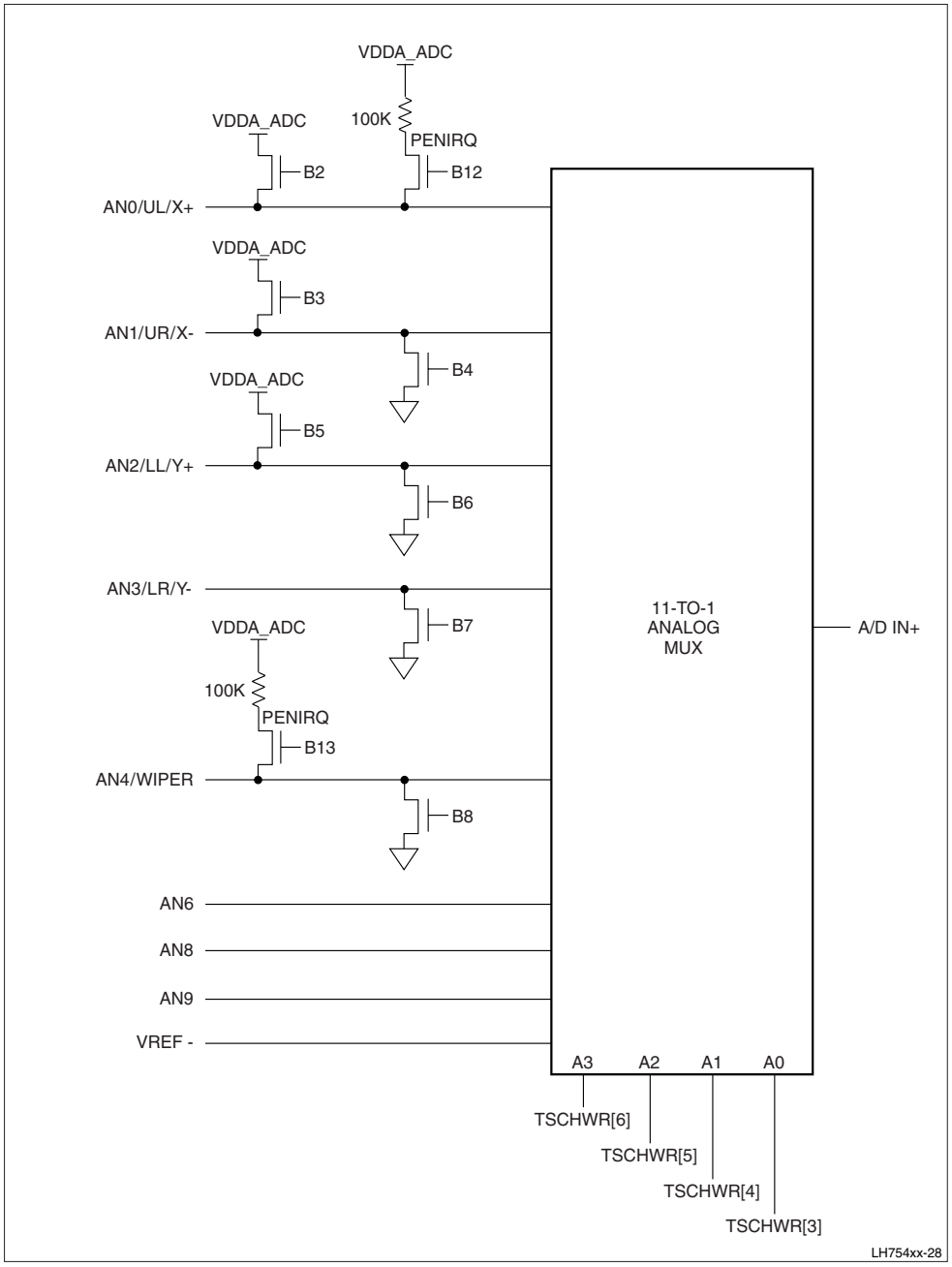


Figure 23-2. Bias-and-Control Network Block Diagram

23.1.3 Brownout Detector

The Brownout Detector is an asynchronous comparator that compares a divided version of the 3.3 V supply and a bandgap-derived reference voltage. If the supply dips below a Trip point, the Brownout Detector sets a bit in the IS Register. The status bit is wired to the VIC and can interrupt the processor core. This allows the Host Controller to warn users of an impending shutdown and may provide the ADC with time to save its state. For Brownout Detector trip point and hysteresis levels, see Chapter 24.

NOTE: The Brownout Detector indicates a brownout condition on startup until the VDDA_ADC pin rises above the trip point.

23.1.4 SAR Architecture

While there are various SAR implementations, the basic architecture is simple. Figure 23-3 shows this architecture.

The analog input voltage (VIN) is held on a track/hold. The N-bit register is set to midscale (100...0, where the most-significant bit is set to 1) to implement the binary search algorithm. This forces the DAC output (VDAC) to be VREF ÷ 2, where VREF is the reference voltage provided to the ADC. Then a comparison is performed to determine whether VIN is less than or greater than VDAC:

- If VIN is less than VDAC, the comparator output is a logic LOW and the most-significant bit of the N-bit register is cleared to 0.
- If VIN is greater than VDAC, the comparator output is a logic HIGH (or 1) and the most-significant bit of the N-bit register remains set to 1.

The SAR control logic then moves to the next bit down, forces that bit HIGH, and conducts another comparison. The SAR control logic repeats this sequence until it reaches the least-significant bit. When the conversion is complete, the N-bit digital word is available in the register.

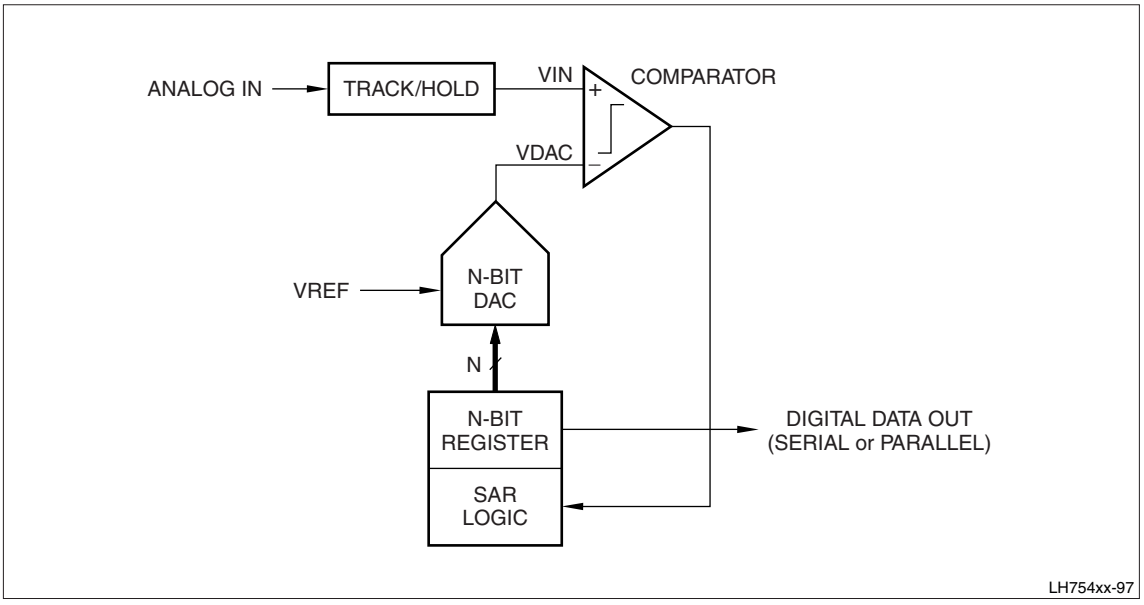


Figure 23-3. Simplified N-bit SAR Architecture

Figure 23-4 shows an example of a 4-bit conversion. In this figure, the y-axis and the bold line show the DAC output voltage. In this example:

1. The first comparison shows that VIN < VDAC. Consequently, bit [3] is set to 0. The DAC is then set to 0100_2 and the second comparison is conducted.
2. In the second comparison, VIN > VDAC, so bit [2] remains at 1. The DAC is then set to 0110_2 and the third comparison is conducted.
3. In the third comparison, bit [1] is set to 0 and the DAC is then set to 0101_2 for the last comparison.
4. In the final comparison, bit [0] remains at 1 because VIN > VDAC.

Four comparison periods are necessary for a 4-bit ADC. Generally, an N-bit SAR ADC requires N comparison periods and will not be ready for the next conversion until the current conversion is completed. This explains why the ADC is power- and space-efficient.

Another feature of SAR ADCs is that power dissipation scales with the sample rate. By comparison, flash or pipelined ADCs usually have constant power dissipation as opposed to sample rate. This SAR ADC feature is especially useful in low-power applications or applications where data acquisition is not continuous.

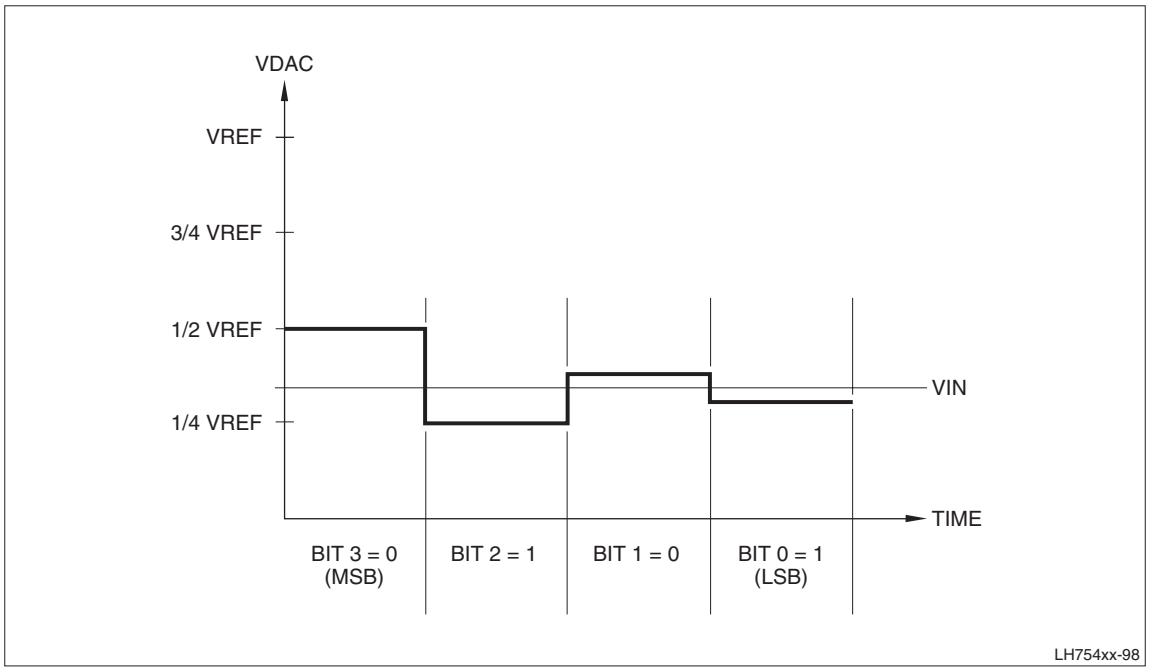


Figure 23-4. Example of a 4-bit SAR ADC Operation

23.2 ADC Theory of Operation

The ADC block can perform a sequence of measurements without intervention from the ARM core. Examples include:

- Determining how the touch-screen biasing switches are configured.
- Ascertaining how much settling time is required before making a measurement.
- Determining which input is muxed into the ADC input and which input is muxed into the ADC reference.

From 1 to 16 different measurements can be performed in a sequence. The number of steps in the sequence is stored in the PC Register.

The biasing switch configuration, settling time, and ADC mux settings for each of the 1 to 16 measurements in the sequence are stored in an entry in the control bank. The measurement sequence can be triggered by either software or a Pen Down Interrupt.

The control bank state machine fetches each entry from the control bank and stores it in the registers LW and HW for the duration of the measurement. When the measurement is complete, the control bank state machine:

- Stores the ADC result and the control bank instruction number in the measurement FIFO.
- Obtains the next configuration from the control bank into LW and HW.

When all steps of the sequence are complete, or at a programmed FIFO watermark level, the control bank state machine can signal the ARM core to read results from the FIFO.

From the FIFO, a program can read each measurement result and corresponding input configuration, as represented by the control bank instruction number. If the FIFO is full:

- The control bank state machine continues to take measurements
- The state machine triggers the FIFO overrun condition.

The ADC can be programmed to repeat the measurement sequence indefinitely, or to pause at the end of a sequence and wait for a new Pen Down Interrupt or software trigger. If the sequence does not repeat continuously, the IHW and ILWCTRL Register values determine the bias and ADC mux settings until a new measurement sequence is triggered.

23.3 ADC Programmer's Model

The base address for the ADC is:

ADC Base Address: 0xFFFC3000

23.3.1 ADC Registers Summary

Table 23-1. Summary of ADC Registers

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
HW	0x00	R	0x0000	High Word Register
LW	0x04	R	0x0000	Low Word Register
RR	0x08	R	0x0000	Results Register
IM	0x0C	RW	0x0000	Interrupt Masking Register
PC	0x10	RW	0x0000	Power Configuration Register
GC	0x14	RW	0x0000	General Configuration Register
GS	0x18	R	0x0210	General Status Register
IS	0x1C	R	0x0010	Interrupt Status Register
FS	0x20	R	0x0004	FIFO Status Register
HWCB0 - HWCB15	0x24 - 0x60	RW	0x0000	High Word Control Bank Registers
LWCB0 - LWCB15	0x64 - 0xA0	RW	0x0000	Low Word Control Bank Registers
IHWCTRL	0xA4	RW	0x0000	Idle High Word Registers
ILWCTRL	0xA8	RW	0x0000	Idle Low Word Registers
MIS	0xAC	R	0x0000	Masked Interrupt Status
IC	0xB0	W		Interrupt Clear Register

23.3.2 ADC Register Definitions

23.3.2.1 High Word Register

HW is the High Word Register. This Read Only status register shows the contents of the current conversion's high word in the control bank. There is a one-to-one correspondence between the contents of the control bank high word and the contents of this register for the current conversion in progress.

Table 23-2. HW Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SETTIME									INP				INM	RefP	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0x00															

Table 23-3. HW Register Definitions

BITS	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:7	SETTIME	**Number of Clock Cycles** Number of clock cycles that the ADC allows for the input signal to settle to within required accuracy before beginning conversion. Used with bits [10:8] of the PC Register to set the acquire time in clock cycles (see Section 23.3.2.5). For example, Frequency In (fIN) 2 MHz (500 ns period) PC[10:8] = 010 (i.e., divide fIN by 4) HW[15:6] = 000100000 (i.e., 32 cycles) Therefore, acquire time is 500 ns × 4 × 32 = 64 μs
6:3	INP	**In+ Mux** Determines the signal connected to the positive input of the ADC. See Table 23-4.
2	INM	**In- Mux** Determines the signal connected to the negative input of the ADC. 0 = Ref- (output of the Ref- Mux) 1 = GND
1:0	RefP	**Ref+ Mux** Determines the signal connected to the positive reference of the ADC. 00 = VREF+ (positive terminal of the internal bandgap reference) 01 = AN0 (UL/X+) 10 = AN2 (LL/Y+) 11 = AN8

Table 23-4. In + Mux Definition

IN+	BIT6	BIT5	BIT4	BIT3
AN0 (UL/X+)	0	0	0	0
AN1 (UR/X-)	0	0	0	1
AN2 (LL/Y+)	0	0	1	0
AN3 (LR/Y-)	0	0	1	1
AN4 (Wiper)	0	1	0	0
RESERVED	0	1	0	1
AN6	0	1	1	0
RESERVED	0	1	1	1
AN8	1	0	0	0
AN9	1	0	0	1
VREF -	1	0	1	0
VREF -	1	0	1	1
VREF -	1	1	0	0
VREF -	1	1	0	1
VREF -	1	1	1	0
VREF -	1	1	1	1

23.3.2.2 Control Bank Low Word Register

LW is the Control Bank Low Word Register. This Read Only status register displays the contents of the current conversion's low word in the control bank. There is a one-to-one correspondence between the contents of the control bank low word and the contents of this register for the current conversion in progress.

Table 23-5. LW Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///		BIASCON												RefM	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0x04															

Table 23-6. LW Register Definitions

BIT	NAME	DESCRIPTION
31:14	///	**Reserved** Write the reset value.
13:2	BIASCON	**Bias Control** These bits drive the FETs, as shown in Figure 23-2. B2 in the figure corresponds to bit [2] in this register. Bits [11:9] must be written as zero.
1:0	RefM	**Ref- Mux** Determines the signal connected to the negative reference of the ADC during Idle Mode. 00 = VREF- (negative terminal of the internal bandgap reference) 01 = AN1 (UR/X-) 10 = AN3 (LR/Y-) 11 = AN9

23.3.2.3 Results Register

RR is the Results Register. This Read Only register is a 16-entry × 16-bit wide FIFO that holds 10-bit ADC output and the 4-bit tag number from the Control Bank State Machine. The read-and-write pointer specifies the FIFO entry to access when a Read or Write is requested. When the FIFO is full, further data writes are temporarily blocked until at least one location is available for the write. Reading from RR removes an entry from the First Out end of the result FIFO.

Table 23-7. RR Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	BITRES										///		CBTAG			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0x08															

Table 23-8. RR Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved** Write the reset value.
15:6	BITRES	**ADC Converter Output** 10-bit digital output of the ADC converter.
5:4	///	**Reserved** Write the reset value.
3:0	CBTAG	**Control Bank Tag** Entry number (HWCTRLBxx or LWCTRLBxx) of the Control bank. The entry number (x) ranges from 0 to 15, corresponding to the conversion associated with the bit result.

23.3.2.4 Interrupt Masking/Enabling Register

IM is the Interrupt Masking/Enabling Register. The active bits used in this register are Read/Write.

This register contains seven bits that enable the interrupts. Software can read the Interrupt Status bits through the IS Register, even if corresponding mask bits are set in this register. Clearing the mask bits clears the pin-level interrupts, but not the interrupt status. The status bits are ANDed with the mask bits to create the pin-level interrupts.

Table 23-9. IM Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									INTEN	///	BOMSK	PMSK	EOSMSK	FWMSK	FOMSK
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC3000 + 0x0C															

Table 23-10. IM Register Definitions

BIT	NAME	DESCRIPTION
31:7	///	**Reserved** Write the reset value.
6	INTEN	**Interrupt Enable** 0 = Disables global interrupt. 1 = Enables global interrupt.
5	///	**Reserved** Write the reset value.
4	BOMSK	**Brown-Out Interrupt Enable** 0 = Disable 1 = Enable
3	PMSK	**Pen Interrupt Enable** 0 = Disable 1 = Enable
2	EOSMSK	**End-of-Sequence Interrupt Enable** 0 = Disable 1 = Enable
1	FWMSK	**FIFO Watermark Interrupt Enable** 0 = Disable 1 = Enable
0	FOMSK	**FIFO Overrun Interrupt Enable** 0 = Disable 1 = Enable

23.3.2.5 Power Configuration Register

PC is the Power Configuration Register. The active bits used in this register are Read/Write.

In this register, the clock divider bits are programmed to set the divider of the system clock for analog operation. Program bits [3:0] to the number of conversions necessary, depending on the conversion.

NOTE: Allow two A2DCLK cycles between successive write cycles to this register. Otherwise, ADC behavior can become erratic.

Table 23-11. PC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///					CLKSEL			PWM		REFEN	///	NOC			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW
ADDR	0xFFFC3000 + 0x10															

Table 23-12. PC Register Definitions

BIT	NAME	DESCRIPTION
31:11	///	**Reserved** Write the reset value.
10:8	CLKSEL	**Clock Select** 000 = Clock oscillator (nominally 14.7456 MHz) 001 = Clock oscillator/2 010 = Clock oscillator/4 011 = Clock oscillator/8 100 = Clock oscillator/16 101 = Clock oscillator/32 110 = Clock oscillator/64 111 = Clock oscillator/128 If the nominal value is used, the only valid settings are 011, 100, 101, and 110.

Table 23-12. PC Register Definitions (Cont'd)

BIT	NAME	DESCRIPTION
7:6	PWM	**Touch Screen Controller Power Mode** 00 =Turns off Power Mode and clock; sets the BROWNOUT field (bit [9]) of the GS Register, indicating that a brownout is detected, even if VDDA_ADC is at the correct voltage. 01 =Standby (wake on SSB or Pen Interrupt, convert, return); clears the BROWNOUT field (bit [9]) of the GS Register, even if VDDA_ADC is at the correct voltage. 10 =Run (always on); clears the BROWNOUT field (bit [9]) of the GS Register, even if VDDA_ADC is at the correct voltage. 11 =Turns off Power Mode and clock; sets the BROWNOUT field (bit [9]) of the GS Register, indicating that a brownout is detected, even if VDDA_ADC is at the correct voltage. The PWM field also affects the status of the A2DCLK signal, Band Gap, and A2D signal (see Table 23-13).
5	REFEN	**Reference Enable** Enables the internal reference buffer.
4	///	**Reserved** Write the reset value.
3:0	NOC	**Number of Conversions (NOC) in Sequence** Actual number of conversions is NOC + 1. The number of conversions ranges from 1 to 16.

Table 23-13. Touch Screen Controller Power Modes

PWM BIT VALUES	nIDLE	A2DCLK ENABLE	BANDGAPON	A2DON
00	0	0	0	0
00	1	0	0	0
01	0	0	1	0
01	1	1	1	1
10	0	0	1	1
10	1	1	1	1
11	0	0	0	0
11	1	0	0	0

NOTE: nIDLE refers to whether the state machine is in the Idle state:
0 = Control Bank State Machine is in the Idle state.
1 = Control Bank State Machine is in another state besides the Idle state.|

A2DCLK ENABLE refers to whether the A2DCLK signal is enabled:
0 = Disables the A2DCLK to the analog circuitry. (The clock is always enabled to the digital circuitry.)
1 = Enables the A2DCLK to the analog circuitry.

BANDGAPON refers to whether Band Gap is turned on (Band Gap is required for the Brownout Detector):
0 = Turns off the Band Gap, disabling the Brownout Detector.
1 = Turns on the Band Gap. This setting is required for the Brownout Detector to work.

A2DON refers to whether the analog circuitry is enabled for the ADC:
0 = Disables the analog circuitry for the ADC.
1 = Enables the analog circuitry for the ADC.

23.3.2.6 General Configuration Register

GC is the General Configuration Register. The active bits used in this register are Read/Write.

In this register, the SSM signal triggers the state machine to retrieve the data from the control bank and store it in the appropriate registers for the ADC. If the SSM bit is set to 11 at the end of a sequence, the state machine continues to convert data.

If the SSM bits are set to 10 and a value of 0b0000110 is written to the GC Register, the EOSINTR_UM bit (bit [2]) of the Interrupt Status Register may never get set. This is normal operation. To accommodate this, wait two A2DCLK periods after setting the SSM bit to 10 before setting the SSB bit.

NOTE: Allow two A2DCLK cycles between successive write cycles to this register. Otherwise, ADC behavior can become erratic.

Table 23-14. GC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///									FIFOWMK				SSB	SSM	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC3000 + 0x14															

Table 23-15. GC Register Definitions

BIT	NAME	DESCRIPTION
31:7	///	**Reserved** Write the reset value.
6:3	FIFOWMK	**FIFO Watermark** Programmed to values between 0 and 15. This value corresponds to watermark levels between 1 and 16, respectively. When the FIFO fills to this level, the FIFO generates an interrupt.
2	SSB	**Start Sequence Bit** 0 = SSB will not start conversion sequence. 1 = SSB will start conversion sequence.
1:0	SSM	**Sequence Start Mode** 00 = SSB or Pen Interrupt starts new conversions. 01 = Pen Interrupt starts new conversions. 10 = SSB starts new conversions. 11 = Continuous conversions. To trigger continuous conversions, set these bits to '11', wait one A2DCLK period, and set the SSB bit to '1'. Thereafter, once any conversions occur and SSM is set to '00' to stop the conversions, conversions can be started again by setting SSM to '11', without having to set SSB. Note that the Pen Interrupt can only be used when the ADC is configured to start on Pen Down. For more information, see Section 23.3.2.7.

23.3.2.7 Sequence Start Mode Issues

Setting the SSM bits to 10 does not generate a Pen IRQ. However, when the Idle Mode is configured to generate a Pen IRQ for a 4-wire Touch Screen, the Pen IRQ detect signal goes LOW on Pen Down. To address this issue, configure the Idle Mode for a Pen IRQ using the Pen-measurement sequence:

1. Idle Step: Configure for Pen Down detect.
2. Connect the AN0 pin to AVDD_ADC to discharge any charge stored in debounce capacitors.
3. Configure for pen down detect.
4. Measure X.
5. Measure Y.
6. Connect the AN0 pin to AVDD_ADC to discharge any charge stored in debounce capacitors.
7. Configure for Pen Down detect.
    a. Set the NOC bits (bits [3:0]) of the PC Register to 1 (two measurements) and set the SSM bits of this register for pen-triggered measurements.
    b. Set up a Pen Interrupt handler that changes the SSM field to SSB-triggered measurements. This prevents Pen Down from retriggering additional measurements.
    c. Set up an End-of-Sequence Interrupt handler that reads out and discards the results from step 2, then reads the results from step 3 to see whether the voltage read indicates that the pen is still down.

If the pen is not still down, set the SSM field back to pen-triggered measurements. Otherwise, start a Pen Down debounce timer, change the PCR NOC field to 5 (all six measurement steps), and change the End-of-Sequence Interrupt handler to be one that handles all six measurements.

23.3.2.8 General Status Register

GS is the General Status Register. In this Read Only register, the 4-bit signal CBSTATE field shows the current state of the Control Bank state machine. The CBTAG signal contains the control bank entry number of the conversion that is taking place.

Table 23-16. GS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///						BRONOUT	PENIRQ	CBSTATE				CBTAG			
RESET	0	0	0	0	0	0	—	0	0	0	0	1	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0x18															

Table 23-17. GS Register Definitions

BIT	NAME	DESCRIPTION
31:10	///	**Reserved** Reads 0.
9	BROWNOUT	**Brown-Out Raw Interrupt Status** 0 = Brown-out Interrupt is not active. 1 = Brown-out Interrupt is active. This bit is undefined at reset.
8	PENIRQ	**Pen IRQ Raw Interrupt Status*** 0 = Pen IRQ Interrupt is not active. 1 = Pen IRQ Interrupt is active.
7:4	CBSTATE	**Control Bank State Machine Status** 0001 = Waiting for sequence start trigger. 0010, 0100, 1000 = Conversion sequence is in progress.
3:0	CBTAG	**Current Conversion Tag Number** Specifies the current conversion tag number.

NOTE: *If the Idle state is configured to bias a 4-wire Touch Screen for Pen IRQ detect, and if the pen is held down, bit [8] is set if the state machine is in the GET_DATA state (Status Register contains 0x120). To resolve this situation set up the measurement sequence for coordinate measurements. Have the first measurement in the sequence and the last measurement in the sequence measure the analog voltage the Pen Detect pin. Then have a software Schmidt Trigger verify the logic level at the beginning and end of the coordinate-measurement sequence. Install a Pen IRQ handler function that changes the measurement mode to software triggered and disables Pen IRQ interrupts. Then install a timer handler that software-triggers a sequence when the timer expires and then stops the timer. Next, install an end-of-sequence interrupt handler that reads the measurement results and determines whether the pen is still down. If the pen is down, the handler starts the timer for triggering the next measurement. The handler discards the first set of measurements taken during the initial Pen Down detection. Otherwise, the handler posts the current pen position to some sort of OS queue. Enable Pen Triggered Measurements to start the system.

23.3.2.9 Interrupt Status Register

IS is the Interrupt Status Register. This Read Only register provides the unmasked value of each interrupt.

Table 23-18. IS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											BROWNOUTINTR_UM	PENSYNC_UM	EOSINTR_UM	FWATERINTR_UM	FOVRNINTR_UM
RESET	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0x1C															

Table 23-19. IS Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Read 0.
4	BROWNOUTINTR_UM	**Brown-Out Interrupt Status** 0 = Brown-out Interrupt is not active. 1 = Brown-out Interrupt is active.
3	PENSYNC_UM	**Pen Interrupt Status** 0 = Pen Interrupt is not active. 1 = Pen Interrupt is active.
2	EOSINTR_UM	**End-of-Sequence Interrupt Active** 0 = EOCIA Interrupt is not active. 1 = EOCIA Interrupt is active.
1	FWATERINTR_UM	**FIFO Watermark Interrupt Active** 0 = FIFO Watermark Interrupt is not active. 1 = FIFO Watermark Interrupt is active.
0	FOVRNINTR_UM	**FIFO Overrun Interrupt Active** 0 = FIFO Overrun Interrupt is not active. 1 = FIFO Overrun Interrupt is active.

23.3.2.10 FIFO Status Register

FS is the FIFO Status Register. This Read Only register indicates the FIFO fill status.

Table 23-20. FS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///				WRPTR				RDPTR				FFF	FEMPTY	FOVRNDET	FGTEWATERMRK
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0x20															

Table 23-21. FS Register Definitions

BIT	NAME	DESCRIPTION
31:12	///	**Reserved** Reads 0.
11:8	WRPTR	**Write Pointer FIFO Location** Current FIFO location where the write pointer is pointing.
7:4	RSPTR	**Read Pointer FIFO Location** Current FIFO location where the read pointer is pointing.
3	FFF	**FIFO Full** 0 = FIFO is not full. 1 = FIFO is full.
2	FEMPTY	**FIFO Empty** 0 = FIFO is not empty. 1 = FIFO is empty.
1	FOVRNDET	**FIFO Overrun Status Bit** Set when the receive logic tries to place data into the FIFO after it has been completely filled. When a new piece of data is received, the FOVRN bit is asserted and the newly received data is discarded. This process repeats for each new piece of data received, until at least one empty FIFO entry exists. When FOVRN is set, an interrupt request is generated. 0 = FIFO has not experienced an overrun. 1 = Logic tried to place data into a full receive FIFO and is requesting an interrupt.
0	FGTEWATERMRK	**FIFO at Watermark** 0 = FIFO has fewer entries than watermark level. 1 = FIFO is at or above watermark level.

23.3.2.11 Control Bank Registers

The Control Bank is a set of 32 16-bit registers. HWCTRLBx and LWCTRLBx are used together and follow the format of the HW and LW Registers (see Section 23.3.2.1 and Section 23.3.2.2).

HWCTRLBx0 contains 16 bits of data for the 4WX conversion. The remaining 14 bits of data for the Control Bank Register conversion is at Tag number LWCBx0. Bits [15] and [14] of the low words are read as zero.

The same logic is used for the Control Bank Registers HWCBx and LWCBx. The High Word Registers should contain the:

- Settling time
- In+ bits
- In- bits
- Ref+ bits.

The Low Word Registers should contain the:

- Bias control settings
- Ref- bits.

For internal access into the control bank, the data writes to the registers from the APB data bus. Each entry is a 16-bit register, with its own address space.

Table 23-22. Sample Entries for Control Bank

TAG TYPE	TAG NAME	CONTENTS			
4WX	HWCBx0	Settling Time[15:7]	In+ [6:3]	In- [2]	Ref+ [1:0]
4WY	HWCBx1	Settling Time[15:7]	In+ [6:3]	In- [2]	Ref+ [1:0]
	HWCBx2 ... HWCBx15				
4WX	LWCBx0	Bias control[13:2]			Ref- [1:0]
4WY	LWCBx1	Bias Control[13:2]			Ref- [1:0]
	LWCBx2 … LWCBx15				

23.3.2.12 Idle High Word Register

IHWCTRL is the high word of the Idle Register. The active bits used in this register are Read/Write.

This register specifies the idle setting time and the inputs connected to the ADC during the Idle state. This register is used with the ILWCTRL Register (see Section 23.3.2.13).

Table 23-23. IHWCTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	SETTIME_ID									INP_ID				INM_ID	REFP_ID	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC3000 + 0xA4															

Table 23-24. IHWCTRL Register Definitions

BIT	NAME	DESCRIPTION
31:16	///	**Reserved** Reads 0.
15:7	SETTIME_ID	**Idle Settling Time** Specifies the delay, in ADC clock cycles, from when the state machine enters the Idle state to when the Pen Interrupt signal can be activated. Prevents spurious trigger of Pen Interrupt while analog signals set up by the IDLE Register are settling.
6:3	INP_ID	**Idle In+ Mux** Specifies the connection to the positive input of the ADC during Idle Mode.
2	INM_ID	**Idle In- Mux** Specifies the connection to the negative input of the ADC during Idle Mode. 0 = Ref- 1 = GND
1:0	REFP_ID	**Idle Ref+ Mux** Specifies the connection to the positive reference of the ADC during Idle Mode. 00 = VREF+ 01 = AN0 (UL/X+) 10 = AN2 (LL/Y+) 11 = AN8

23.3.2.13 Idle Low Word Register

ILWCTRL is the low word of the Idle Register. The active bits used in this register are Read/Write.

This register specifies the inputs connected to the ADC during the Idle state. This register is used with the IHWCTRL Register (see Section 23.3.2.12).

Table 23-25. ILWCTRL Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///		BIASCON_ID												REFM_ID	
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
ADDR	0xFFFC3000 + 0xA8															

Table 23-26. ILWCTRL Register Definitions

BIT	NAME	DESCRIPTION
31:14	///	**Reserved** Write the reset value.
13:2	BIASCON_ID	**Idle Bias Control** Bits [11:9] must be written as zero.
1:0	REFM_ID	**Idle Ref- Mux** Specifies the connection to the negative reference of the ADC during Idle Mode. 00 = VREF- 01 = AN1 (UR/X-) 10 = AN3 (LR/Y-) 11 = AN9

23.3.2.14 Masked Interrupt Status Register

MIS is the Masked Interrupt Status Register. This Read Only register gives the masked value of each interrupt.

Table 23-27. MIS Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///											BROWNOUT	PENSYNC	EOSINTR	FWATERINTR	FOVRNINTR
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ADDR	0xFFFC3000 + 0xAC															

Table 23-28. MIS Register Definitions

BIT	NAME	DESCRIPTION
31:5	///	**Reserved** Reads 0.
4	BROWNOUT	**Brown-Out Interrupt Status** 0 = Brown-out Interrupt is not active. 1 = Brown-out Interrupt is active.
3	PENSYNC	**Pen Interrupt Status** 0 = Pen Interrupt is not active. 1 = Pen Interrupt is active.
2	EOSINTR	**End-of-Sequence Interrupt Active** 0 = EOCIA Interrupt is not active. 1 = EOCIA Interrupt is active.
1	FWATERINTR	**FIFO Watermark Interrupt Active** 0 = FIFO Watermark Interrupt is not active. 1 = FIFO Watermark Interrupt is active.
0	FOVRNINTR	**FIFO Overrun Interrupt Active** 0 = FIFO Overrun Interrupt is not active. 1 = FIFO Overrun Interrupt is active.

23.3.2.15 Interrupt Clear Register

IC is the Interrupt Clear Register. Bits [2:0] of this correspond to the three latched interrupts:

- Writing a 1 to a bit clears the corresponding interrupt.
- Writing a 0 to a bit has no effect.

This register is self-clearing; therefore, the Clear signal lasts one cycle of the system clock.

Table 23-29. IC Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIELD	///															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///													BOIC	PENIC	EOSINTC
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RW	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
ADDR	0xFFFC3000 + 0xB0															

NOTE: The reset value of this register's bits is indeterminate.

Table 23-30. IC Register Definitions

BITS	NAME	DESCRIPTION
31:3	///	**Reserved** Reads undefined. Write 0.
2	BOIC	**Brown-Out Interrupt Clear** 1 = Clear BrownOutINTR.
1	PENIC	**Pen Interrupt Clear** 1 = Clear PenIRQ.
0	EOSINTC	**End of Sequence Interrupt Clear** 1 = Clear EOSINTR.

23.3.3 ADC Timing Formulas

The throughput-conversion time consists of 1 cycle of Get Data state added to 16 cycles of measurement. Starting from the Idle state, the time for a complete measurement sequence, in clock cycles, is calculated as:

$$1CIS + \#MS \times (TCT + \#STC) + 1CEOS$$

where:

- 1CIS is one cycle in Idle state
- #MS is the number of measurements in the sequence
- TCT is the throughput conversion time of 17 cycles
- #STC is the number of settling time cycles per measurement
- 1CEOS is one cycle in the End of Sequence state.

This equals:

- Two cycles, plus
- The number of measurements in the sequence, times
- The throughput conversion time (17 cycles), plus
- The number of settling time cycles per measurement.

Note that in continuous conversion mode, there is no idle cycle, so subtract 1 from this formula.

23.3.4 ADC Interrupts

The ADC has five interrupts:

- Brownout Interrupt (BrownOutINTR)
- Pen Interrupt (PenIRQ)
- End of Sequence Interrupt
- FIFO Watermark Interrupt
- FIFO Overrun Interrupt.

All five interrupts make up the combined interrupt TSCIRQ.

Each of the five individual maskable interrupts is enabled or disabled by changing the mask bits in the IM Register (see Section 23.3.2.4). Software can read the interrupt status bits through the IS Register, even if corresponding mask bits are set (see Section 23.3.2.9). Clearing the mask bits does not clear the interrupt status.

23.3.4.1 Brownout Interrupt

The Brownout Interrupt (BrownOutINTR) is asserted when the supply voltage goes below the trip-point voltage. This interrupt is latched and remains HIGH until the BOIC bit of the ICR register is asserted.

This interrupt is the status of the latched value of the Brownout Interrupt stored in the Interrupt Status Registers. Raw status of the Brownout is stored in the GS Register (see Section 23.3.2.8).

The Brownout Interrupt has its own dedicated output.

NOTE: The latency between clearing the latched Brownout Interrupt and the time when that bit can be set again is one A2DCLK cycle. Polled systems should use the unlatched Brown-Out Raw Interrupt Status bit (bit [9]) in the GS Register (described in Section 23.3.2.8) instead of the latched interrupt status in the ISR.

23.3.4.2 Pen Interrupt

The Pen Interrupt (PenIRQ) occurs when the settings on the bias switches are switched to the Pen Interrupt Mode configuration. The Pen Interrupt is used internally to the Touch Screen Controller as control signal for the state machine. The state machine may begin a sequence of conversions, depending on the contents of the GC Register, when a Pen Interrupt occurs (see Section 23.3.2.6). The PENIRQ is latched and remains HIGH until the PENIC bit of the IC Register is asserted (see Section 23.3.2.15). It is the latched value of the Pen Interrupt that's stored in the Interrupt Status Register. The raw status of the Pen Interrupt is stored in the GS Register (see Section 23.3.2.8).

The Pen Interrupt has its own dedicated output.

NOTES:

1. If a measurement sequence is configured to keep the Touch Screen biased for Pen detect on every measurement, PENIRQ is not generated on every sequence. If, on the other hand, the Pen detect circuit is disconnected, there will be an edge every time the system enters Idle state.
2. For pen-triggered interrupts, use the following procedure instead of using the WIPER's Pen Interrupt (PENIRQ) pull-up. Before checking the Pen Down state, use bias-and-control network bit [2] to short the AN0 pin to VDDA_ADC. This discharges the capacitor formed by the Touch Screen and any capacitance added to the AN0 pin. To generate a Pen Down Interrupt, connect AN4 to VSSA_ADC using bit [8] of the bias-and-control network. Then connect the PENIRQ detector to AN0 using bit [12] of the bias-and-control network.

23.3.4.3 End-of-Sequence Interrupt

The End-of-Sequence Interrupt occurs after the programmed NOCs occurs. After the ADC converts all the data for a given sequence of conversions, this interrupt goes HIGH. The End-of-Sequence Interrupt is latched and remains HIGH until the EOSINTC bit of the IC Register is asserted.

23.3.4.4 FIFO Watermark Interrupt

The FIFO Watermark Interrupt occurs when the number of entries in the FIFO is greater than or equal to the programmed watermark level FIFOWMK (GC Register, bits [6:3]). This interrupt clears when the FIFO level is read from and the FIFO is below the watermark level.

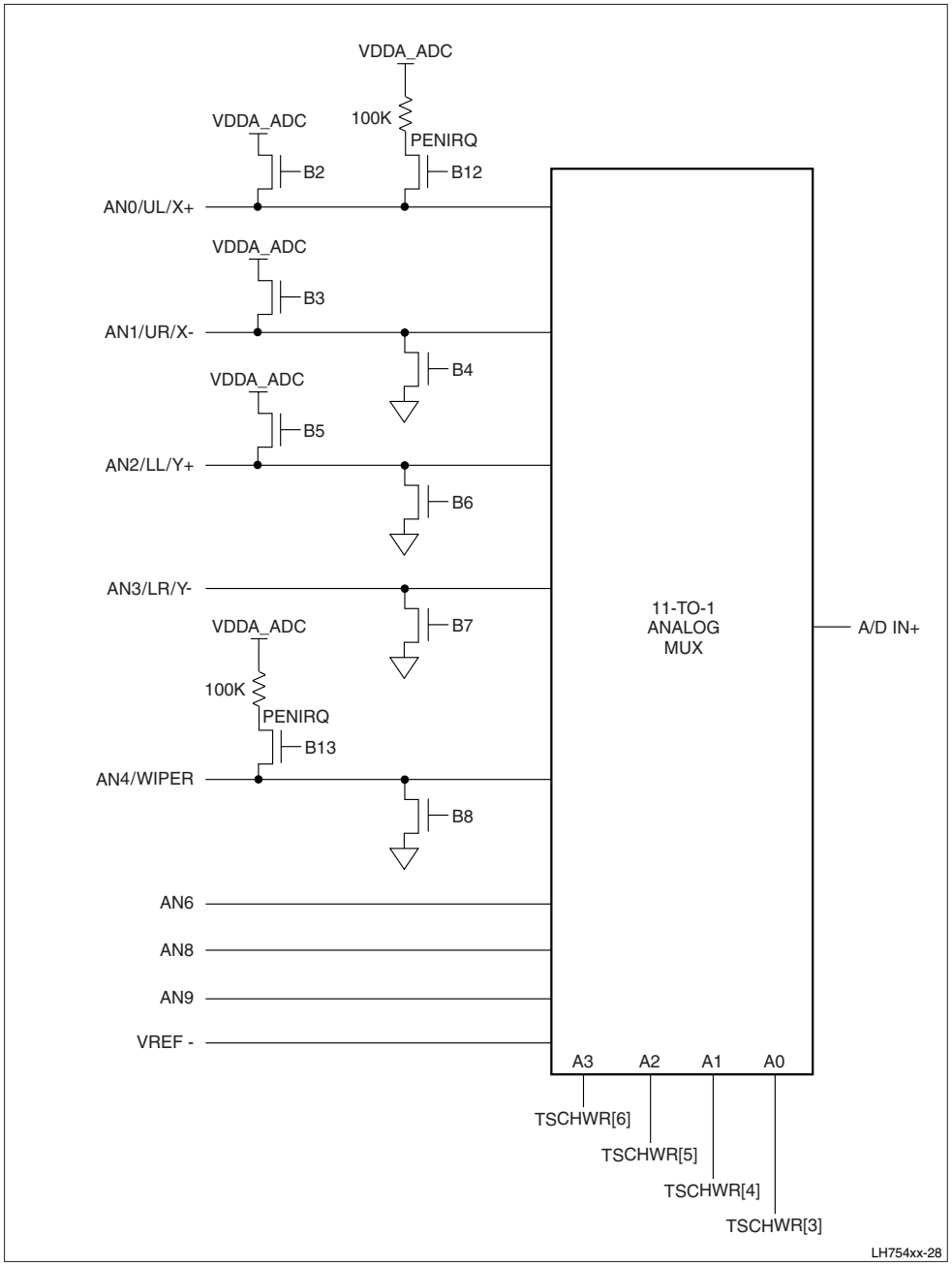


Figure 23-5. Bias-and-Control Network Block Diagram

23.3.4.5 FIFO Overrun Interrupt

The FIFO Overrun Interrupt occurs when the receiving logic tries to place data into the FIFO after the FIFO has been completely filled, exceeding the FIFO's maximum capacity of 16 entries. The interrupt is cleared when the FIFO is read.

Chapter 24
LCD Pin Multiplexing

24.1 LCD Panel Signal Multiplexing Details

The LCDVD bus has three operating modes. These modes correspond to:

- 4-bit Mono STN single panel
- 4-bit Mono STN dual panel
- 8-bit Mono STN single panel

Table 24-1 shows which LCDVD pins are used to supply the pixel data to the panel for each of the above modes of operation. Table 24-2 shows all of the muxing possibilities for each LCD Driver pin.

Table 24-1. LCD Panel Signal Multiplexing

EXTERNAL PIN	4-BIT MONO STN		8-BIT MONO STN SINGLE PANEL
	SINGLE PANEL	DUAL PANEL	
LVCVD[11]	Reserved	MLSTN[3]	Reserved
LVCVD[10]	Reserved	MLSTN[2]	Reserved
LVCVD[9]	Reserved	MLSTN[1]	Reserved
LVCVD[8]	Reserved	MLSTN[0]	Reserved
LVCVD[7]	Reserved	Reserved	MUSTN[7]
LVCVD[6]	Reserved	Reserved	MUSTN[6]
LVCVD[5]	Reserved	Reserved	MUSTN[5]
LVCVD[4]	Reserved	Reserved	MUSTN[4]
LVCVD[3]	MUSTN[3]	MUSTN[3]	MUSTN[3]
LVCVD[2]	MUSTN[2]	MUSTN[2]	MUSTN[2]
LVCVD[1]	MUSTN[1]	MUSTN[1]	MUSTN[1]
LVCVD[0]	MUSTN[0]	MUSTN[0]	MUSTN[0]

NOTES:
1. MUSTN = Mono upper panel STN, dual, and/or single panel.
2. MLSTN = Mono lower panel STN, dual.

Table 24-2. LCD External Pin Multiplexing (LH75401 and LH75411)

EXTERNAL PIN	DEFAULT MODE (NO LCD)	4-BIT MONO STN MODE		8-BIT MONO/ COLOR STN MODE	TFT MODE	AD-TFT, HR-TFT MODE
		SINGLE	DUAL			
PG4/LCDVEEEN/ LCDMOD	PG4	LCDVEEEN	LCDVEEEN	LCDVEEEN	LCDVEEEN	LCDMOD
PG3/LCDVDDEN	PG3	LCDVDDEN	LCDVDDEN	LCDVDDEN	LCDVDDEN	LCDVDDEN
PG2/LCDDSPLEN/ LCDREV	PG2	LCDDSPLEN	LCDDSPLEN	LCDDSPLEN	LCDDSPLEN	LCDREV
PG1/LCDCLS	PG1	PG1	PG1	PG1	PG1	LCDCLS
PG0/LCDPS	PG0	PG0	PG0	PG0	PG0	LCDPS
PH7/LCDDCLK	PH7	LCDDCLK	LCDDCLK	LCDDCLK	LCDDCLK	LCDDCLK
PH6/LCDLP/ LCDHRLP	PH6	LCDLP	LCDLP	LCDLP	LCDLP	LCDLP
PH5/LCDFP/ LCDSPS	PH5	LCDFP	LCDFP	LCDFP	LCDFP	LCDFP
PH4/LCDEN/ LCDEN	PH4	LCDEN	LCDEN	LCDEN	LCDEN	LCDEN
PH3/LCDVD11	PH3	PH3	MLSTN3	PH3	LCDVD11	LCDVD11
PH2/LCDVD10	PH2	PH2	MLSTN2	PH2	LCDVD10	LCDVD10
PH1/LCDVD9	PH1	PH1	MLSTN1	PH1	LCDVD9	LCDVD9
PH0/LCDVD8	PH0	PH0	MLSTN0	PH0	LCDVD8	LCDVD8
PI7/LCDVD7	PI7	PI7	PI7	STN7	LCDVD7	LCDVD7
PI6/LCDVD6	PI6	PI6	PI6	STN6	LCDVD6	LCDVD6
PI5/LCDVD5	PI5	PI5	PI5	STN5	LCDVD5	LCDVD5
PI4/LCDVD4	PI4	PI4	PI4	STN4	LCDVD4	LCDVD4
PI3/LCDVD3	PI3	MUSTN3	MUSTN3	STN3	LCDVD3	LCDVD3
PI2/LCDVD2	PI2	MUSTN2	MUSTN2	STN2	LCDVD2	LCDVD2
PI1/LCDVD1	PI1	MUSTN1	MUSTN1	STN1	LCDVD1	LCDVD1
PI0/LCDVD0	PI0	MUSTN0	MUSTN0	STN0	LCDVD0	LCDVD0

Table 24-3. LCD External Pin Multiplexing (LH75400 and LH75410)

EXTERNAL PIN	DEFAULT MODE (NO LCD)	4-BIT MONO STN MODE		8-BIT MONO STN MODE
		SINGLE	DUAL	
PG4/LCDVEEEN	PG4	LCDVEEEN	LCDVEEEN	LCDVEEEN
PG3/LCDVDDEN	PG3	LCDVDDEN	LCDVDDEN	LCDVDDEN
PG2/LCDDSPLEN	PG2	LCDDSPLEN	LCDDSPLEN	LCDDSPLEN
PG1	PG1	PG1	PG1	PG1
PG0	PG0	PG0	PG0	PG0
PH7/LCDDCLK	PH7	LCDDCLK	LCDDCLK	LCDDCLK
PH6/LCDLP	PH6	LCDLP	LCDLP	LCDLP
PH5/LCDFP	PH5	LCDFP	LCDFP	LCDFP
PH4/LCDEN	PH4	LCDEN	LCDEN	LCDEN
PH3/LCDVD11	PH3	PH3	MLSTN3	PH3
PH2/LCDVD10	PH2	PH2	MLSTN2	PH2
PH1/LCDVD9	PH1	PH1	MLSTN1	PH1
PH0/LCDVD8	PH0	PH0	MLSTN0	PH0
PI7/LCDVD7	PI7	PI7	PI7	STN7
PI6/LCDVD6	PI6	PI6	PI6	STN6
PI5/LCDVD5	PI5	PI5	PI5	STN5
PI4/LCDVD4	PI4	PI4	PI4	STN4
PI3/LCDVD3	PI3	MUSTN3	MUSTN3	STN3
PI2/LCDVD2	PI2	MUSTN2	MUSTN2	STN2
PI1/LCDVD1	PI1	MUSTN1	MUSTN1	STN1
PI0/LCDVD0	PI0	MUSTN0	MUSTN0	STN0

Chapter 25
Recommended Layout Practices

25.1 Protecting Against ElectroStatic Discharge

Before handling the LH75400/01/10/11, read these electrostatic discharge (ESD) instructions to prevent damage to the devices.

- Always wear an ESD-preventive grounding wrist strap or observe similar ESD precautions when working with the LH75400/01/10/11 board. Be sure the wrist strap makes good skin contact. Do not remove the wrist strap until you finish working with the LH75400/01/10/11 board.
- Avoid contact between equipment and clothing. The wrist strap only protects the equipment from ESD voltages on the body; ESD voltages on clothing can still cause damage.
- Place the LH75400/01/10/11 board component side up on an antistatic mat. When an LH75400/01/10/11 board is not being used, place it in a static-shielding bag.
- Handle an LH75400/01/10/11 board by the edges only; avoid touching the components, traces, or any connector pins.

25.1.1 Special ESD Considerations

A typical application for the Analog-to-Digital converter in the LH75400/01/10/11 devices is interfacing to a Touch Screen panel. Normally, this application requires filtering on the inputs, and can be subject to ESD from users. It is recommended that ESD design precautions and good board-design practices in general be followed for pins 89 through 96. An example of a filter circuit that provides good ESD protection is shown in Figure 1.

The capacitors in this circuit should be ceramic, with a grade of X7R or better and rated for 50 Volts or higher. Designers should select capacitor values appropriate for their design. For more information about suggested capacitance values, circuit, and layout recommendations, refer to the appropriate sections in this chapter.

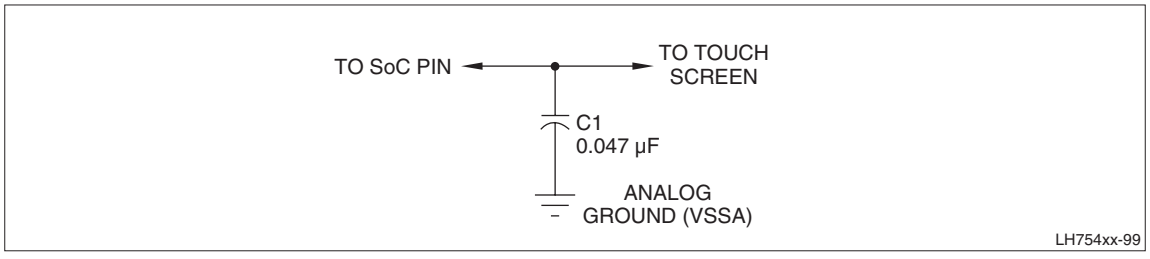


Figure 25-1. ESD Filter Circuit Example

25.2 Printed Circuit Board Layout Practices

25.2.1 Power Supply Decoupling

The SoCs have separate power and ground pins for different internal circuitry sections. The VDD and VSS pins supply power to I/O buffers, while VDDC and VSSC supply power to the core logic.

Each of the VDD and VDDC pins must be provided with a low impedance path to the corresponding board power supply. Likewise, the VSS and VSSC pins must be provided with a low impedance path to the board ground.

Each power supply must be decoupled to ground using at least one 0.1 μF high frequency capacitor located as close as possible to a VDDx, VSSx pin pair on each of the four sides of the chip. If room on the circuit board allows, add one 0.01 μF high frequency capacitor near each VDDx, VSSx pair on the chip.

To be effective, the capacitor leads and associated circuit board traces connecting to the chip VDDx, VSSx pins must be kept to less than half an inch (12.7 mm) per capacitor lead. There must be one bulk 10 μF capacitor for each power supply placed near one side of the chip.

25.2.2 Required VDDA_PLL, VSSA_PLL Filter

The VDDA_PLL pin supplies power to the chip PLL circuitry. VSSA_PLL is the ground return path for the PLL circuit. If the internal PLL circuit will be used, these pins must have a low-pass filter attached as shown in Figure 25-1.

The power pin VDDA_PLL path must be a single wire from the IC package pin to the high frequency capacitor, then to the low frequency capacitor, and finally through the series resistor to the board power supply. The distance from the IC pin to the high frequency capacitor must be kept as short as possible.

Similarly, the VSSA_PLL path is from the IC pin to the high frequency capacitor, then to the low frequency capacitor, keeping the distance from the IC pin to the high frequency cap as short as possible.

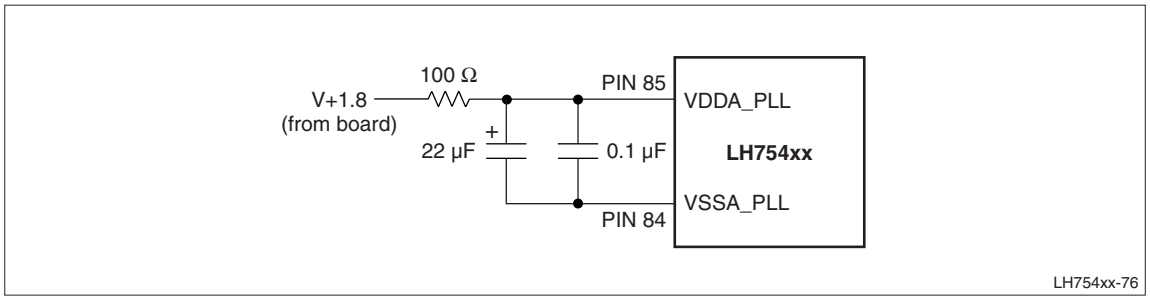


Figure 25-1. VDD_PLL, VSSA_PLL Filter Circuit

CAUTION

Note that the VSSA_PLL pin specifically does not have a connection to the circuit board ground. The PLL circuit has an internal DC ground connection to VSS (GND), so the external VSSA_PLL pin must NOT be connected to the circuit board ground, but only to the filter components.

25.2.3 Unused Input Signal Conditioning

Floating input signals can cause excessive power consumption. Unused inputs which do not include internal pull-up or pull-down resistors should be pulled up or down externally, to tie the signal to its inactive state.

Some GPIO signals may default to inputs. If the pins which carry these signals are unused, software can program these signals as outputs, to eliminate the need for pull-ups or pull-downs. Power consumption may be higher than expected until such software executes.

Some inputs have internal pull-ups or pull-downs. If unused, these inputs do not require external conditioning.

Consider all signals that are Inputs at reset (see Chapter 1, Section 1.8).

25.2.4 Other Circuit Board Layout Practices

All output pins have fast rise and fall times. Printed circuit trace interconnection length must therefore be reduced to minimize overshoot, undershoot and reflections caused by transmission line effects of these fast output switching times. This recommendation particularly applies to the address and data buses.

When considering capacitance, calculations must consider all device loads and capacitances due to the circuit board traces. Capacitance due to the traces will depend upon a number of factors, including the trace width, dielectric material the circuit board is made from and proximity to ground and power planes.

Attention to power supply decoupling and printed circuit board layout becomes more critical in systems with higher capacitive loads. As these capacitive loads increase, transient currents in the power supply and ground return paths also increase.

Chapter 26
Register Map

26.1 SMC Registers

Base address: 0x40000000 (also 0x00000000 if REMAP is '00')

Table 26-1. SMC Register Summary

NAME	ADDRESS OFFSET	TYPE	WIDTH	RESET VALUE	DESCRIPTION
BCR0	SMC RegBase + 0x00	RW	32	0x1000FFEF (16-bit) or 0x0000FBEF (8-bit)	Configuration Register for Memory Bank 0
BCR1	SMC RegBase + 0x04	RW	32	0x1000FFEF	Configuration Register for Memory Bank 1
BCR2	SMC RegBase + 0x08	RW	32	0x1000FFEF	Configuration Register for Memory Bank 2
BCR3	SMC RegBase + 0x0C	RW	32	0x1000FFEF	Configuration Register for Memory Bank 3

NOTE: The reset value of the first SMC base register depends on bus width. If PD2/INT2 is pulled HIGH on Reset, Bank0 defaults to a 16-bit memory width. If PD2/INT2 is pulled LOW on Reset, Bank0 defaults to an 8-bit memory width.

26.2 RCPC Registers

Base address: 0xFFFE2000

Table 26-2. RCPC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Ctrl	0x00	RW	0x063	RCPC Control Register
Identification	0x04	R	0x5400	ID Register
Remap	0x08	RW	0x0	Remap Control Register
SoftReset	0x0C	RW	0x0000	Soft Reset Register
ResetStatus	0x10	R	0x1	Reset Status Register
ResetStatusClr	0x14	W		Reset Status Clear Register
SysClkPrescaler	0x18	RW	0xF	System Clock Prescaler Register
///	0x1C-0x20			Reserved
APBPeriphClkCtrl0	0x24	RW	0x3FF	Peripheral Clock Control 0 Register
APBPeriphClkCtrl1	0x28	RW	0x3	Peripheral Clock Control 1 Register
AhbClkCtrl	0x2C	RW	0x1	AHB Clock Control
///	0x30-0x3C			Reserved
LCDPrescaler	0x40	RW	0x00	LCD Prescaler Register
SSPPrescaler	0x44	RW	0x00	SSP Prescaler Register
///	0x48-0x7C			Reserved
IntConfig	0x80	RW	0x0000	External Interrupt Configuration Register
IntClear	0x84	W		External Interrupt Clear Register
///	0x88			Reserved

26.3 VIC Registers

Base address: 0xFFFFF000

Table 26-3. VIC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
IRQStatus	0x000	R	0x00000000	IRQ Status Register
FIQStatus	0x004	R	0x00000000	FIQ Status Register
RawIntr	0x008	R		Raw Interrupt Status Register
IntSelect	0x00C	RW	0x00000000	Interrupt Select Register
IntEnable	0x010	RW	0x00000000	Interrupt Enable Register
IntEnClear	0x014	W		Interrupt Enable Clear Register
SoftInt	0x018	RW	0x00000000	Software Interrupt Register
SoftInt Clear	0x01C	W		Software Interrupt Clear Register
///	0x020			Reserved
VectAddr	0x030	RW	0x00000000	Vector Address Register
DefVectAddr	0x034	RW	0x00000000	Default Vector Address Register
VectAddr 0	0x100	RW	0x00000000	Vector Address 0 Register
VectAddr 1	0x104	RW	0x00000000	Vector Address 1 Register
VectAddr 2	0x108	RW	0x00000000	Vector Address 2 Register
VectAddr 3	0x10C	RW	0x00000000	Vector Address 3 Register
VectAddr 4	0x110	RW	0x00000000	Vector Address 4 Register
VectAddr 5	0x114	RW	0x00000000	Vector Address 5 Register
VectAdd 6	0x118	RW	0x00000000	Vector Address 6 Register
VectAddr 7	0x11C	RW	0x00000000	Vector Address 7 Register
VectAddr 8	0x120	RW	0x00000000	Vector Address 8 Register
VectAddr 9	0x124	RW	0x00000000	Vector Address 9 Register
VectAddr 10	0x128	RW	0x00000000	Vector Address 10 Register
VectAddr 11	0x12C	RW	0x00000000	Vector Address 11 Register
VectAddr 12	0x130	RW	0x00000000	Vector Address 12 Register
VectAddr 13	0x134	RW	0x00000000	Vector Address 13 Register
VectAddr 14	0x138	RW	0x00000000	Vector Address 14 Register
VectAddr 15	0x13C	RW	0x00000000	Vector Address 15 Register
VectCtrl 0	0x200	RW	0x00	Vector Control 0 Register
VectCtrl 1	0x204	RW	0x00	Vector Control 1 Register
VectCtrl 2	0x208	RW	0x00	Vector Control 2 Register
VectCtrl 3	0x20C	RW	0x00	Vector Control 3 Register
VectCtrl 4	0x210	RW	0x00	Vector Control 4 Register
VectCtrl 5	0x214	RW	0x00	Vector Control 5 Register
VectCtrl 6	0x218	RW	0x00	Vector Control 6 Register
VectCtrl 7	0x21C	RW	0x00	Vector Control 7 Register
VectCtrl 8	0x220	RW	0x00	Vector Control 8 Register
VectCtrl 9	0x224	RW	0x00	Vector Control 9 Register
VectCtrl 10	0x228	RW	0x00	Vector Control 10 Register

Table 26-3. VIC Register Summary (Cont'd)

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
VectCtrl 11	0x22C	RW	0x00	Vector Control 11 Register
VectCtrl 12	0x230	RW	0x00	Vector Control 12 Register
VectCtrl 13	0x234	RW	0x00	Vector Control 13 Register
VectCtrl 14	0x238	RW	0x00	Vector Control 14 Register
VectCtrl 15	0x23C	RW	0x00	Vector Control 15 Register
///	0x300			Reserved
///	0x304			Reserved
///	0x308			Reserved
///	0x30C	R	0x0	Reserved
///	0x310			Reserved

26.4 IOCON Registers

Base address: 0xFFFE5000

Table 26-4. IOCON Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
EBI_MUX	0x00	RW	See Note	EBI Interface Muxing Register
PD_MUX	0x04	RW	0x0000	Pins PD6/INT6 to PD0/INT0 Muxing Register
PE_MUX	0x08	RW	0x0000	Pins PE7/SSPRM to PE0/UARTRX2 Muxing Register
TIMER_MUX	0x0C	RW	0x0000	Timer Muxing Register
LCD_MUX	0x10	RW	0x0000	LCD Mode Muxing Register
PA_RES_MUX	0x14	RW	0xAAAA	Pins PA7/D15 to PA0/D8 Resistor Muxing Register
PB_RES_MUX	0x18	RW	0x0555	Pins PB5/nWAIT to PB0/nCS1 Resistor Muxing Register
PC_RES_MUX	0x1C	RW	0x0000	Pins PC7/A23 to PC0/A16 Resistor Muxing Register
PD_RES_MUX	0x20	RW	0x095A	Pins PD6/INT6 to PD0/INT0 Resistor Muxing Register
PE_RES_MUX	0x24	RW	0x4455	Pins PE7/SSPRM to PE0/UARTRX2 Resistor Muxing Register
ADC_MUX	0x28	RW	0x0000	Pins AN3/PJ7 to AN0/PJ0 Muxing Register

NOTE: The reset value of the EBI Interface Muxing Register is based on whether the system is booted in 16-bit or 8-bit Mode. In 16-bit Mode, the reset value is 0x4000. During 8-bit Mode, the reset value is 0x0000.

26.5 DMA Controller Registers

Base address: 0xFFFE1000

Table 26-5. DMA Controller Register Summary

NAME	ADRESS OFFSET	TYPE	DESCRIPTION
Stream0	0x000	RW	Data Stream0 Register Base
Stream1	0x040	RW	Data Stream1 Register Base
Stream2	0x080	RW	Data Stream2 Register Base
Stream3	0x0C0	RW	Data Stream3 Register Base
Mask	0x0F0	RW	DMA Interrupt Mask Register
Clr	0x0F4	W	DMA Interrupt Clear
Status	0x0F8	R	DMA Status Register

26.6 DMA Stream Registers

Base address: 0xFFFE1000

Table 26-6. DMA Stream Register Summary

NAME	ADDRESS OFFSET	TYPE	DESCRIPTION
SourceLo	0x000	RW	Source base address, lower 16 bits
SourceHi	0x004	RW	Source base address, higher 16 bits
DestLo	0x008	RW	Destination base address, lower 16 bits
DestHi	0x00C	RW	Destination base address, higher 16 bits
Max	0x010	RW	Maximum Count Register
Ctrl	0x014	RW	Control Register
SoCurrHi	0x018	R	Current source address, higher 16 bits
SoCurrLo	0x01C	R	Current source address, lower 16 bits
DeCurrHi	0x020	R	Current destination address, lower 16 bits
DeCurrLo	0x024	R	Current destination address, higher 16 bits
TCnt	0x028	R	Terminal counter
///	0x2C - 0x3C		Reserved

26.7 CLCDC Registers

Base address: 0xFFFF4000

Table 26-7. CLCDC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Timing0	0x000	RW	0x00000000	Horizontal Timing Panel Control Register
Timing1	0x004	RW	0x00000000	Vertical Timing Panel Control Register
Timing2	0x008	RW	0x0000000	Clock and Signal Polarity Control Register
///	0x00C	RW		Reserved
UPBASE	0x010	RW	0x0000000	Upper Panel Frame Buffer Base Address Register
LPBASE	0x014	RW	0x00000000	Lower Panel Frame Buffer Base Address Register
INTRENABLE	0x018	RW	0x00000000	Interrupt Enable Register
Ctrl	0x01C	RW	0x0000	LCD Panel Parameters, LCD Panel Power, and CLCDC Control Register
Status	0x020	RW	0x00000000	Raw Interrupt Status Register
Interrupt	0x024	R	0x00000000	Final Masked Interrupts Register
UPCURR	0x028	R	0x00000000	Upper Panel Frame Buffer Current Address Register
LPCURR	0x02C	R	0x00000000	Lower Panel Frame Buffer Current Address Register
///	0x030 - 0x1FC		0x00000	Reserved
Palette	0x200 - 0x3FC	RW		256 × 16-bit Color Palette Register. Palette is addressed at 32 bits.
///	0x400 - 0x7FF			Reserved

NOTE: These registers pertain to the LH75401 and LH75411 SoC devices only.

26.8 ALI Registers

Base address: 0xFFFE4000

Table 26-8. ALI Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Setup	0x000	RW	0x000C	LCD Interface Peripheral Setup Register
CTRL	0x004	RW	0x0000	LCD Interface Peripheral Control Register
Timing1	0x008	RW	0x1000	LCD Interface Peripheral Timing Register 1
Timing2	0x00C	RW	0x0000	LCD Interface Peripheral Timing Register 2
///	0x010 - 0xFFF			Reserved

NOTE: These registers pertain to the LH75401 and LH75411 SoC devices only.

26.9 LCDC Registers

Base Address: 0xFFFF4000

Table 26-9. LCDC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
Timing0	0x000	RW	0x00000000	Horizontal Timing Panel Control Register
Timing1	0x004	RW	0x00000000	Vertical Timing Panel Control Register
Timing2	0x008	RW	0x0000000	Clock and Signal Polarity Control Register
///	0x00C	RW		Reserved
UPBASE	0x010	RW	0x0000000	Upper Panel Frame Buffer Base Address Register
LPBASE	0x014	RW	0x00000000	Lower Panel Frame Buffer Base Address Register
INTRENABLE	0x018	RW	0x00000000	Interrupt Enable Register
CTRL	0x01C	RW	0x0000	LCD Panel Parameters, LCD Panel Power, and LCDC Control Register
Status	0x020	RW	0x00000000	Raw Interrupt Status Register
Interrupt	0x024	R	0x00000000	Final Masked Interrupts Register
UPCURR	0x028	R	0x00000000	Upper Panel Frame Buffer Current Address Register
LPCURR	0x02C	R	0x00000000	Lower Panel Frame Buffer Current Address Register
///	0x030 - 0x1FC		0x00000	Reserved
Palette	0x200 - 0x3FC	RW		LCD Palette Register. Palette is addressed at 32 bits.
///	0x400 - 0x7FF			Reserved

NOTE: These registers pertain to the LH75400 and LH75410 SoC devices only.

26.10 Timer Registers

Base address: 0xFFFC4000

Table 26-10. Timer 0 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x00	RW	0x0000	Timer 0 Control Register
CMP_CAP_CTRL	0x04	RW	0x0000	Timer 0 Compare/Capture Control Register
INT_CTRL	0x08	RW	0x0000	Timer 0 Interrupt Control Register
STATUS	0x0C	RW	0x0000	Timer 0 Status Register
CNT	0x10	RW	0x0000	Timer 0 Counter Register
CMP0	0x14	RW	0xFFFF	Timer 0 Compare Register 0
CMP1	0x18	RW	0xFFFF	Timer 0 Compare Register 1
CAP0	0x1C	R	0x0000	Timer 0 Capture Register 0
CAP1	0x20	R	0x0000	Timer 0 Capture Register 1
CAP2	0x24	R	0x0000	Timer 0 Capture Register 2
CAP3	0x28	R	0x0000	Timer 0 Capture Register 3
CAP4	0x2C	R	0x0000	Timer 0 Capture Register 4

Table 26-11. Timer 1 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x30	RW	0x0000	Timer 1 Control Register
INT_CTRL	0x34	RW	0x0000	Timer 1 Interrupt Control Register
STATUS	0x38	RW	0x0000	Timer 1 Status Register
CNT	0x3C	RW	0x0000	Timer 1 Counter Register
CMP0	0x40	RW	0xFFFF	Timer 1 Compare Register 0
CMP1	0x44	RW	0xFFFF	Timer 1 Compare Register 1
CAP0	0x48	R	0x0000	Timer 1 Capture Register 0
CAP1	0x4C	R	0x0000	Timer 1 Capture Register 1

Table 26-12. Timer 1 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x50	RW	0x0000	Timer 2 Control Register
INT_CTRL	0x54	RW	0x0000	Timer 2 Interrupt Control Register
STATUS	0x58	RW	0x0000	Timer 2 Status Register
CNT	0x5C	RW	0x0000	Timer 2 Counter Register
CMP0	0x60	RW	0xFFFF	Timer 2 Compare Register 0
CMP1	0x64	RW	0xFFFF	Timer 2 Compare Register 1
CAP0	0x68	R	0x0000	Timer 2 Capture Register 0
CAP1	0x6C	R	0x0000	Timer 2 Capture Register 1

26.11 WDT Registers

Base Address: 0xFFFE3000

Table 26-13. WDT Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL	0x00	RW	0x00	Watchdog Control Register
CNTR	0x04	W		Watchdog Counter Reset
TSTR	0x08	RW	0x40	Watchdog Register
CNT0	0x0C	R	0x00	WDT Counter Section 0
CNT1	0x10	R	0x00	WDT Counter Section 1
CNT2	0x14	R	0x01	WDT Counter Section 2
CNT3	0x18	R	0x00	WDT Counter Section 3

26.12 RTC Registers

Base Address: 0xFFFE0000

Table 26-14. RTC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
DR0	0x00	R		Lower 16-bit Data Register
DR1	0x04	R		Upper 16-bit Data Register
MR0	0x08	RW		Lower 16-bit Match Register
MR1	0x0C	RW		Upper 16-bit Match Register
STAT/EOI	0x10	RW		Interrupt Status Register (read)/ Interrupt Clear Register (write)
LR0	0x14	RW		Lower 16-bit Counter Load Register
LR1	0x18	RW		Upper 16-bit Counter Load Register
CTRL	0x1C	RW		Control Register

26.13 SSP Registers

Base address: 0xFFFC6000

Table 26-15. SSP Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
CTRL0	0x000	RW	0x0000	Control Register 0
CTRL1	0x004	RW	0x0000	Control Register 1
DR	0x008	RW	0x0000	Receive FIFO (Read)/Transmit FIFO (Write)
SR	0x00C	R	0x03	Status Register
CPSR	0x010	RW	0x00	Clock Prescale Register
IIR/ICR	0x014	RW	0x0	Interrupt Identification Register (read)/Interrupt Clear Register (write)
RXTO	0x018	RW	0x000	Receive Timeout Register
///	0x01C - 0xFFF			Reserved

NOTE: All other address locations are reserved and must not be used during normal operation.

26.14 UART0 and UART1 Registers

UART0 Base Address: 0xFFFC0000
UART1 Base Address: 0xFFFC1000

Table 26-16. UART0 and UART1 Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
DR	0x000	RW	0x---	Data read or written from the interface. It is 12 bits wide on a read and 8 on a write.
RSR/ECR	0x004	RW	0x0	Receive Status Register (read)/Error Clear Register (write)
///	0x008 - 0x014			Reserved
FR	0x018	R	0x00000090	Flag Register (read only)
///	0x01C - 0x020			Reserved
IBRD	0x024	R	0x0000	Integer Baud Rate Divisor Register
FBRD	0x028	RW	0x00	Fractional Baud Rate Divisor Register
LCTRL_H	0x02C	RW	0x00	Line Control Register, HIGH byte
CTRL	0x030	RW	0x0300	UART Control Register
IFLS	0x034	RW	0x12	Interrupt FIFO Level Select Register
IMSC	0x038	RW	0x000	Interrupt Mask Set/Clear Register
RIS	0x03C	R	0x00-	Raw Interrupt Status Register
MIS	0x040	R	0x00-	Masked Interrupt Status Register
ICR	0x044	W		Interrupt Clear Register
DMACTRL	0x048	RW	0x00	DMA Control Register
///	0x04C - 0x07C			Reserved
///	0x080 - 0x08C			Reserved
///	0x090 - 0xFFC			Reserved

26.15 UART2 Registers

Base address: 0xFFFC2000

Table 26-17. UART2 Register Summary (Register Bank 0)

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
TXD	0x00	0	W		Transmit Buffered Data Register
RXD	0x00	0	R	0x00	Receive Buffered Data Register
BAL	0x00	1	RW	0x02	BRGA Divisor Least Significant Byte Register. The DLAB bit in the LCR Register must be set to access this register.
BAH	0x04	1	RW	0x00	BRGA Divisor Most Significant Byte Register. The DLAB bit in the LCR Register must be set to access this register.
GER	0x04	0	RW	0x00	General Enable Register
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register
LCR	0x0C		RW	0x00	Line Control Register
MCTRL	0x10		RW	0x00	Loopback Control Register
LSR	0x14		RW	0x60	Line Status Register
///	0x18				Reserved
ACTRL0	0x1C		RW	0x00	Address/Control Character Register 0

Table 26-18. UART2 Register Summary (Register Bank 1)

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
TXD	0x00		W		Transmit Buffered Data Register
RXD	0x00		R	0x00	Receive Buffered Data Register
TXF	0x04		W		Transmit Character Flag Register
RXF	0x04		R	0x40	Receive Character Flag Register
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register (same register as in bank 0)
TMCTRL	0x0C		W		Timer Control Register
TMST	0x0C		R	0x30	Timer Status Register
MCTRL	0x10		W		Loopback Control Register
FLR	0x10		R	0x00	FIFO Level Register
RCM	0x14		W		Receive Command Register
RST	0x14		R	0x00	Receive Machine Status Register
TCM	0x18		W		Transmit Command Register
///	0x18		R		Reserved
ICM	0x1C		W		Internal Command Register
GSR	0x1C		R	0x12	General Status Register

Table 26-19. UART2 Register Summary (Register Bank 2)

NAME	ADDRESS OFFSET	DLAB	TYPE	RESET VALUE	DESCRIPTION
///	0x00				Reserved
FMD	0x04		RW	0x00	FIFO Mode Register
GIR	0x08		RW	0x01	General Interrupt Register/Bank Register (same register as in bank 0)
TMD	0x0C		RW	0x00	Transmit Machine Mode Register
IMD	0x10		RW	0x0C	Internal Mode Register
ACTRL1	0x14		RW	0x00	Address/Control Character Register 1
RIE	0x18		RW	0x1E	Interrupt Enable Register
RMD	0x1C		RW	0x00	Receive Machine Mode Register

26.16 GPIO Registers

Ports A and B Base Address: 0xFFFDF000
Ports C and D Base Address: 0xFFFDE000
Ports C and F Base Address: 0xFFFDD000
Ports G and H Base Address: 0xFFFDC000
Ports I and J Base Address: 0xFFFDB000

Table 26-20. GPIO Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
PIDR	0x00	RW	0x00000000	Port I Data Register
PJDR	0x04	RW	0x00000000	Port J Data Register
PIDDR	0x08	RW	0x00000000	Port I Data Direction Register
	0x0C	Reserved		Reserved
PGDR	0x00	RW	0x00000000	Port G Data Register
PHDR	0x04	RW	0x00000000	Port H Data Register
PGDDR	0x08	RW	0x00000000	Port G Data Direction Register
PHDDR	0x0C	RW	0x00000000	Port H Data Direction Register
PEDR	0x00	RW	0x00000000	Port E Data Register
PFDR	0x04	RW	0x00000000	Port F Data Register
PEDDR	0x08	RW	0x00000000	Port E Data Direction Register
PFDDR	0x0C	RW	0x00000000	Port F Data Direction Register
PCDR	0x00	RW	0x00000000	Port C Data Register
PDDR	0x04	RW	0x00000000	Port D Data Register
PCDDR	0x08	RW	0x00000000	Port C Data Direction Register
PDDDR	0x0C	RW	0x00000000	Port D Data Direction Register
PADR	0x00	RW	0x00000000	Port A Data Register
PBDR	0x04	RW	0x00000000	Port B Data Register
PADDR	0x08	RW	0x00000000	Port A Data Direction Register
PBDDR	0x0C	RW	0x00000000	Port B Data Direction Register

26.17 CANBUS Controller Registers

Base address: 0xFFFC5000

Table 26-21. CANBUS Controller Register Summary

REGISTER	ADDRESS OFFSET	TYPE		DESCRIPTION	RESET VALUE	NOTES
		OPERATING MODE	RESET MODE			
MOD	0x00	RW	RW	Mode Register	0x01	1
CMR	0x04	W	W	Command Register	0x00	
SR	0x08	R	R	Status Register	0x3C	
IR	0x0C	R	R	Interrupt Register	0x00	
IER	0x10	RW	RW	Interrupt Enable Register	0x00	
///	0x14			Reserved (returns 00h when read)	—	
BTR0	0x18	R	RW	Bus Timing 0 Register	0x00	
BTR1	0x1C	R	RW	Bus Timing 1 Register	0x00	
///	0x20			Reserved		
///	0x24			Reserved		
///	0x28			Reserved (returns 00h when read)		
ALC	0x02C	R	R	Arbitration Lost Capture Register	0x00	
ECC	0x30	R	R	Error Code Capture Register	0x00	
EWLR	0x34	R	RW	Error Warning Limit Register	0x60	
RXERR	0x38	R	RW	Receive Error Counter Register	0x00	
TXERR	0x3C	R	RW	Transmit Error Counter Register	0x00	
Transmit Buffer	0x40	W	RW	Transmit Frame Information Register (read back from 0x180)		2
	0x44 - 0x70	W	RW	Transmit Data Information (read back from 0x184 - 0x1B0)		2
Receive Window	0x40	R	RW	Receive Frame Information Register		2
	0x44 - 0x70	R	RW	Receive Data Information		2
ACR0	0x40	R	RW	Acceptance Code Register 0	0x00	3
ACR1	0x44	R	RW	Acceptance Code Register 1	0x00	3
ACR2	0x48	R	RW	Acceptance Code Register 2	0x00	3
ACR3	0x4C	R	RW	Acceptance Code Register 3	0x00	3
AMR0	0x50	R	RW	Acceptance Mask Register 0	0x00	3
AMR1	0x54	R	RW	Acceptance Mask Register 1	0x00	3
AMR2	0x58	R	RW	Acceptance Mask Register 2	0x00	3
AMR3	0x5C	R	RW	Acceptance Mask Register 3	0x00	3
RMC	0x74	R	R	Receive Message Counter Register	0x00	
RBSA	0x78	R	RW	Receive Buffer Start Address Register	0x00	
///	0x7C			Reserved	0x7C	
///	0x80 - 0x17C	R	RW	Receive FIFO		
///	0x180 - 0x1B0	R	R	Transmit Buffer		
///	0x1B4 - 0x1FC			Reserved (returns 00h when read)		

NOTES:

1. The Mode Register sets the behavior of the CAN Controller. Bits can be set or reset from the CPU, which sees the Mode Register as part of its Read/Write memory. Reserved bits are read as '0'.
2. Receive data is read from same CAN address where transmit data is written (0x40-0x70). However, transmit data may be read back from 0x180-1B0.
3. The Mode Register sets the behavior of the CAN Controller. Bits can be set or reset from the CPU, which sees the Mode Register as part of its Read/Write memory. Reserved bits are read as '0'.

26.18 ADC Registers

Base address: 0xFFFC3000

Table 26-22. ADC Register Summary

NAME	ADDRESS OFFSET	TYPE	RESET VALUE	DESCRIPTION
HW	0x00	R	0x0000	High Word Register
LW	0x04	R	0x0000	Low Word Register
RR	0x08	R	0x0000	Results Register
IM	0x0C	RW	0x0000	Interrupt Masking Register
PC	0x10	RW	0x0000	Power Configuration Register
GC	0x14	RW	0x0000	General Configuration Register
GS	0x18	R	0x0210	General Status Register
IS	0x1C	R	0x0010	Interrupt Status Register
FS	0x20	R	0x0004	FIFO Status Register
HWCB0 - HWCB15	0x24 - 0x60	RW	0x0000	High Word Control Bank Registers
LWCB0 - LWCB15	0x64 - 0xA0	RW	0x0000	Low Word Control Bank Registers
IHWCTRL	0xA4	RW	0x0000	Idle High Word Registers
ILWCTRL	0xA8	RW	0x0000	Idle Low Word Registers
MIS	0xAC	R	0x0000	Masked Interrupt Status
IC	0xB0	W		Interrupt Clear Register

Chapter 27
Glossary

AD-TFT

Advanced Thin-Film Transistor Liquid Crystal Display. Capable of using either transmitted light or reflected light to form a display.

AHB

Advanced High-performance Bus. Defined in the AMBA specification, the AHB connects the high-performance blocks within the SoC.

ALI

Advanced LCD Interface. Allows for direct connection of the SoC to the Row and Column Driver chips in newer superthin panels that do not incorporate a separate timing ASIC.

AMBA

Advanced Microprocessor Bus Architecture. This architecture is an open standard for a high-speed on-chip bus connecting the blocks of an SoC.

APB

Advanced Peripheral Bus. Defined in the AMBA specification, the APB connects the low power, lower-bandwidth peripheral blocks within the SoC.

APB Bridge

Interface between the AHB and APB. The APB Bridge converts data into a suitable format for the slave devices on the APB. It is a slave on the AHB and the only bus master on the APB.

ARM7TDMI-S Core

Synthesizeable version of the ARM7TDMI Central Processing Unit, belonging to the ARM7 family of processors. For more information, see the ARM Ltd. website: www.arm.com.

Big-endian

The most significant byte or half-word of the data is stored at the lowest storage address or transmitted or received first.

Block

In Flash Memory terminology, a Block (or sector) is a logical group of memory cells that need to be erased together. Blocks are usually 64KB in size, but other sizes exist (8KB boot blocks for example).

BPP

Bits-Per-Pixel.

Byte

An 8-bit data element. Bytes in this User's Guide are shown with the most significant bit on the left (or top) and the least significant bit on the right (or bottom). Also see Half Word and Word.

Byte Lane

A data path that is one byte wide.

CAN

Controller Area Network. A serial bus protocol specifically designed for electrically noisy environments. For more information, see www.can.bosch.com

Chip

A functional element made by dividing a portion of semiconductor wafer material, sometimes referred to as a 'die' or plural, 'dice'.

CLCDC

The on-chip Color Liquid Crystal Display Controller.

Core

See ARM7TDMI-S Core.

CPSR

Current Program Status Register. In ARM architecture, the CPSR is the register that stores the condition code bits.

CSTN

Color Super-Twist Nematic. A type of color Liquid Crystal Display that uses transmitted light to form a display.

Embedded SRAM

Static Random Access Memory that is present in the processor for application use. The LH75400/01/10/11 has 32KB of embedded SRAM (eSRAM).

Endianness

Describes the bit, byte, or word sequence of data communication or storage, associating the most significant or least significant end of a data sequence with the lowest address or with the beginning of reception or transmission. See Big-endian and Little-endian.

FIQ

Fast Interrupt Request. For more information, refer to ARM Architecture Manual at the ARM Ltd. website: www.arm.com.

GPIO

General Purpose Input and Output

Half-Word

In the context of 32-bit SoCs, a half-word is an ordered pair of bytes totaling 16-bits. Half-words are always shown with the MSB at the left and the LSB on the right.

HBP

Horizontal Back Porch. The HBP, whose name is derived from the shape of the signal in the time domain, is the delay between when the LCD clock is pulsed to index to a new line and when the actual pixel data is sent to the LCD driver.

HFP

Horizontal Front Porch. The HFP, whose name is derived from the shape of the signal in the time domain, is the delay between the end of each line, or row, of pixels before the LCD line clock is pulsed to index to the next line.

HR-TFT

Highly Reflective Thin Film Transistor; a type of Liquid Crystal Display that relies upon reflected light rather than transmitted light to form a display.

HSW

Horizontal Synchronization Pulse Width. The HSW is the width of the pixel clock when in passive mode, or the width of the horizontal synchronization pulse in active mode. The required width may vary from one LCD panel to another, and the required HSW width should be determined from the particular LCD data sheet.

IrDA

Infrared Data Association. More commonly, IrDA refers to the specification defined and maintained by this group that is the de facto standard for wireless, infrared serial communications.

IRQ

Interrupt Request. For more information, refer to ARM Architecture Manual at the ARM Ltd. website: www.arm.com.

ISR

Interrupt Service Routine.

k

'kilo-' prefix. 1,000 Ω = 1 kΩ.

KB

Kilobyte. A unit of measurement signifying 1,024 bytes of memory, as in 64KB of memory.

Kb

A unit of measurement representing a quantity of 1,024 bits.

Little-endian
The least significant part of the data is stored at the lowest storage address or transmitted or received first.

LSb
Least significant bit of an ordered sequence.

LSB
Least significant byte of an ordered sequence.

LSW
Least significant word of an ordered sequence.

LPP
Lines Per Panel. The quantity of active lines per LCD screen, or 'panel'.

M
'Mega-' prefix. 1,000,000 Ω = 1 MΩ.

MCU
Microcontroller, or SoC.

Mask
Bit pattern, or (by itself), to disable. A Masking register enables or disables something.

MB
Megabyte, 1,024K

MSb
Most-significant bit of a byte, half-word or word.

MSB
Most-significant byte of a half-word or word.

MSW
Most significant word of an ordered sequence.

Non-Volatile Memory
A memory technology that retains its contents when power is removed. Examples are ROM and Flash. Also see Volatile Memory.

Pixel
Picture Element. The smallest controllable unit of a matrix LCD display.

R
In register tables: Read Only

RTC

Real Time Clock. The RTC provides a periodic interrupt that is referenced to real time. Generally, the RTC is referenced to 1 second (1 Hz). The interrupt is programmable to occur after *n* 1-second ticks, set in a 32-bit register. The interrupt can be used to control other peripherals or software tasks requiring reference to real-time.

RW

In register tables: Read or Write. RW fields may be read or written.

SoC

System-on-Chip. A microprocessor and supporting peripherals on a single chip.

SSP

Synchronous Serial Port. The SSP implements a serial interface that can communicate with external devices using the Motorola SPI interface, Texas Instruments Synchronous Serial Interface, or National Semiconductor MICROWIRE™ interface. The SSP performs all serial-to-parallel and parallel-to-serial data conversions, and can work with data frames ranging from 4 to 16 bits.

STN

Super-Twist Nematic (Grayscale) Liquid Crystal Display. Uses transmitted light to form a display.

SWI

Software Interrupt. An SWI causes an interrupt to be asserted due to a decision within the software, rather than in response to a hardware stimulus. SWIs can be assigned priority within the VIC, just as hardware interrupts.

TCM

Tightly Coupled Memory. Memory that is directly connected to CPU core, bypassing the AHB. Allows the CPU core to access this memory without AHB access overhead. The LH75400/01/10/11 has 16KB of TCM.

UART

Universal Asynchronous Receiver and Transmitter. A UART's primary function is to provide the translation and timing necessary to convert serial data streams to parallel data groups and vice-versa. The UARTs on the LH754xx convert the internal parallel data to an external serial data stream, add necessary start and stop bits, generate parity if required, and clock the serial data at a programmed baud rate.

USB

Universal Serial Bus. USB defines the hardware and protocol for a 'plug and play' serial communications interface, which has largely replaced the once-standard serial port on PCs and other devices. USB devices require virtually no intervention by the user one the device is connected to the bus. USB hosts communicate with USB clients.

VBP

Vertical Back Porch. The VBP, whose name is derived from the shape of the signal in the time domain, is the delay between the vertical synchronization signal indexes to a new pixel data frame, and when the actual pixel data is sent to the LCD driver.

VIC

Vectored Interrupt Controller. The VIC hardware sorts and prioritizes interrupts, allowing interrupts to cause automatic routing (vectoring) of the software to unique interrupt service routines (ISR). This allows faster interrupt servicing than requiring software to determine the interrupting source prior to entering the proper ISR. The VIC also allows prioritizing interrupts such that a higher priority interrupt can interrupt a lower priority ISR.

VFP

Vertical Front Porch. The VFP, whose name is derived from the shape of the signal in the time domain, is the delay between the actual pixel data sent to the LCD driver, and the vertical synchronization signal indexes.

Volatile Memory

A general term for any memory technology that loses its contents when power is removed. Examples are RAM, SRAM and SDRAM. See Non-Volatile Memory.

VSW

Vertical Synchronization Pulse Width. The quantity of vertical synchronization lines fed to an LCD panel.

W

In register tables, Write-Only. Write Only fields should not be read as the data is invalid.

WDT

Watchdog Timer. The WDT can be programmed to reset the SoC if it is not periodically serviced by software. This allows for a 'sanity check', which can prevent error conditions from locking up the SoC. If the WDT is not serviced within the programmed time, it is assumed that a non-recoverable error has occurred and the system must be reset to clear the error. The LH75400/01/10/11 WDT can be programmed to issue a reset upon timeout, or an interrupt. The interrupt condition allows an ISR to attempt to resolve the error prior to a system reset. In this mode, a second consecutive timeout of the WDT results in a system reset.

Word

In the context of 32-bit SoCs, a word is an ordered set of four bytes, totaling 32-bits. Words are always shown with the MSB at the left and the LSB on the right.

SPECIFICATIONS ARE SUBJECT TO CHANGE WITHOUT NOTICE.

Suggested applications (if any) are for standard use; See Important Restrictions for limitations on special applications. See Limited Warranty for SHARP's product warranty. The Limited Warranty is in lieu, and exclusive of, all other warranties, express or implied. ALL EXPRESS AND IMPLIED WARRANTIES, INCLUDING THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR USE AND FITNESS FOR A PARTICULAR PURPOSE, ARE SPECIFICALLY EXCLUDED. In no event will SHARP be liable, or in any way responsible, for any incidental or consequential economic or property damage.

SHARP®

NORTH AMERICA

SHARP Microelectronics of the Americas
5700 NW Pacific Rim Blvd.
Camas, WA 98607, U.S.A.
Phone: (1) 360-834-2500
Fax: (1) 360-834-8903
www.sharpsma.com

EUROPE

SHARP Microelectronics Europe
Division of Sharp Electronics (Europe) GmbH
Sonninstrasse 3
20097 Hamburg, Germany
Phone: (49) 40-2376-2286
Fax: (49) 40-2376-2232
www.sharpsme.com

JAPAN

SHARP Corporation
Electronic Components & Devices
22-22 Nagaike-cho, Abeno-Ku
Osaka 545-8522, Japan
Phone: (81) 6-6621-1221
Fax: (81) 6117-725300/6117-725301
www.sharp-world.com

TAIWAN

SHARP Electronic Components
(Taiwan) Corporation
8F-A, No. 16, Sec. 4, Nanking E. Rd.
Taipei, Taiwan, Republic of China
Phone: (886) 2-2577-7341
Fax: (886) 2-2577-7326/2-2577-7328

SINGAPORE

SHARP Electronics (Singapore) PTE., Ltd.
438A, Alexandra Road, #05-01/02
Alexandra Technopark,
Singapore 119967
Phone: (65) 271-3566
Fax: (65) 271-3855

KOREA

SHARP Electronic Components
(Korea) Corporation
RM 501 Geosung B/D, 541
Dohwa-dong, Mapo-ku
Seoul 121-701, Korea
Phone: (82) 2-711-5813 ~ 8
Fax: (82) 2-711-5819

CHINA

SHARP Microelectronics of China
(Shanghai) Co., Ltd.
28 Xin Jin Qiao Road King Tower 16F
Pudong Shanghai, 201206 P.R. China
Phone: (86) 21-5854-7710/21-5834-6056
Fax: (86) 21-5854-4340/21-5834-6057
Head Office:
No. 360, Bashen Road,
Xin Development Bldg. 22
Waigaoqiao Free Trade Zone Shanghai
200131 P.R. China
Email: smc@china.global.sharp.co.jp

HONG KONG

SHARP-ROXY (Hong Kong) Ltd.
3rd Business Division,
17/F, Admiralty Centre, Tower 1
18 Harcourt Road, Hong Kong
Phone: (852) 28229311
Fax: (852) 28660779
www.sharp.com.hk
Shenzhen Representative Office:
Room 13B1, Tower C,
Electronics Science & Technology Building
Shen Nan Zhong Road
Shenzhen, P.R. China
Phone: (86) 755-3273731
Fax: (86) 755-3273735