March 2007
Revision 1.0

WPC8763L Notebook Embedded Controller with SPI™ Flash Interface

General Description

The Winbond WPC8763L is a highly integrated embedded controller (EC) with an embedded RISC core and integrated advanced functions. It is targeted for a wide range of portable applications.

The WPC8763L incorporates the CompactRISC® CR16CPlus core (a high-performance 16-bit RISC processor), on-chip ROM and RAM memories, system support functions and a Flash Interface Unit (FIU) that interfaces directly with external SPI flash memory devices.

System support functions include: watchdog, PWM, timers, interrupt control, General-Purpose I/O (GPIO) with internal keyboard matrix scanning, PS/2® interface, SMBus® interface, UART, high-accuracy analog-to-digital (ADC) and digital-to-analog (DAC) converters for battery charging, system control, system health monitoring and analog controls.

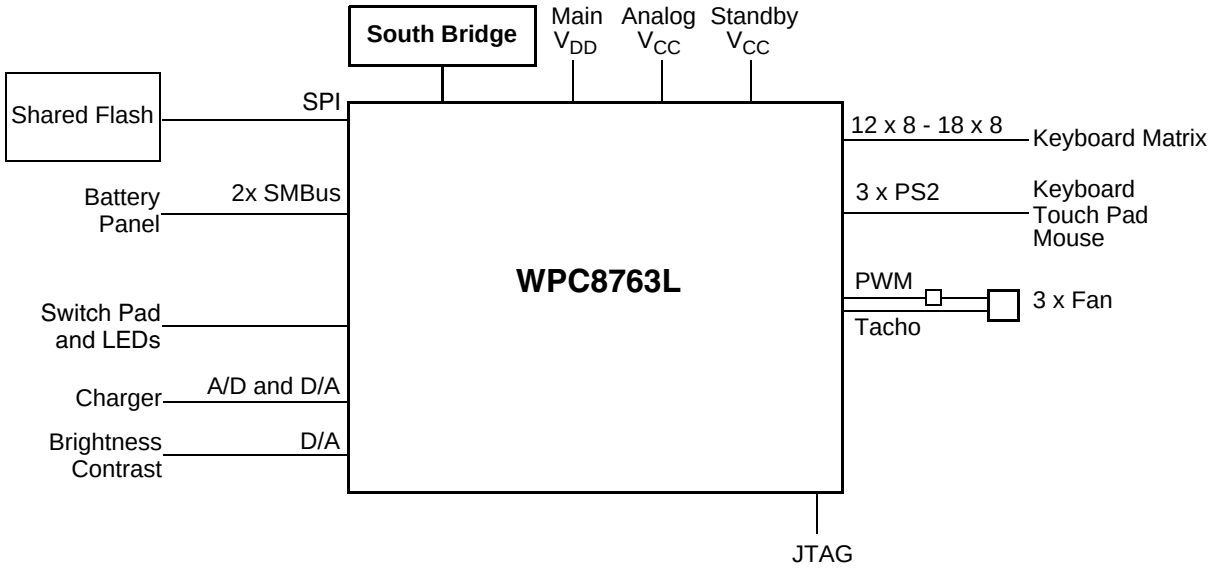
The WPC8763L interfaces with the host via an LPC interface.

The WPC8763L is PC01 and ACPI compliant, and offers a single-chip solution for the most commonly used notebook PC I/O peripherals.

Outstanding Features

- Shared BIOS flash memory
- Support for SPI flash memories
- Flash page programing support
- High-accuracy, high-speed ADC
- Up to 88 GPIO ports (including keyboard scanning) with a variety of wake-up events
- 16-bit RISC core, with up to 4 Mbytes of external address space, running at up to 25 MHz
- 128-pin LQFP package

System Block Diagram



Features

Embedded Controller Features

- Processing Unit
  - CompactRISC CR16CPlus 16-bit embedded RISC processor core (the “core”)
  - Up to 4 Mbytes of external address space
- Internal Memory
  - 1 Kbyte of ROM
  - 4 Kbytes of on-chip RAM
  - All memory types can hold both code and data
- Flash Interface Unit (FIU)
  - Up to 4 Mbytes of code and data
  - Hardware-protected boot zone block protection
  - SPI External Memory
    - Up to 32 Mbits
    - Fast Read mode
    - Page programing support
    - Configurable clock rate
  - Field upgradeable
- Shared Memory Controller (SHM)
  - Supports BIOS (flash) memory sharing with PC host
  - Supports host-controlled code download and update
  - Memory access protection

LPC System Interface

- Based on Intel’s *LPC Interface Specification Revision 1.1*, August 2002
- I/O, Memory and 8-bit Firmware Memory read and write cycles, Firmware Memory writes may insert wait cycles
- Bootable Memory Support
- Base Address (BADDR1-0) straps to determine the base address of the index-data register pair
  - Alternate base address configurable by the core
- $\overline{\text{LPCPD}}$ and $\overline{\text{CLKRUN}}$ support

Embedded Controller System Features

- Host Interface
  - Comprises host interface channels, typically used for KBC and ACPI Private or Shared EC channels
  - 8042 KBC-standard interface (legacy 60h, 64h)
  - Two PM interface ports (legacy 62h, 66h; 68h, 6Ch)
  - ACPI EC with either Shared or Private interface through the PM interface
  - Two Mailbox areas for host-core communication, up to 4 Kbytes each; maximum 4 Kbytes total
  - Generates IRQ, SMI and SCI
  - Provides IRQ1 and IRQ12 support
  - Provides Fast Gate A20 and Fast Host reset via firmware
- Interrupt Control Unit (ICU)
  - 31 maskable vectored interrupts (of which eight are external)
  - General-purpose external interrupt inputs through MIWU
  - Enable and pending indication for each interrupt
  - Non-maskable interrupt input
- Multi-Input Wake-Up (MIWU)
  - Up to 40 wake-up or interrupt inputs
  - Generates wake-up event to PMC (Power Management Controller)
  - Generates interrupts to ICU
  - User-selectable trigger conditions
- Internal Keyboard Matrix Scanning
  - Up to 18 open-collector outputs (at least 12)
  - Eight Schmitt inputs with internal pull-ups
- General-Purpose I/O (GPIO) Ports
  - 64 port pins
  - I/O pins individually configured as input or output
  - Configurable internal pull-up / pull-down resistors
  - Outputs individually configured as push-pull or open-drain
  - Two echo inputs with wake-enabled interrupts
  - Additional 12 GPIOs with wake-enabled interrupts
  - Four GPIOs capable of 12 mA sink current
  - Seven GPIOs are accessible to the host
  - Optional low-cost external GPIO expansion through the SensorPath interface
- PS/2 Interface
  - Three external ports: can be used for keyboard, mouse and an additional pointing device
  - Byte-level handling via hardware accelerator
- Two SMBus (SMB) Interface Modules; each module:
  - Is Intel SMBus, Philips $I^2C^{®}$ and ACCESS.bus compatible
  - Is SMBus master and slave
  - Supports up to two simultaneous slave addresses
  - Supports polling- and interrupt-controlled operations
  - Generates a wake-up signal on detection of a Start condition while in Idle mode
  - Supports an optional internal pull-up on SDA and SCL pins
- Core Universal Asynchronous Receiver-Transmitter (CR_UART) Module
  - A full-duplex UART channel
  - Programmable baud rate
  - Data transfer via interrupt or polling
- Two 16-bit Multi-Function Timer (MFT16) Modules; each module has:
  - Two 16-bit timers with a 5-bit prescaler
  - Pulse Width Modulation (PWM), Capture and Timer/Counter modes
  - Capture inputs with programmable edge detection
  - An interrupt on compare match

Features (Continued)

- Two Pulse Width Modulation (PWM) Modules
  - Group A_PWM: two outputs
  - Group B_PWM: one output
- Timer and Watchdog (TWD)
  - 16-bit periodic interrupt timer with 30 µs resolution and 5-bit prescaler for system tick and periodic wake-up tasks
  - 8-bit watchdog timer with enable/disable
  - "Watchdog occurred" flag
  - Two watchdog reset options: warm or cold
- SensorPath™ Bus Interface
  - Single Wire bus master
  - Supports up to seven slave devices
  - x1, x4 SensorPath clock rate support
- Analog-to-Digital Converter (ADC)
  - Four channels, with 8-bit resolution
  - 125 µs conversion time
  - External voltage reference
- Digital-to-Analog Converter (DAC)
  - Two channels, 8-bit resolution
  - 1 µs conversion time for 50 pF load
  - Full output range from AGND to AVCC
- Development Support
  - Interface to debugger via Nexus 5001 interface
    - ❏ Physical connection using JTAG
    - ❏ On-board Debug mode with eight hardware breakpoints
    - ❏ Embedded memory programing via JTAG with content read protection
- Core Access to Host Modules
  - Enabled via lock mechanism

Host Function Features

- Mobile System Wake-Up Control (MSWC)
  - Software-controlled off events
  - Event routing to IRQ, $\overline{\text{SMI}}$ or $\overline{\text{PWUREQ}}$
- Supports *Microsoft® Advanced Power Management (APM) Specifications Revision 1.2*, February 1996
  - Generates the System Management Interrupt (SMI)
- PC01 Rev 1.0 and ACPI 3.0 Compliant
  - PnP configuration register structure
  - Flexible resource allocation for all logical devices
    - ❏ Relocatable base address
    - ❏ 15 IRQ routing options

Clocking, Supply and Package Information

- Strap Input-Controlled Operating Modes:
  - Shared BIOS memory mode
  - TRI-STATE® mode
  - Development mode
- Clocks
  - Single 32.768 KHz crystal oscillator
  - On-chip high-frequency clock generator
  - Either 32.768 KHz or CR16CPlus clock out
- Testability
  - XOR-tree structure includes all device pins (except supply, A/D, D/A, and crystal oscillator pins), selected at power-up by strap inputs
  - TRI-STATE device pins, selected at power-up by strap input ($\overline{\text{TRIS}}$)
- Power Supply
  - 3.3V supply operation
  - 5V tolerance and back-drive protection on all pins (except crystal oscillator, A/D, D/A, LPC bus, and SPI flash pins)
  - Separate supply for host I/F (V_{DD}) and EC functions (V_{CC})
  - Reduced power consumption capability
  - Software- or hardware-switched power modes:
    - ❏ Active mode
    - ❏ Active mode executing WAIT
    - ❏ Idle
    - ❏ Deep Idle
  - Automatic wake-up on system events
- Package Options
  - 128-pin LQFP package

Datasheet Revision Record

Revision Date	Status	Comments
October 4, 2006	Revision 0.96	First Preliminary Datasheet.
March 12, 2007	Revision 1.0	Datasheet revision. List of changes: 1. In cover page and Features: 1.1 Changed the order of the Host Function Features. 2. In Detailed Signal/Pin Descriptions, PS/2 Interface, added clarification on quasi-bi-directional buffers. 3. In Decoupling and Filter Capacitors, changed the 10-47 µF capacitors to tantalum/ceramic. 4. In SPI Flash Interface, UMA section, added clarification on byte order in UMA Extended Write transactions. 5. In MIWU, Input Assignments table: 5.1 Improved readability of footnote #3. 5.2 Added footnote #4 to $\overline{\text{LRESET}}$ Active Wake-Up source
March 12, 2007 (cont.)	Revision 1.0 (cont.)	Datasheet revision. List of changes (cont.): 6. In PWM Block Diagram, reversed the “R” and “S” inputs of the flip-flops. 7. In SPI, Features, changed the data clock rate to 10 MHz. 8. In PM Channels: 8.1 In Enhanced PM Mode, IRQ, $\overline{\text{ECSCI}}$ and $\overline{\text{SMI}}$ Control figure, changed the signal names related to the $\overline{\text{SMI}}$ output. 8.2 To the description of SCIPOL bit (HIPMnCTL register) and SMIPOL bit (HIPMnIC register), added recommendation to use active low polarity setting. 9. In MSWC Core registers, MSWCTL1 register, added clarification to LPCRSTA bit. 10. In Booter Data, Header Contents table, updated footnote #2. 11. In Power Supply Current Consumption... updated the I_{DD}, I_{DDLP}, I_{CC} and I_{CCW} parameters. 12. In DC Characteristics of Pins: 12.1 Added Leakage Current section. 12.2 In Notes and Exceptions, updated item #1 and removed items #4 and #5 (they are replaced by the new Leakage Current section). 13. In Internal Resistors, changed the Min. Max. values for the R_{PU80} and R_{PD80} parameters. 14. In DAC Characteristics: 14.1 Changed the values of output resistance (R_S). 15. In AC Test Conditions, added C_L = 15 pF for CLKOUT. 16. In PS/2 Interface Timing, changed the “guaranteed by” for t_{PSCLKL} and t_{PSCLKH}.

Table of Contents

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

1.0 Introduction

1.1 GENERAL DESCRIPTION

The WPC8763L is a highly integrated, embedded controller (EC) with an embedded RISC core and system functions. It is targeted for a wide range of portable applications that use the Low Pin Count (LPC) interface and is designed to best meet the requirements of mobile systems.

1.1.1 System Connections

The major elements of the WPC8763L are:

- Core with internal ROM and RAM.
- EC functions, which include: PS/2 devices, keyboard matrix scanning, SMBus, UART, timers, PWM, SensorPath interface, D/A and A/D converters and GPIO pins that can be assigned to various functions, as needed. External memory and peripheral devices may be added to extend the functionality of the on-chip resources.
- Flash interface, supporting SPI external flash devices.
- Host Processor interface, based on the LPC bus and additional signals for interrupts, keyboard interface and system power management.
- Power supplies for host interface, host functions, core and EC functions.
- Clocks (using a 32.768 KHz crystal) and optional clock output.
- Strap inputs, for initializing the WPC8763L operation mode.

In addition to the wide range of internal modules, the WPC8763L provides hooks so that the system can be expanded in an easy and cost-effective manner, as follows:

- I/O expansion to support additional I/O port pins, using low-cost, SensorPath-based GPIO expanders.

1.1.2 Power Management

The WPC8763L has an advanced power management mechanism controlled by the host and/or the WPC8763L firmware. The core may also interface with the host via one or two I/O communication channels and also via the mailbox communication channel. Power Management events are available for ACPI-compliant operation with the host chipset.

The WPC8763L is designed to operate as the EC of an ACPI-compliant system. It is equipped with various system power monitor and control functions and advanced methods to control its own power consumption and power modes.

The host functions are ACPI compliant, including all associated activity level control and wake-up methods. The host function I/O buffers are powered by V_{DD}, which can be turned off when the system enters power states S3 and lower.

ACPI-compliant wake-up and sleep control is integrated into the WPC8763L. This function is powered by V_{CC}.

The Keyboard controller and the EC functions (core and associated modules) are powered by V_{CC}, which remains active as long as the system has a power source (e.g., main battery or AC outlet). Using V_{CC}, the core may be programed to monitor and control the system even when the host processor is turned off. To support this, the WPC8763L is equipped with the following advanced methods to control its power consumption:

- Software-controlled clock frequency
- The ability to disable modules
- Power modes:
  - Active - Full functionality
  - Active Executing WAIT Instruction - Core execution and associated operations (such as memory access) are suspended.
  - Idle - The core clock domain is stopped; the device can be woken up by internal or external events. The system tick timer is still operational and can be used to wake up the device periodically.
  - Deep Idle - Same as Idle, but the core clock domain generator is also stopped.

1.2 INTERNAL ARCHITECTURE

The WPC8763L consists of several main components, divided into three groups:

- Core Domain
  - CR16CPlus core processing unit
  - Flash Interface Unit (FIU)
  - RAM and ROM memory
  - Core modules
- Host Domain
  - Host functions
  - Host interface
- Host-to-EC Interface

1.0 Introduction (Continued)

1.3 OPERATING ENVIRONMENTS

The WPC8763L has two operating environments:

- In-system Regular Environment (IRE)
- On Board Development (OBD)

The $\overline{\text{JEN0}}$ strap pin or $\overline{\text{JENK}}$ strap pin selects the environment after V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset.

1.3.1 IRE Environment

IRE environment is used for WPC8763L operation in the production system and for normal execution of applications. The flash memory is the main source of code for the device. In this environment, after reset, the WPC8763L starts running the code written in the first address of the internal ROM.

The WPC8763L is shipped with 1 Kbyte of on-chip boot code. The user is expected to use an external memory for most of the code and constant data.

In this environment, the $\overline{\text{JEN0}}$ strap pin or $\overline{\text{JENK}}$ strap pin does not need an external pull-down resistor.

1.3.2 OBD Environment

OBD environment is used for debugging the WPC8763L firmware while the device is mounted on the final production board. All pins have IRE functionality except for the pins used for the JTAG interface. The JTAG-based debugger interface allows communication with a debugger running on a host. In OBD environment, as in IRE environment, code is executed from the flash memory. Breakpoints on data and code access may be applied using the core hardware breakpoint mechanism. Erasing and reprograming the flash is enabled through the JTAG-based debugger interface.

OBD environment is binary and cycle-by-cycle compatible with IRE environment.

In OBD environment, the $\overline{\text{JEN0}}$ strap pin or $\overline{\text{JENK}}$ strap pin requires an external pull-down resistor.

1.4 MEMORY MAP

The WPC8763L has two address domains: core and host. Section 1.4.1 discusses the mapping of memories and modules into the core address space. Section 1.4.2 discusses the mapping of the host address space and ways of accessing it.

The WPC8763L enables several areas in the core address space to have restricted access to the host. These areas are referred to as “shared memory” (or “shared BIOS memory”). In addition, the core can access the host functions. Section 1.4.2 and Section 1.4.3 discuss cross-domain access (shared memory and core access to host modules, respectively).

1.4.1 Core Address Domain Memory Map

The memory and I/O devices are mapped directly into the lowest 16 Mbytes of the core address space. The core address space can include both code and data; however, access to data stored in the first 1 Mbyte of address space is more efficient.

Most of the core code and constant data is stored in the flash. The code contains a ROM-based core boot block. The on-chip RAM and the various modules are also mapped in the core address space.

Table 1 shows how the WPC8763L memory and modules are mapped in the core address space. Appendix A on page 302 shows the detailed address map of the modules.

Addresses not included in the following table or Appendix A are reserved. Attempts to access reserved addresses (including non-implemented areas) generate an Illegal Address Trap (IAD).

Table 1. Memory Map

Address	Size (Bytes)		Description
	Allocated	Implemented[1]	
00 0000h – 00 FFFFh	64K	1 Kbyte	ROM
01 0000h – 01 FFFFh	64K	4 Kbytes	Core System RAM
02 0000h – 41 FFFFh	4M	128K to 4M	Flash
FF 0000h – FF 07FFh	2K	-	On-Chip Module Registers
FF F000h – FF FFFFh	4K	-	On-Chip Module Registers

1. The implemented addresses start from the bottom of the allocated address space.

1.0 Introduction (Continued)

Register Abbreviations and Access

The following abbreviations are used to indicate the Register Type:

- R/W = Read/Write.
- R = Read from register (data written to this address is sent to a different register).
- W = Write (see above).
- RO = Read-only. Writing to the register/bit is ignored.
- ROC = Read-only, clear bits. Reading from the register clears all its bits.
- RUP = Read unpredictable. Reading from the register returns an unpredictable value; writing to the register is ignored.
- WO = Write-only. Reading from the register/bit returns 0.
- R/W1C = Read/Write 1 to Clear. Writing 1 to a bit clears it to 0. Writing 0 has no effect.
- R/W1S = Read/Write 1 to Set. Writing 1 to a bit sets its value to 1. Writing 0 has no effect.

Access Rules for Memory and On-Chip Registers

Byte-wide, word-wide or double word-wide transactions to any address within the memory address space may be used to access memory devices. An exception to this is a core-to-flash write transaction, which should not exceed the configured write burst length; see Section 4.1.2 on page 55.

The register access width must match the register width; e.g., only byte-wide transactions may be used to access byte-wide registers. Attempts to read a write-only register or write to a read-only register may cause unpredictable results.

Zeros must be written to reserved bits unless stated otherwise. Reading reserved bits returns an undefined value. When modifying a register with reserved bits, the data read from a reserved bit can be written back to that bit.

Flash Memory Mapping into Shared BIOS Memory

When the shared BIOS memory is enabled using the SHBM strap input (SHBM=0) or the Shared Memory configuration registers (LDN=0Fh), the flash memory address range is mapped into the address range of the host. See Section 5.4 on page 216 for details of the memory mapping mechanism.

1.4.2 Host Address Domain Memory Map

The host address space includes memory space and I/O space.

The I/O space used by the WPC8763L is configured via the WPC8763L configuration registers. The configuration register address is defined by strap inputs (BADDR0-1) to be either one of two fixed addresses or an address defined by the core using registers in the MSWC module.

When a Shared BIOS memory mechanism is enabled via the SHBM strap input, the WPC8763L is mapped to enable the host to boot from a shared flash device. The WPC8763L supports either Memory or Firmware Memory transactions for the host memory interface, with automatic selection between them. Section 5.4 on page 216 discusses the mapping of memory between the host and core domains; it also discusses the read and write access protection mechanism from the host and core domains.

Following the boot process, the Shared Memory configuration registers (see Section 6.1.13 on page 263) enable setting memory sharing. The configuration setting includes defining the memory protocol in use (Memory or Firmware Memory) and the address range used in the host address space. The configuration registers allow the defaults set by the SHBM strap input to be overridden.

Host Access to Core RAM

The host and core may use a semaphore mechanism to access the same on-chip RAM space for data interchange. Access is controlled by the EC. Mapping of either LPC Memory or Firmware Memory transactions to the core bus address space is based on the Host Access Control registers (see Section 6.1.13 on page 263) for host base address and window size. Two host address windows may be mapped to the core internal RAM. The size of each window is configurable up to a maximum of 4 Kbytes and together up to a maximum of 4 Kbytes.

1.4.3 Core Access to Host Modules

The core can access host domain modules through the “Core Access to Host Modules” bridge. The bridge employs an indirect mapping mechanism, as described in Section 5.3 on page 211.

There is only a single set of module registers for host and core use. The bus arbitration guarantees that only one of the two register accesses occurs at any given time; however, this does not prevent problems that may be caused by conflicting write transactions. When such a conflict is expected, the core lock mechanism can be used to protect access to one or more of the modules. The lock can also be used for security reasons, to prevent the host from accessing modules.

1.0 Introduction (Continued)

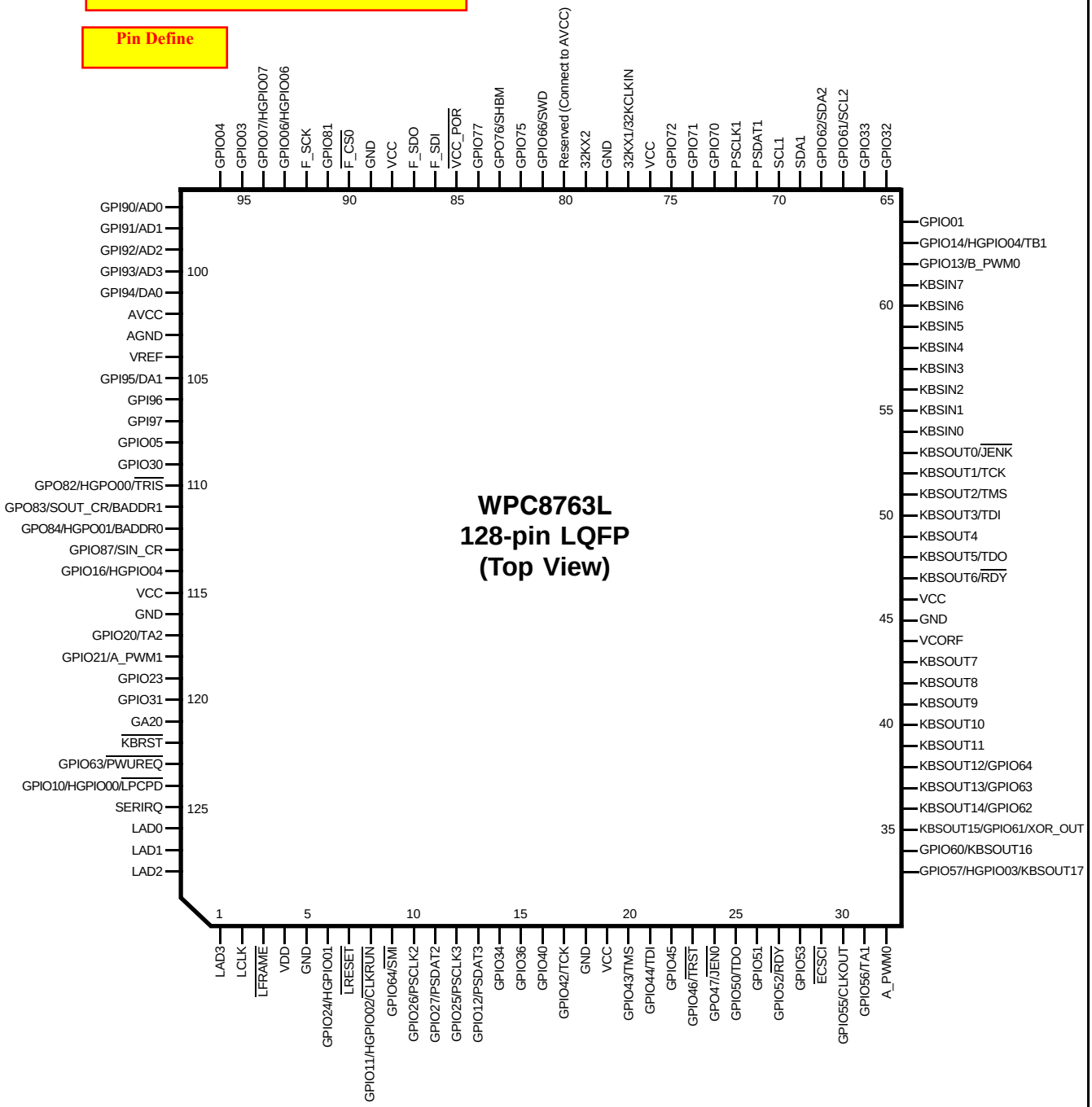
To access a register, the core specifies the logical device and the offset of the register within the logical device. Note that the configuration index and data registers are also considered as a logical device. The core triggers a read by writing 1 to the read start bit. It then waits for the bit to clear. When the bit clears, the core can read the data from the data register. The core triggers a write by performing a write to the data register. It then waits for a status bit to clear; when the status bit clears, the core can proceed to the next operation.

2.0 Signal/Pin Description and Configuration

2.1 CONNECTION DIAGRAM

The symbol and geometry same as 71.08763.A0G

Pin Define



128-Pin Low-Profile Plastic Quad Flatpack (LQFP)
Order Number WPC8763LDG

2.0 Signal/Pin Description and Configuration (Continued)

2.2 BUFFER TYPES

The signal DC characteristics of the pins described in Section 2.4 on page 25 are denoted by a buffer type symbol described briefly below and in further detail in Section 8.2 on page 279.

Table 2. Buffer Types

Symbol	Description
IN_{AD}	Input, analog to ADC
IN_{CS}	Input, CMOS compatible, with Schmitt Trigger
IN_{OSC}	Input, from crystal oscillator (not characterized)
IN_{PCI}	Input, PCI 3.3V
IN_{SM}	Input, SMBus compatible
IN_{T}	Input, TTL compatible
IN_{TS}	Input, TTL compatible, with Schmitt Trigger
$O_{p/n}$	Output, push-pull output buffer that is capable of sourcing *p* mA and sinking *n* mA
OD_{n}	Output, open-drain output buffer that is capable of sinking *n* mA
O_{DA}	Output, analog from DAC
O_{OSC}	Output, to crystal oscillator (not characterized)
O_{PCI}	Output, PCI 3.3V
PWR	Power pin
GND	Ground pin

2.3 PIN MULTIPLEXING

Important Note: Table 3 shows only multiplexed pins and the configuration bits for the selection of the multiplexed options used in the device. A configuration select bit on the right side of the table has priority over configuration select bits to its left. If none of the configuration select bits are set, the default signal is selected. For example, if the bit in Configuration Select 3 column is set, the signal in Alternate Signal 3 column is selected regardless of the value of the bits in Configuration Select 2 and 1 columns. If the bit in Configuration Select 3 column is cleared and the bit in Configuration Select 2 column is set, the signal in Alternate Signal 2 column is selected regardless of the value of the bit in Configuration Select 1 column, and so on.

Table 3. Pin Multiplexing Configuration

Default Signal	Alternate Signal 1	Configuration Select 1	Alternate Signal 2	Configuration Select 2	Alternate Signal 3	Configuration Select 3
GPIO24	HGPIO01	DEVALT8.HGP01_SL				
GPIO10	HGPIO00	DEVALT8.HGP00_SL	$\overline{\text{LPCPD}}$	DEVALT2.LPCPD_SL		
GPIO11	HGPIO02	DEVALT8.HGP02_SL	$\overline{\text{CLKRUN}}$	DEVALT2.CLKRN_SL		
GPIO64	$\overline{\text{SMI}}$	DEVALT2.SMI_SL				
GPIO63	$\overline{\text{PWUREQ}}$	DEVALT2.PWUR_SL				
GPIO56	TA1	DEVALT3.TA1_SL				
GPIO20	TA2	DEVALT3.TA2_SL				
GPIO14	HGPIO04	DEVALT8.HGP04_SL	TB1	DEVALT3.TB1_SL		
GPIO21	A_PWM1	DEVALT5.APWM1_SL				

2.0 Signal/Pin Description and Configuration (Continued)

Table 3. Pin Multiplexing Configuration (Continued)

Default Signal	Alternate Signal 1	Configuration Select 1	Alternate Signal 2	Configuration Select 2	Alternate Signal 3	Configuration Select 3
GPIO13	B_PWM0	DEVALT5.BPWM0_SL				
GPI90	AD0	DEVALT6.ADC0_SL				
GPI91	AD1	DEVALT6.ADC1_SL				
GPI92	AD2	DEVALT6.ADC2_SL				
GPI93	AD3	DEVALT6.ADC3_SL				
GPIO07	HGPIO07	DEVALT8.HGP07_SL				
GPI94	DA0	DEVALT4.DAC0_SL				
GPI95	DA1	DEVALT4.DAC1_SL				
GPIO61	SCL2	DEVALT2.SMB2_SL				
GPIO62	SDA2					
GPIO26	PSCLK2	DEVALT4.PS2_2SL				
GPIO27	PSDAT2					
GPIO25	PSCLK3	DEVALT4.PS2_3SL				
GPIO12	PSDAT3					
KBSOUT12	GPIO64	DEVALT7.KBO12_SL				
KBSOUT13	GPIO63	DEVALT7.KBO13_SL				
KBSOUT14	GPIO62	DEVALT7.KBO14_SL				
KBSOUT15	GPIO61	DEVALT7.KBO15_SL	XOR_OUT	Test		
GPIO60	KBSOUT16	DEVALT7.KBO16_SL				
GPIO57	HGPIO03	DEVALT8.HGP03_SL	KBSOUT17	DEVALT7.KBO17_SL		
GPIO55	CLKOUT	DEVALT0.CLKOUT_SL				
GPO47					$\overline{\text{JEN0}}$	Strap
GPIO42					TCK	DEVALT0.JTAG_SL
GPIO43					TMS	
GPIO44					TDI	
GPIO46					$\overline{\text{TRST}}$	
GPIO50					TDO	
GPIO52					$\overline{\text{RDY}}$	
KBSOUT0					$\overline{\text{JENK}}$	Strap
KBSOUT1					TCK	STRPST.JENK
KBSOUT2					TMS	
KBSOUT3					TDI	
KBSOUT5					TDO	
KBSOUT6					$\overline{\text{RDY}}$	
GPO82	HGPO00	DEVALT9.HGP00_SL			$\overline{\text{TRIS}}$	Strap
GPIO16	HGPIO04	DEVALT9.HGP04_SL				
GPO84	HGPO01	DEVALT9.HGP01_SL			BADDR0	Strap
GPIO87	SIN_CR	DEVALT1.UART_SL				

2.0 Signal/Pin Description and Configuration (Continued)

Table 3. Pin Multiplexing Configuration (Continued)

Default Signal	Alternate Signal 1	Configuration Select 1	Alternate Signal 2	Configuration Select 2	Alternate Signal 3	Configuration Select 3
GPO83	SOUT_CR	DEVALT1.UART_SL			BADDR1	Strap
GPIO06	HGPIO06	DEVALT8.HGP06_SL				
GPIO76					SHBM	Strap
GPIO66	SWD	DEVALT0.SNP_SL				

2.4 DETAILED SIGNAL/PIN DESCRIPTIONS

This section describes all signals of the WPC8763L device. The signals are organized by functional group.

The "✔" symbol in the "5VT" column indicates that those pins are 5-volt tolerant and back-drive protected.

2.4.1 Host Interface

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
$\overline{\text{CLKRUN}}$	8	I/O	IN_{PCI}/OD_6		V_{DD}	**Clock Run.** Same as PCI $\overline{\text{CLKRUN}}$. When high, it indicates that the LPC clock will be stopped. In this case, the WPC8763L may pull it down to request full speed of the clock.
GA20	121	O	$O_{1/2}$		V_{CC}	**Gate A20.** See Section 5.5.4 on page 235 for signal operation and behavior when V_{DD} is off.
$\overline{\text{KBRST}}$	122	O	$O_{1/2}$		V_{CC}	**Keyboard Reset Output.** See Section 5.5.4 on page 235 for signal operation and behavior when V_{DD} is off.
LAD3-0	1, 128, 127, 126	I/O	IN_{PCI}/O_{PCI}		V_{DD}	**LPC Address-Data.** Multiplexed command, address bidirectional data and cycle status.
LCLK	2	I	IN_{PCI}		V_{DD}	**LPC Clock.** PCI clock (0 or 33 MHz).
$\overline{\text{ECSCI}}$	29	I/O	$IN_{TS}/O_{2/10}$	✔	V_{CC}	**EC SCI.** Generates an EC SCI interrupt to the chipset. This signal is typically connected to one of the chipset GPI inputs.
$\overline{\text{LFRAME}}$	3	I	IN_{PCI}		V_{DD}	**LPC Frame**. Low pulse indicates the beginning of a new LPC cycle or the termination of a broken cycle.
$\overline{\text{LPCPD}}$	124	I	IN_{PCI}		V_{CC}	**Power Down.** Indicates that the LPC interface power will be turned off.
$\overline{\text{LRESET}}$	7	I	IN_{PCI}		V_{DD}	**LPC Reset.** A level low reset to the LPC interface and host function configuration registers and Shared Memory host registers.
$\overline{\text{PWUREQ}}$	123	O	$O_{1/2}$	✔	V_{CC}	**Power-Up Request.**
SERIRQ	125	I/O	IN_{PCI}/O_{PCI}		V_{DD}	**Serial IRQ.** The interrupt requests are serialized over a single pin, where each IRQ level is delivered during a designated time slot.
$\overline{\text{SMI}}$	9	I/O	IN_{TS}/OD_{10}	✔	V_{CC}	**System Management Interrupt.**

2.0 Signal/Pin Description and Configuration (Continued)

2.4.2 SPI Flash Interface

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
F_SDI	86	I	IN_{TS}		V_{CC}	**Serial Data Input.**
F_SDO	87	O	O_{PCI}		V_{CC}	**Serial Data Output.**
F_SCK	92	O	O_{PCI}		V_{CC}	**Serial Clock.**
$\overline{F\_CS0}$	90	O	O_{PCI}		V_{CC}	**Chip Select 0.**

2.4.3 Internal Keyboard Scan

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
KBSIN7-0	61, 60, 59, 58, 57, 56, 55, 54	I	IN_{CS}	✔	V_{CC}	**Keyboard Scan Inputs.** The input side of the internal keyboard scan lines.
KBSOUT 17-0	33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 47, 48, 49, 50, 51, 52, 53	O	OD_6	✔	V_{CC}	**Keyboard Scan Outputs.** The output side of the internal keyboard scan lines.

2.0 Signal/Pin Description and Configuration (Continued)

2.4.4 General-Purpose I/O (GPIO)

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
GPIO01	64	I/O	$IN_{TS}/O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports with Wake-Up/Interrupt and Echo Input.**
GPIO03	95		IN_{TS}/OD_2			
GPIO05-04	108, 96		IN_{TS}/OD_2		V_{CC}	**General-Purpose I/O Ports with Wake-Up/Interrupt.**
GPIO06	93		$IN_{TS}/O_{2/4}$	✔		
GPIO07	94		IN_{TS}/OD_2			
GPIO10	124		$IN_{TS}/O_{2/4}$	✔	V_{CC}	
GPIO11	8		$IN_{TS}/O_{3/6}$	✔	V_{DD}	
GPIO14-12	63, 62, 13		$IN_{TS}/O_{2/4}$	✔	V_{CC}	
GPIO16	114		$IN_{TS}/O_{2/4}$	✔	V_{CC}	
GPIO21-20	118, 117		$IN_{TS}/O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports with Echo Output.**
GPIO23	119		$IN_{TS}/O_{2/4}$	✔	V_{CC}	
GPIO24	6		IN_{TS}/O_{PCI}		V_{DD}	**General-Purpose I/O Ports.**
GPIO27-25	11, 10, 12		$IN_{TS}/O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports.**
GPIO33-30	66, 65, 120, 109		$IN_{TS}/O_{2/20}$	✔		**General-Purpose I/O Ports with LED Drive.**
GPIO34	14		$IN_{TS}/O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports.**
GPIO36	15			✔		
GPIO40	16			✔		
GPIO46-42	23, 22, 21, 20, 17			✔		
GPO47	24	O	$O_{2/4}$	✔		**General-Purpose Output Ports.**
GPIO53-50	28, 27, 26, 25	I/O	$IN_{TS}/O_{2/4}$	✔		**General-Purpose I/O Ports.**
GPIO57-55	33, 31, 30			✔		
GPIO60	34			✔		
GPIO61	35		$IN_{TS}/O_{2/4}$	✔		
	67		$IN_{SM}/O_{2/4}$	✔		
GPIO62	36		$IN_{TS}/O_{2/4}$	✔		
	68		$IN_{SM}/O_{2/4}$	✔		
GPIO63	37		$IN_{TS}/O_{2/4}$	✔		
	123			✔		
GPIO64	38		$IN_{TS}/O_{2/4}$	✔		
	9			✔		
GPIO66	81		$IN_{TS}/O_{2/4}$	✔		
GPIO72-70	75, 74, 73			✔		
GPIO75	82			✔		

2.0 Signal/Pin Description and Configuration (Continued)

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
GPO76	83	O	$O_{2/4}$	✔	V_{CC}	**General-Purpose Output Ports.**
GPIO77	84	I/O	$IN_{TS}/O_{2/4}$	✔		**General-Purpose I/O Ports.**
GPIO81	91	I/O	$IN_{TS}/O_{2/4}$	✔		
GPO84-82	112, 111, 110	O	$O_{2/4}$	✔		**General-Purpose Output Ports.**
GPIO87	113	I/O	$IN_{TS}/O_{2/4}$	✔		**General-Purpose I/O Ports.**
GPI97-90	107, 106, 105, 101, 100, 99, 98, 97	I	IN_{TS}			**General-Purpose Input Ports with Wake-Up/Interrupt.**

2.4.5 Host General-Purpose I/O (HGPIO)

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
HGPO00	110	O	$O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports with Wake-Up/Interrupt.**
HGPIO00	124	I/O	$IN_{TS}/O_{2/4}$	✔		
HGPIO01	6	I/O	IN_{TS}/O_{PCI}		V_{DD}	**General-Purpose I/O Ports with Wake-Up/Interrupt.**
HGPO01	112	O	$O_{2/4}$	✔	V_{CC}	
HGPIO02	8	I/O	$IN_{TS}/O_{3/6}$	✔	V_{DD}	**General-Purpose I/O Ports with Wake-Up/Interrupt.**
HGPIO 04-03	114 or 63, 33	I/O	$IN_{TS}/O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports with Wake-Up/Interrupt.**
HGPIO06	93	I/O	$IN_{TS}/O_{2/4}$	✔	V_{CC}	**General-Purpose I/O Ports with Wake-up/Interrupt.**
HGPIO07	94	I/O	IN_{TS}/OD_2		V_{CC}	**General-Purpose I/O Ports with Wake-up/Interrupt.**

2.4.6 Analog Interface

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
AD3-0	100, 99, 98, 97	I	IN_{AD}		AV_{CC}	**Analog to Digital Converter Inputs.**
DA1-0	105, 101	O	O_{DA}		AV_{CC}	**Digital to Analog Converter Outputs.**
VREF	104	I	IN_{REF}		AV_{CC}	**Reference Voltage.**

2.4.7 PS/2 Interface

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
PSCLK3-1	12, 10, 72	I/O	$IN_T/O_{2/12}$	✔	V_{CC}	**PS/2 Channel 1 through 3 Clock signal.** This signal has a quasi-bidirectional buffer (i.e., both $O_{2/12}$ and OD_{12}) - see "Quasi-Bidirectional Drivers" on page 92.
PSDAT3-1	13, 11, 71	I/O	$IN_T/O_{2/12}$	✔	V_{CC}	**PS/2 Channel 1 through 3 Data signal.** This signal has a quasi-bidirectional buffer (i.e., both $O_{2/12}$ and OD_{12}) - see "Quasi-Bidirectional Drivers" on page 92.

2.0 Signal/Pin Description and Configuration (Continued)

2.4.8 SMBus Interface

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
SCL2-1	67, 70	I/O	IN_{SM}/OD_4	✔	V_{CC}	**SMBus Serial Clock 1 to 2 Signals**. An internal pull-up for this pin is optional.
SDA2-1	68, 69	I/O	IN_{SM}/OD_4	✔	V_{CC}	**SMBus Serial Data 1 to 2 Signals**. An internal pull-up for this pin is optional.

2.4.9 Timers and PWM

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
TA1	31	I/O	$IN_{TS,}$ $O_{1/2}$	✔	V_{CC}	**Timer Pin A for Timers 2 and 1.**
TA2	117					
TB1	63	I	IN_{TS}	✔	V_{CC}	**Timer Pin B for Timer 1.**
A_PWM1-0	118, 32	O	$O_{3/6}$	✔	V_{CC}	**PWM Type A Outputs 1-0.**
B_PWM0	62	O	$O_{3/6}$	✔	V_{CC}	**PWM Type B Output 0.**

2.4.10 SensorPath Interface

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
SWD	81	I/O	IN_{SM}/OD_4	✔	V_{CC}	**SensorPath Data.** Bidirectional, SensorPath Data interface signal to GPIO expande. An internal pull-up for this pin is optional.

2.4.11 CR_UART

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
SIN_CR	113	I	IN_{TS}	✔	V_{CC}	**Receive Data.**
SOUT_CR	111	O	$O_{3/6}$	✔	V_{CC}	**Transmit Data.**

2.4.12 JTAG

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
$\overline{RDY}$	27	O	$O_{4/4}$	✔	V_{CC}	**Ready.** Nexus specific JTAG transfer ready.
	47	O	$O_{3/6}$	✔	V_{CC}	
TDO	25	O	$O_{4/4}$	✔	V_{CC}	**Test Data Output.** Serial JTAG Data Output.
	48	O	$O_{3/6}$	✔	V_{CC}	
$\overline{TRST}$	23	I	IN_{TS}	✔	V_{CC}	**Test Reset.** TAP Controller reset.
TDI	21 or 50	I	IN_{TS}	✔	V_{CC}	**Test Data Input.** Serial JTAG Data Input.
TMS	20 or 51	I	IN_{TS}	✔	V_{CC}	**Test Mode Select.** Selects TAP controller states.
TCK	17 or 52	I	IN_{TS}	✔	V_{CC}	**Test Clock.** Serial shift clock for JTAG interface.

2.0 Signal/Pin Description and Configuration (Continued)

2.4.13 Miscellaneous

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
$\overline{VCC\_POR}$	85	I/O	IN_{TS}/OD_6	✔	V_{CC}	**V_{CC} Power On Reset Output.** Also functions as EC reset input (see Section 3.2.7 on page 50). Must be connected to an external pull-up resistor (to V_{CC} supply) even if the pin is not used.

2.4.14 Clocks

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
32KCLKIN	77	I	IN_T		AV_{CC}	**32.768 KHz Clock Input.**
32KX1	77	I	IN_{OSC}		AV_{CC}	**32.768 KHz Crystal Oscillator Input.** Input from external crystal oscillator circuitry; see Section 3.3.1 on page 51.
32KX2	79	O	O_{OSC}		AV_{CC}	**32.768 KHz Crystal Oscillator Output.** Output to external crystal oscillator circuit.
CLKOUT	30	O	$O_{10/10}$	✔	V_{CC}	**Clock Output.** Clock Output. Output pin for either the 32.768 KHz clock or the core clock domain (CLK). Recommended only for debugging.

2.4.15 Strap Configuration and Testing

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
BADDR1-0	111, 112	I	IN_{TS}	✔	V_{CC}	**I/O Base Address.** Sampled at V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset, to determine the base address of the configuration index-data register pair as follows: • No pull-down resistor: 164Eh-164Fh • 10 KΩ external pull-down resistor on BADDR0: 2Eh-2Fh • 10 KΩ external pull-down resistor on BADDR1: Core defined • 10 KΩ external pull-down resistor on BADDR0 and BADDR1:XOR-Tree Test Mode. Note: $\overline{TRIS}$ must be 1, i.e., left unconnected.
$\overline{TRIS}$	110	I	IN_{TS}	✔	V_{CC}	**TRI-STATE.** Forces the device to float all its output and I/O pins (except for DAC outputs and 32KX2) if an external 10 KΩ pull-down resistor is connected. Sampled at V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset. When $\overline{TRIS}$ is set to 0 (by an external pull-down resistor), BADDR0 or BADDR1 must be 1 (i.e., left unconnected).
$\overline{JEN0}$, $\overline{JENK}$	24, 53	I	IN_{TS}	✔	V_{CC}	**JTAG Select.** Sampled at V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset, to select the JTAG signals to device pins (see Table 4 on page 31 for details). Both $\overline{JEN0}$ and $\overline{JENK}$, are pulled to 1 by an internal resistor. The external 10 KΩ pull-down resistor must be connected to GND.

2.0 Signal/Pin Description and Configuration (Continued)

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
SHBM	83	I	IN_{TS}	✔	V_{CC}	**Shared Host BIOS Memory.** Sampled at V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset, to determine the state of the shared BIOS memory. SHBM is pulled to 1 (disables the shared BIOS memory) by an internal resistor or set to 0 by an external 10 KΩ pull-down resistor to enable shared BIOS memory.
XOR_OUT	35	O	$O_{4/8}$	✔	V_{CC}	**XOR-Tree Output.** All the device pins (except power type and analog type pins) are internally connected in a XOR-tree structure.

Table 4. JTAG Select Configuration

$\overline{JEN0}$ (Pin 24)	$\overline{JENK}$ (Pin 53)	Functionality of Pins 17, 20, 21, 23 25, 27	Functionality of Pins 47, 48, 50, 51, 52
No pull-down resistor		GPIO Port signals	Keyboard Scan outputs
10 KΩ external pull-down resistor	No pull-down resistor	JTAG signals	Keyboard Scan outputs
No pull-down resistor	10 KΩ external pull-down resistor	GPIO Port signals	JTAG signals
10 KΩ external pull-down resistor		Illegal Strap Combination	

2.4.16 Power and Ground

Signal	Pin(s)	I/O	Buffer Type	5VT	Power Well	Description
AGND	103	I	GND	N/A[1]		**Analog Ground.** Serves as ground for the Analog to Digital Converter (ADC), the Digital to Analog Converter (DAC) and the 32K oscillator.
AVCC	102	I	PWR	N/A		**Analog 3.3V Power supply.** Serves as the power supply for the Analog to Digital Converter (ADC), the Digital to Analog Converter (DAC) and the 32K oscillator.
GND	116, 89, 78, 45, 18, 5	I	GND	N/A		**Ground.** Serves as ground for main, standby and back-up battery powered internal logic and output drivers; see Section 3.1.3 on page 47 for details on connections with AGND.
VCC	115, 88, 76, 46, 19	I	PWR	N/A		**Standby Digital 3.3V Power Supply.** Serves as the power supply for the core-controlled functions and all associated I/O pins.
Reserved	80	N/A	N/A	N/A		**Must be connected to AV_{CC}.**
VCORF	44	I/O	PWR	N/A		**On-Chip Core Power Converter Filter.** On-chip core power converter output. Powers the core logic of all the device modules. An external 1 µF ceramic filter capacitor must be connected between this pin and GND.
VDD	4	I	PWR	N/A		**Main Digital 3.3V Power Supply.** Serves as the power supply for the LPC interface and some of the host functions.

1. “Not Applicable.”

2.0 Signal/Pin Description and Configuration (Continued)

2.5 INTERNAL PULL-UP AND PULL-DOWN RESISTORS

The signals listed below have internal pull-up (PU) and/or pull-down (PD) resistors; see Section 8.3 on page 284 for the values of each resistor type. Some of the internal resistors are enabled when the respective signals are selected to the device pins. Other internal resistors are controlled by the respective bits in the P*x*PULL and P*x*PUD GPIO configuration registers (see Section 4.4 on page 81). In the second case, the respective bit in the P*x*DIR register must be set to 0; also, if the signal is powered by V_{DD}, the respective bit in the P*x*ENVDD register must be set to 1. Internal resistors not labeled as “programmable” in the tables below cannot be disabled.

2.5.1 Internal Keyboard Scan

Signal	Pin(s)	Type	Comments
KBSIN7-0	61, 60, 59, 58, 57, 56, 55, 54	PU_{30}	Programmable, defaults to float

2.5.2 General-Purpose I/O (GPIO)

Signal	Pin(s)	Type	Comments
GPIO01	64	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO07-03	94, 93, 108, 96, 95	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO10	124	PU_{30}/PD_{30}	Programmable, defaults to PU
GPIO11	8	PU_{30}/PD_{30}	Programmable, defaults to float
GPIO14-12	63, 62, 13	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO16	114	PU_{80}/PD_{80}	Programmable, defaults to PD
GPIO21-20	118, 117	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO23	119	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO24	6	PU_{30}/PD_{30}	Programmable, defaults to PU
GPIO27-25	11, 10, 12	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO31-30	120, 109	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO33-32	66, 65	PU_{30}/PD_{30}	Programmable, defaults to float
GPIO34	14	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO36	15	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO40	16	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO46-42	23, 22, 21, 20, 17	PU_{80}/PD_{80}	Programmable, defaults to float
GPO47	24	PU_{30}/PD_{30}	Programmable, defaults to float[1]
GPIO53-50	28, 27, 26, 25	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO57-55	33, 31, 30	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO60	34	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO62-61	36, 35	PU_{80}/PD_{80}	Programmable, defaults to float
	68, 67	PU_{30}/PD_{30}	
GPIO64-63	38, 37 or 9, 123	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO66	81	$PU_{1.25}/PD_{30}$	Programmable, defaults to float

2.0 Signal/Pin Description and Configuration (Continued)

Signal	Pin(s)	Type	Comments
GPIO70	73	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO71	74	PU_{80}/PD_{80}	Programmable, defaults to PD
GPIO72	75	PU_{80}/PD_{80}	Programmable, defaults to float
GPIO75	82	PU_{80}/PD_{80}	Programmable, defaults to float
GPO76	83	PU_{30}/PD_{30}	Programmable, defaults to float[1]
GPIO77	84	PU_{80}/PD_{80}	Programmable, defaults to PU
GPIO81	91	PU_{80}/PD_{80}	Programmable, defaults to float
GPO84-82	112, 111, 110	PU_{30}/PD_{30}	Programmable, defaults to float[1]
GPIO87	113	PU_{80}/PD_{80}	Programmable, defaults to float

1. For details, see Section 4.4.3 on page 82.

2.5.3 Host General-Purpose I/O (HGPIO)

Signal	Pin(s)	Type	Comments
HGPIO00	124	PU_{30}	Programmable, defaults to PU
HGPO00	110		Programmable, defaults to PU
HGPIO01	6	PU_{30}	Programmable, defaults to PU
HGPO01	112		Programmable, defaults to PU
HGPIO02	8	PU_{30}	Programmable, defaults to PU
HGPIO04-03	114 or 63, 33	PU_{80}	Programmable, defaults to PU
HGPIO07-06	94, 93	PU_{80}	Programmable, defaults to PU

2.5.4 PS/2 Interface

Signal	Pin(s)	Type	Comments
PSCLK3-1	12, 10, 72	PU_{80}	Programmable, defaults to float
PSDAT3-1	13, 11, 71	PU_{80}	Programmable, defaults to float

2.5.5 SMBus Interface

Signal	Pin(s)	Type	Comments
SCL2-1	67, 70	PU_{30}	Programmable, defaults to float
SDA2-1	68, 69	PU_{30}	Programmable, defaults to float

2.0 Signal/Pin Description and Configuration (Continued)

2.5.6 Timers

Signal	Pin(s)	Type	Comments
TA1	31	PU_{80}/PD_{80}	Programmable via GPIO[1], defaults to float
TA2	117	PU_{80}/PD_{80}	Programmable via GPIO[1], defaults to float
TB1	63	PU_{80}/PD_{80}	Programmable via GPIO[1], defaults to float

1. See details in Section 2.5 on page 32.

2.5.7 SensorPath Interface

Signal	Pin(s)	Type	Comments
SWD	81	$PU_{1.25}$	Programmable, defaults to float

2.5.8 JTAG

Signal	Pin(s)	Type	Comments
$\overline{TRST}$	23	PU_{80}	
TDI	21 or 50	PU_{80}	
TMS	20 or 51	PU_{80}	
TCK	17 or 52	PD_{80}	

2.5.9 Strap Configuration and Testing

Signal	Pin(s)	Default Type	Comments
BADDR1-0	111, 112	PU_{30}	strap
SHBM	83	PU_{30}	strap
$\overline{TRIS}$	110	PU_{30}	strap
$\overline{JEN0}$	24	PU_{30}	strap
$\overline{JENK}$	53	PU_{80}	strap

2.0 Signal/Pin Description and Configuration (Continued)

2.6 SYSTEM CONFIGURATION

The System Configuration provides the following functions:

- Controls the general WPC8763L configuration
- Provides the status of the WPC8763L straps
- Controls pin multiplexing via the DEVALTx registers
- Controls the connection of some of the GPIO ports to the MIWU
- Controls some of the pull-up resistors (for SMBus and SensorPath pins)

2.6.1 System Configuration Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

All system configuration registers are powered by V_{CC} and are set to their default values by Core Domain reset unless otherwise specified in the register description below.

Register Map

Location	Mnemonic	Register Name	Size	Type
FF F000h	DEVCNT	Device Control	Byte	R/W
FF F001h	STRPST	Straps Status	Byte	RO
FF F002h	RSTCTL	Reset Control and Status	Byte	Varies per bit
FF F010h	DEVALT0	Device Alternate Function 0	Byte	Varies per bit
FF F011h	DEVALT1	Device Alternate Function 1	Byte	R/W
FF F012h	DEVALT2	Device Alternate Function 2	Byte	R/W
FF F013h	DEVALT3	Device Alternate Function 3	Byte	R/W
FF F014h	DEVALT4	Device Alternate Function 4	Byte	R/W
FF F015h	DEVALT5	Device Alternate Function 5	Byte	R/W
FF F0**35**h	DEVALT6	Device Alternate Function 6	Byte	R/W
FF F016h	Reserved			
FF F017h	DEVALT7	Device Alternate Function 7	Byte	R/W
FF F018h	DEVALT8	Device Alternate Function 8	Byte	R/W
FF F019h	DEVALT9	Device Alternate Function 9	Byte	R/W
FF F020h	Reserved			
FF F021h	GES1	GPIO Event Select 1	Byte	R/W
FF F028h	DEVPU0	Device Pull-Up 0	Byte	R/W
FF F029h	DEVPU1	Device Pull-Up 1	Byte	R/W

2.0 Signal/Pin Description and Configuration (Continued)

Device Control Register (DEVCNT)

Location: FF F000h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**							**CLKOM**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-1	**Reserved.**
0	**CLKOM (Clock Out Mode).** Selects the internal clock signal routed to the CLKOUT pin. If CKOUT_SL bit in DEVALT0 (page 37) is 1: 0: Core clock (CLK); see Section 4.19 on page 181 (default). 1: 32.768 KHz clock (LFCLK). Otherwise this bit is ignored.

Straps Status Register (STRPST)

STRPST enables the software to read the value set to strap pins. All the strap pins are sampled (and STRPST register is updated) at V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset.

Location: FF F001h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**TRIS**	**JENK**	**JEN0**	**Reserved**	**BADDR1**	**BADDR0**	**SHBM**
Reset	0	Strap	Strap	Strap	0	Strap	Strap	Strap

Bit	Description
7	**Reserved.**
6	**TRIS (TRI-STATE Test Mode).** Controls the TRI-STATE condition of all WPC8763L pins. 0: WPC8763L pins are floating (TRI-STATE condition). 1: WPC8763L pins have normal functionality.
5	**JENK (JTAG Over KBSOUT Enable).** Selects the JTAG interface to the corresponding KBSOUT pins. The value of this bit is set according to the value of the $\overline{JENK}$ strap (see Table 4 on page 31). 0: TCK, TMS, TDI, TDO, $\overline{RDY}$ - JTAG interface. 1: KBSOUT1-3, KBSOUT5-6 - Keyboard Scan outputs.
4	**JEN0 (JTAG Enable).** Controls the JTAG_SL bit in DEVALT0 register; see page 37. JTAG_SL selects the JTAG signals to the device pins. The value of this bit is set according to the value of the $\overline{JEN0}$ strap (see Table 4 on page 31). 0: JTAG_SL set to 1. 1: JTAG_SL set to 0.
3	**Reserved.**
2-1	**BADDR1 and BADDR0 (Base Address Select).** Selects either the base address of the host configuration index-data register pair or the XOR-Tree Test mode. For more information see Section 6.1.1 on page 246.
0	**SHBM (Shared Host BIOS Memory).** Controls the host BIOS shared memory functionality. This enables the host to boot from a shared flash device. For more information see Section 5.4.2 on page 216. 0: Shared BIOS memory is enabled. 1: Shared BIOS memory is disabled.

2.0 Signal/Pin Description and Configuration (Continued)

Reset Control and Status Register (RSTCTL)

Location: FF F002h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**DBGRST_MODE**	**HIPRST_MODE**	**Reserved**				**DBGRST_STS**	**VCC_POR_STS**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W	**DBGRST_MODE (Debugger Reset Mode Select).** Selects the mode of the Debugger reset. Cleared by V_{CC} Power-Up reset. 0: Debugger **Cold** reset: All modules, registers and bits that are reset by a Watchdog Cold reset are set to their default values also by a Debugger reset (default). 1: Debugger **Warm** reset: Only the core, the debugger interface and the PMC power state are set to their default values by a Debugger reset.
6	R/W	**HIPRST_MODE (Host Interface Modules Reset Mode Select).** Selects the reset source for both the host and the core interface registers of the Keyboard and Mouse interface and Power Management Channels 1 and 2. Cleared by V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset. 0: The host interface registers and the core interface registers of KBC, PM1 and PM2 are reset only by Core Domain reset (default). 1: The host interface registers and the core interface registers of KBC, PM1 and PM2 are reset by both Core Domain reset **and** either V_{DD} Power-Up reset or Host Hardware reset.
5-2		**Reserved.**
1	R/W1C	**DBGRST_STS (Debugger Reset Status).** Indicates the occurrence of a Debugger reset. The type of Debugger reset (Cold or Warm) is according to the setting of the DBGRST_MODE bit. Cleared by V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset. 0: No Debugger reset from the last V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset (default). 1: Debugger reset occurred.
0	RO	**VCC_POR_STS ($\overline{VCC\_POR}$ Input Status).** Indicates the occurrence of $\overline{VCC\_POR}$ input reset. Cleared by V_{CC} Power-Up reset. 0: No $\overline{VCC\_POR}$ Input reset from the last V_{CC} Power-Up reset (default). 1: $\overline{VCC\_POR}$ Input reset occurred.

Device Alternate Function 0 Register (DEVALT0)

Location: FF F010h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**SNP_SL**	**JTAG_SL**	**Reserved**		**CKOUT_SL**	**Reserved**	
Reset	0	0	strap	0	0	0	0	0

Bit	Type	Description
6	R/W	**SNP_SL (SensorPath Select).** Selects the SensorPath interface to the corresponding pin. 0: GPIO66 - GPIO port (default). 1: SWD - SensorPath interface.
5	RO	**JTAG_SL (JTAG Select).** Selects the JTAG interface to the corresponding pins. The value of this bit is set according to the value of $\overline{JEN0}$ strap (see Table 4 on page 31). 0: GPIO42-44, GPIO46 GPIO50, GPIO52 - GPIO ports. 1: TCK, TMS, TDI, $\overline{TRST}$, TDO, $\overline{RDY}$ - JTAG interface.

2.0 Signal/Pin Description and Configuration (Continued)

Bit	Type	Description
4-3		**Reserved.**
2	R/W	**CKOUT_SL (Clock Out Select).** Selects the clock out option to the corresponding pin. 0: GPIO55 - GPIO port (default). 1: CLKOUT (see also CLKOM bit in DEVCNT on page 36) - Clocks.
1-0		**Reserved.**

Device Alternate Function 1 Register (DEVALT1)

Location: FF F011h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**				**UART_SL**	**Reserved**		
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-4	**Reserved.**
3	**UART_SL (CR_UART Select).** Selects the CR_UART function to the corresponding pins. 0: GPIO87, GPO83 - GPIO port (default). 1: SIN_CR, SOUT_CR - CR_UART.
2-0	**Reserved.**

Device Alternate Function 2 Register (DEVALT2)

Location: FF F012h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**SMB2_SL**	**Reserved**		**PWUR_SL**	**SMI_SL**	**CLKRN_SL**	**LPCPD_SL**	**Reserved**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**SMB2_SL (SMBus2 Select).** Selects the SMB2 function to the corresponding pins. When SMB2_SL bit is set to 0, both KBO15_SL bit **and** KBO14_SL bit in DEVALT7 register **must** be set to 1. 0: GPIO61, GPIO62 - GPIO Ports (default). 1: SCL2, SDA2 - SMB Interface.
4	**PWUR_SL (PWUREQ Select).** Selects the PWUREQ function to the corresponding pins. When PWUR_SL bit is set to 0, KBO13_SL bit in DEVALT7 register **must** be set to 1. 0: GPIO63 - GPIO Ports (default). 1: PWUREQ - LPC Interface.
3	**SMI_SL (SMI Select).** Selects the SMI function to the corresponding pins. When SMI_SL bit is set to 0, KBO12_SL bit in DEVALT7 register **must** be set to 1. 0: GPIO64 - GPIO Ports (default). 1: SMI - LPC Interface.
2	**CLKRN_SL (CLKRUN Select).** Selects the CLKRUN function to the corresponding pin. 0: HGPIO02/GPIO11 (see also HGP02_SL bit in DEVALT8 on page 41) - Host GPIO port or GPIO port (default). 1: CLKRUN - LPC interface.

2.0 Signal/Pin Description and Configuration (Continued)

Bit	Description
1	**LPCPD_SL (LPCPD Select).** Selects the $\overline{\text{LPCPD}}$ function to the corresponding pin. 0: HGPIO00/GPIO10 (see also HGP00_SL bit in DEVALT8 on page 41) - Host GPIO port or GPIO port (default). 1: $\overline{\text{LPCPD}}$ - LPC interface.
0	**Reserved.**

Device Alternate Function 3 Register (DEVALT3)

Location: FF F013h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**TB1_SL**	**TA2_SL**	**TA1_SL**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-3	**Reserved.**
2	**TB1_SL (TB1 Select).** Selects the TB1 function to the corresponding pin. 0: GPIO14/HGPIO04 (see also HGP04_SL bit in DEVALT8 on page 41) - GPIO port or Host GPIO port (default). 1: TB1 - Timer.
1	**TA2_SL (TA2 Select).** Selects the TA2 function to the corresponding pin. 0: GPIO20 - GPIO port (default). 1: TA2 - Timer.
0	**TA1_SL (TA1 Select).** Selects the TA1 function to the corresponding pin. 0: GPIO56 - GPIO port (default). 1: TA1 - Timer.

Device Alternate Function 4 Register (DEVALT4)

Location: FF F014h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**PS2_3_SL**	**PS2_2_SL**	**Reserved**				**DAC1_SL**	**DAC0_SL**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**PS2_3_SL (PS/2 #3 Select).** Selects the PS/2 interface Channel 3 function to the corresponding pins. 0: GPIO25, GPIO12 - GPIO ports (default). 1: PSCLK3, PSDAT3 - PS/2 Interface.
6	**PS2_2_SL (PS/2 #2 Select).** Selects PS/2 interface Channel 2 to the corresponding pins. 0: GPIO26-27 - GPIO ports (default). 1: PSCLK2, PSDAT2 - PS/2 interface.
5-2	**Reserved.**
1	**DAC1_SL (DA1 Select).** Selects Digital to Analog converter output 1 to the corresponding pin. 0: GPI95 - General-Purpose Input port (default). 1: DA1 - Digital to Analog interface.

2.0 Signal/Pin Description and Configuration (Continued)

Bit	Description
0	**DAC0_SL (DA0 Select).** Selects Digital to Analog converter output 0 to the corresponding pin. 0: GPI94 - General-Purpose Input port (default). 1: DA0 - Digital to Analog interface.

Device Alternate Function 5 Register (DEVALT5)

Location: FF F015h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**BPWM0_SL**	**Reserved**		**APWM1_SL**	**Reserved**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-5	**Reserved.**
4	**BPWM0_SL (B_PWM0 Select).** Selects the B_PWM0 function to the corresponding pin. 0: GPIO13 - GPIO Port (default). 1: B_PWM0 - PWM interface. The input buffer of GPIO13 is also selected.
3-2	**Reserved.**
1	**APWM1_SL (A_PWM1 Select).** Selects the A_PWM1 function to the corresponding pin. 0: GPIO21 - GPIO Port (default). 1: A_PWM1 - PWM interface. The input buffer of GPIO21 is also selected.
0	**Reserved.**

Device Alternate Function 6 Register (DEVALT6)

Location: FF F035h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**ADC3_SL**	**ADC2_SL**	**ADC1_SL**	**ADC0_SL**	**Reserved (must be '11')**	
Reset	0	0	0	0	0	0	1	1

Bit	Description
7-6	**Reserved.**
5	**ADC3_SL (AD3 Select).** Selects Analog to Digital converter input 3 to the corresponding pin. 0: GPI93 - GPIO ports (default). 1: AD3 - Analog to Digital Interface.
4	**ADC2_SL (AD2 Select).** Selects Analog to Digital converter input 2 to the corresponding pin. 0: GPI92 - GPIO ports (default). 1: AD2 - Analog to Digital Interface.
3	**ADC1_SL (AD1 Select).** Selects Analog to Digital converter input 1 to the corresponding pin. 0: GPI91 - GPIO ports (default). 1: AD1 - Analog to Digital Interface.
2	**ADC0_SL (AD0 Select).** Selects Analog to Digital converter input 0 to the corresponding pin. 0: GPI90 - GPIO ports (default). 1: AD0 - Analog to Digital Interface.
1-0	**Reserved.** (The bits must be '11'.)

2.0 Signal/Pin Description and Configuration (Continued)

Device Alternate Function 7 Register (DEVALT7)

Location: FF F017h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**KBO17_SL**	**KBO16_SL**	**KBO15_SL**	**KBO14_SL**	**KBO13_SL**	**KBO12_SL**	**Reserved**	
Reset	0	0	1	1	1	1	0	0

Bit	Description
7	**KBO17_SL (KBSOUT17 Select).** Selects the KBSOUT17 function to the corresponding pin. 0: GPIO57/HGPIO03 (see also HGP03_SL bit in DEVALT8 on page 41) - GPIO port or Host GPIO port (default). 1: KBSOUT17- Keyboard scan.
6	**KBO16_SL (KBSOUT16 Select).** Selects the KBSOUT16 function to the corresponding pin. 0: GPIO60 - GPIO port (default). 1: KBSOUT16- Keyboard scan.
5	**KBO15_SL (KBSOUT15 Select).** Selects the KBSOUT15 function to the corresponding pin. When KBO15_SL bit is set to 0**,** SMB2_SL bit in DEVALT2 register **must** be set to 1. 0: GPIO61 - GPIO port. 1: KBSOUT15 - Keyboard Scan (default).
4	**KBO14_SL (KBSOUT14 Select).** Selects the KBSOUT14 function to the corresponding pin. When KBO14_SL bit is set to 0**,** SMB2_SL bit in DEVALT2 register **must** be set to 1. 0: GPIO62 - GPIO port. 1: KBSOUT14 - Keyboard scan (default).
3	**KBO13_SL (KBSOUT13 Select).** Selects the KBSOUT13 function to the corresponding pin. When KBO13_SL bit is set to 0, PWUR_SL bit in DEVALT2 register **must** be set to 1. 0: GPIO63 - GPIO port. 1: KBSOUT13 - Keyboard scan (default).
2	**KBO12_SL (KBSOUT12 Select).** Selects the KBSOUT12 function to the corresponding pin. When KBO12_SL bit is set to 0, SMI_SL bit in DEVALT2 register **must** be set to 1. 0: GPIO64 - GPIO port. 1: KBSOUT12 - Keyboard scan (default).
1-0	**Reserved.**

Device Alternate Function 8 Register (DEVALT8)

Note: If two corresponding bits in DEVALT8 and DEVALT9 are set (i.e., select a Host GPIO channel), two pins are connected to the same channel. In this case both pins are controlled by the output of this HGPIOxx channel. The input of the HGPIOxx channel is connected to the pin controlled by DEVALT8 register.

Location: FF F018h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**HGP07_SL**	**HGP06_SL**	**Reserved**	**HGP04_SL**	**HGP03_SL**	**HGP02_SL**	**HGP01_SL**	**HGP00_SL**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**HGP07_SL (HGP07 Select).** Selects the HGPIO07 function to the corresponding pin. 0: GPIO07 - GPIO port (default). 1: HGPIO07 - Host GPIO port.

2.0 Signal/Pin Description and Configuration (Continued)

Bit	Description
6	**HGP06_SL (HGPIO06 Select).** Selects the HGPIO06 function to the corresponding pin. 0: GPIO06 - GPIO port (default). 1: HGPIO06 - Host GPIO port.
5	**Reserved.**
4	**HGP04_SL (HGPIO04 Select).** Selects the HGPIO04 function to the corresponding pin. If TB1_SL bit in DEVALT3 (page 39) is 0: 0: GPIO14 - GPIO port (default). 1: HGPIO04 - Host GPIO port. Otherwise, HGP04_SL is ignored.
3	**HGP03_SL (HGPIO03 Select).** Selects the HGPIO03 function to the corresponding pin. If KBO17_SL bit in DEVALT7 (page 41) is 0: 0: GPIO57 - GPIO port (default). 1: HGPIO03 - Host GPIO port. Otherwise, HGP03_SL is ignored.
2	**HGP02_SL (HGPIO02 Select).** Selects the HGPIO02 function to the corresponding pin. If CLKRUN_SL bit in DEVALT2 (page 38) is 0: 0: GPIO11 - GPIO port (default). 1: HGPIO02 - Host GPIO port. Otherwise, HGP02_SL is ignored.
1	**HGP01_SL (HGPIO01 Select).** Selects the HGPIO01 function to the corresponding pin. If LDRQ_SL bit in DEVALT2 (page 38) is 0: 0: GPIO24 - GPIO port (default) 1: HGPIO01 - Host GPIO port. Otherwise, HGP01_SL is ignored.
0	**HGP00_SL (HGPIO00 Select).** Selects the HGPIO00 function to the corresponding pin. If LPCPD_SL bit in DEVALT2 (page 38) is 0: 0: GPIO10 - GPIO port (default). 1: HGPIO00 - Host GPIO port. Otherwise, HGP00_SL is ignored.

2.0 Signal/Pin Description and Configuration (Continued)

Device Alternate Function 9 Register (DEVALT9)

Note: If two corresponding bits in DEVALT8 and DEVALT9 are set (i.e., select a Host GPIO channel), two pins are connected to the same channel. In this case both pins are controlled by the output of this HGPIOxx channel. The input of the HGPIOxx channel is connected to the pin controlled by DEVALT8 register.

Location: FF F019h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**HGP04_SL**	**Reserved**		**HGP01_SL**	**HGP00_SL**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-5	**Reserved.**
4	**HGP04_SL (HGPIO04 Select).** Selects the HGPIO04 function to the corresponding pin. 0: GPIO16 - GPIO port (default). 1: HGPIO04 - Host GPIO port.
3-2	**Reserved.**
1	**HGP01_SL (HGPIO01 Select).** Selects the HGPO01 function to the corresponding pin. 0: GPO84 - GPIO port (default). 1: HGPO01- Host GPIO port.
0	**HGP00_SL (HGPIO00 Select).** Selects the HGPO00 function to the corresponding pin. 0: GPO82 - GPIO port (default). 1: HGPO00- Host GPIO port.

GPIO Event Select 1 Register (GES1)

Some GPIO pins can be configured to generate wake-up events and interrupts. Control over the connectivity of these GPIO pins to the MIWU is via GES1 register. To generate a MIWU event, each signal pair of GPIO inputs are multiplexed and one of each pair is selected by the respective bit in GES1 register.

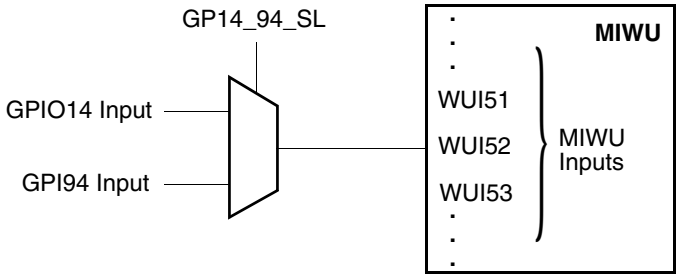


Figure 1. GPIO Connectivity of GPIO14/GPI94 to MIWU

For the connection of GPIO ports to MIWU inputs, see Table 10 on page 66 in the MIWU.

Location: FF F021h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**GP16_96 _SL**	**Reserved**	**GP14_94 _SL**	**GP13_93 _SL**	**GP12_92 _SL**	**GP11_91 _SL**	**GP10_90 _SL**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Reserved.**

2.0 Signal/Pin Description and Configuration (Continued)

Bit	Description
6	**GP16_96_SL (GPIO16 or GPI96 Select).** Selects the GPIO input to connect o the respective MIWU input. 0: GPIO16 is selected (default). 1: GPI96 is selected.
5	**Reserved.**
4	**GP14_94_SL (GPIO14 or GPI94 Select).** Selects the GPIO input to connect to the respective MIWU input. 0: GPIO14 is selected (default). 1: GPI94 is selected.
3	**GP13_93_SL (GPIO13 or GPI93 Select).** Selects the GPIO input to connect to the respective MIWU input. 0: GPIO13 is selected (default). 1: GPI93 is selected.
2	**GP12_92_SL (GPIO12 or GPI92 Select).** Selects the GPIO input to connect to the respective MIWU input. 0: GPIO12 is selected (default). 1: GPI92 is selected.
1	**GP11_91_SL (GPIO11 or GPI91 Select).** Selects the GPIO input to connect to the respective MIWU input. 0: GPIO11 is selected (default). 1: GPI91 is selected.
0	**GP10_90_SL (GPIO10 or GPI90 Select).** Selects the GPIO input to connect to the respective MIWU input. 0: GPIO10 is selected (default). 1: GPI90 is selected.

Device Pull-Up Enable 0 Register (DEVPU0)

DEVPU0 affects the pull-up resistors on the corresponding pins only if the functions described are selected. Otherwise, the bits of this register have no effect.

Location: FF F028h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**SMB2_PUE**	**SMB1_PUE**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-2	**Reserved.**
1	**SMB2_PUE (SMBus2 Pull-Up).** Enables the pull-up resistors of SMB2 pins. 0: SCL2, SDA2 are not pulled up (default). 1: SCL2, SDA2 are pulled up.
0	**SMB1_PUE (SMBus1 Pull-Up).** Enables the pull-up resistors of SMB1 pins. 0: SCL1, SDA1 are not pulled up (default). 1: SCL1, SDA1 are pulled up.

2.0 Signal/Pin Description and Configuration (Continued)

Device Pull-Up Enable 1 Register (DEVPU1)

DEVPU1 affects the pull-up resistors on the corresponding pins only if the functions described are selected. Otherwise, the bits of this register have no effect.

Location: FF F029h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**							**SWD_PUE**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-1	**Reserved.**
0	**SWD_PUE (SWD Pin Pull-Up Enable).** If the SensorPath function is selected, SWD_PUE enables the pull-up resistors on the corresponding pin. 0: SWD pin is not pulled up (default). 1: SWD pin is pulled up.

3.0 Power, Reset and Clocks

3.1 POWER

3.1.1 Power Planes

The WPC8763L has five power planes (wells), as shown in Table 5.

Table 5. WPC8763L Power Planes

Power Plane	Description	Power Plane Notation	Power Pins	Ground Pins
Host Domain	Powers the LPC interface and host modules (except for Host GPIO and MSWC) and some external signals.[1]	V_{DD}	VDD	GND
Core Domain	Powers the core domain modules, the Shared Memory, “Core Access to Host Modules”, MSWC, Host GPIO modules and their external signals.[1]	V_{CC}	VCC	GND
Internal	Powers the internal logic of all the device modules except those powered by AV_{CC}.	V_{CORF}[2]	VCORF	GND
Analog	Powers the ADC and DAC modules and respective external signals.[1] Also powers some MSWC registers, the 32.768 KHz clock/crystal oscillator signals and some other storage elements.	AV_{CC}	AVCC	AGND

1. See the tables in Section 2.4.1 on page 25 to Section 2.4.16 on page 31, specifically the Power Well column, for how the WPC8763L external interface signals are assigned to various power planes.
2. V_{CORF} is internally generated from V_{CC} by an on-chip power converter.

For correct WPC8763L operation, V_{CC} must be active when V_{DD} is applied. In addition, AV_{CC} must be applied at the same time that V_{CC} is applied. Protection is provided only against rise-time differences between the different power planes.

3.1.2 Power States

The WPC8763L has the following main power states:

- **Power Fail**
  Host domain, core domain and analog power planes are powered off (i.e., V_{DD}, V_{CC} and AV_{CC} are inactive).
- **Power Active**
  Core domain, internal and analog power planes are powered on; host domain power plane may be on or off (i.e., V_{CC}, V_{CORF} and AV_{CC} are active; V_{DD} may be active or inactive).
  Power Active state has several sub-states, depending on the domain:
  — The host domain has Host Power On and Host Power Off states according to the V_{DD} status.
  — The core domain and host-core interface have Active, Idle and Deep Idle states; see Section 4.18 on page 177.

The following power states are illegal (i.e., WPC8763L operation is not guaranteed):

- Host domain on and core domain off and/or Analog off (i.e., V_{DD} active and V_{CC} inactive and/or AV_{CC} inactive).
- Core domain on and analog off (i.e., V_{CC} active and AV_{CC} inactive) and vice-versa.

Table 6 summarizes the power states related to the WPC8763L power planes.

Table 6. WPC8763L Power States and Related Power Planes

Power State		Host Domain (V_{DD})	Core Domain (V_{CC})	Internal (V_{CORF})	Analog (AV_{CC})
Power Fail		–[1]	–	–	–
Power Active	Active, Idle and Deep Idle	x	+	+	+
	Host Power Off	–	+	+	+
	Host Power On	+	+	+	+
Illegal[2]		+	–	–	–
		x	+	+	–
		x	–	–	+

1. ‘–’ means Off; ‘+’ means On; ‘x’ means may be on or off.
2. Operation is not guaranteed.

3.0 Power, Reset and Clocks (Continued)

3.1.3 Power Connection and Layout Guidelines

The WPC8763L requires a power supply voltage of 3.0V–3.6V for the digital supplies (V_{CC} and V_{DD}) and 3.135V–3.465V for the analog power supply (AV_{CC}).

For correct operation, both AV_{CC} and V_{CC} should be applied. Although the WPC8763L is protected against damage if operated with only one supply (AV_{CC} or V_{CC}), the ADC and DAC modules do not function correctly in this case and might cause current leakage. Therefore, ADC and DAC modules should be left at their reset condition (disabled) or should be disabled by firmware until AV_{CC} is within the limits specified above. In addition, the AV_{CC} is internally isolated from V_{CC} to allow the AV_{CC} to be externally filtered or driven by a low-noise power supply.

All WPC8763L internal logic is powered by an on-chip power converter, which generates the V_{CORF} power supply from the V_{CC} supply. The V_{CORF} power plane is available at the VCORF pin to connect a decoupling capacitor.

V_{DD}, V_{CC}, and V_{CORF} use a common ground return, named digital ground and marked GND. The analog circuits and the 32.768 KHz oscillator, powered by AV_{CC}, use a separate ground return, named analog ground and marked AGND. This ensures effective isolation of the analog modules and the 32.768 KHz oscillator from noise caused by the digital modules.

The following directives are recommended for the WPC8763L power and ground connections (see Figure 2 for the power supply connections):

Ground Connection

Use two ground planes, one for digital signals (GND) and one for analog signals (AGND). Make the following ground connections:

- Connect the analog ground plane (AGND) to the digital ground plane (GND) at one point only. This point should be physically located close to the WPC8763L AGND pin.
- Connect the analog ground return pin of the WPC8763L (AGND) to the analog ground plane.
- Connect the decoupling capacitors of the analog supply (AV_{CC}) to the analog ground plane, as close as possible to the AGND pin.
- Connect the ground reference of the ADC module voltage input signals (V_{IN} in Figure 2) to the AGND plane, close to the WPC8763L.
- Connect the ground reference of the DAC module voltage output signals (V_{OUT} in Figure 2) to the AGND plane, close to the WPC8763L.
- Connect all GND pins of the WPC8763L to the GND plane.
- Locate the decoupling capacitors of the core domain digital supply (V_{CC}) pin close to each VCC-GND pin pair; connect one terminal of each capacitor to the ground plane.
- Locate the filter capacitor of the on-chip power converter (V_{CORF}) pin close to the WPC8763L; connect one of the capacitor terminals to the ground plane.
- Locate the decoupling capacitors of the host domain digital supply (V_{DD}) pin close to the VDD-GND pair; connect one terminal of each capacitor to the ground plane.

Note that low-impedance ground layers improve noise isolation and reduce ground bounce problems.

Power Connection

All WPC8763L supply pins must be connected to the appropriate power plane, and decoupling capacitors must be used as recommended below.

Connect the analog supply pin (AV_{CC}) to a low-noise, 3.3V power supply. If the AVCC pin is connected to the same power supply as the VCC pin, it is recommended to use an external L-C or R-C filter for the AVCC pin. Figure 2 shows an example of an L-C filter [L_1 and (C_1+C_2)].

Decoupling and Filter Capacitors

Use the following capacitors to reduce power supply glitches, ground bounce and EMI (see Figure 2 for the position of capacitors, e.g., C1, C2, etc.):

- Core domain digital supply (V_{CC}): Place one 0.1 µF capacitor (C3) as close as possible to **each** VCC-GND pin pair. Also, place one 10–47 µF C4 tantalum/ceramic capacitor on the common net, as close as possible to the chip.
- On-chip power converter (V_{CORF}): Place one 1 µF ceramic capacitor (C8) as close as possible to the VCORF pin.
- Host digital supply (V_{DD}): Place one 0.1 µF capacitor (C5) as close as possible to the VDD-GND pin pair. Also, place one 10-47 µF tantalum/ceramic capacitor (C6) on the common net, as close as possible to the chip.
- Analog supply (AV_{CC}): Place one 0.1 µF capacitor (C2) and one 10-47 µF tantalum/ceramic capacitor (C1) as close as possible to the AVCC pin.

3.0 Power, Reset and Clocks (Continued)

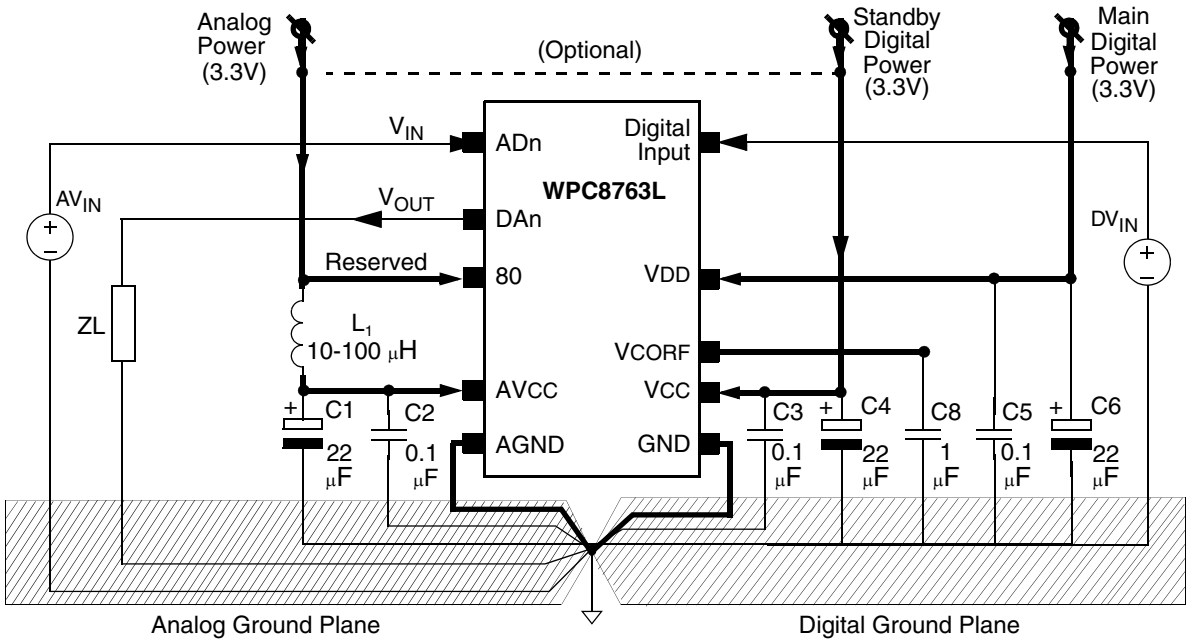


Figure 2. WPC8763L Power Supply Connection

3.2 RESET SOURCES AND TYPES

The WPC8763L has several input reset types. They are divided in two groups: resets for the core domain and resets for the host domain.

Reset Types for the Core Domain

- V_{CC} Power-Up reset
  Activated when the V_{CC} supply is powered up. Note that for correct operation of the chip, AV_{CC} must also be powered up.
- $\overline{\text{VCC\_POR}}$ Input reset
  Activated by a low level at the $\overline{\text{VCC\_POR}}$ pin except while the $\overline{\text{VCC\_POR}}$ signal is active, as a result of a V_{CC} Power-Up reset.
- Watchdog Cold reset
  Activated when a Watchdog Cold condition is detected by the TWD module; see Section 4.11 on page 133.
- Debugger Cold reset
  Activated via a SYSRST_ON (JTAG) command by the Debugger Interface module (see Section 4.20 on page 187) if DBGRST_MODE bit in RSTCTL register is set to 0; see page 37.
- Debugger Warm reset
  Activated also via a SYSRST_ON (JTAG) command, but when DBGRST_MODE bit in RSTCTL register is set to 1.
- Watchdog Warm reset
  Activated when a Watchdog Warm condition is detected by the TWD module; see Section 4.11 on page 133.

The **Core Domain reset** generic group covers the following resets: V_{CC} Power-Up reset, $\overline{\text{VCC\_POR}}$ Input reset, Watchdog Cold reset and Debugger Cold reset.

Reset Types for the Host Domain

- V_{DD} Power-Up reset
  Activated when the V_{DD} supply is powered up.
- Host Hardware reset
  Activated when $\overline{\text{LRESET}}$ input is asserted.

Unless otherwise noted, reset references throughout the WPC8763L modules default to the following types:

- For core domain functions and host-core interface functions: all reset types for the core domain, described above.
- For host domain functions: all the reset types for host domain described above.

The following sections detail the sources and effects of the various resets on the WPC8763L, per reset type.

3.0 Power, Reset and Clocks (Continued)

3.2.1 V_{CC} Power-Up Reset

V_{CC} Power-Up reset is generated by an internal circuit. The WPC8763L performs a V_{CC} Power-Up reset when V_{CC} power is detected as "good" (i.e., V_{CC} is above V_{CCON};see Section 8.1.5 on page 279). Note that AV_{CC} power must be applied together with V_{CC} power to the WPC8763L.

V_{CC} Power-Up reset timing is described in "Core Domain Reset" on page 288.

If the host processor accesses the WPC8763L during internal reset time, the result is:

- The host is stalled (by WPC8763L driving a "Long WAIT" SYNC response on the LPC bus) until after the reset process is completed (i.e., after the HOSTWAIT bit in SMC_CTL register is set to 0) and the transaction can be answered by WPC8763L.
- If HRAPU bit in MSWCTL3 register is set to 1, the host is reset by WPC8763L asserting $\overline{\text{KBRST}}$ until the internal reset is completed.

On V_{CC} Power-Up reset, the WPC8763L performs the following:

- Enables the 32.768 KHz crystal oscillator.
- Drives a low level at the $\overline{\text{VCC\_POR}}$ output.
- Loads default values to all host-controlled registers.
- Performs all actions done by a $\overline{\text{VCC\_POR}}$ Input reset.

3.2.2 $\overline{\text{VCC\_POR}}$ Input Reset

$\overline{\text{VCC\_POR}}$ Input reset is generated by an externally generated low level at the $\overline{\text{VCC\_POR}}$ pin. The occurrence of a $\overline{\text{VCC\_POR}}$ Input reset is indicated by the VCC_POR_STS bit set to 1 in RSTCTL register; see page 37.

$\overline{\text{VCC\_POR}}$ Input reset timing is described in "Core Domain Reset" on page 288. The WPC8763L response to a host processor access during the internal reset time is the same as for V_{CC} Power-Up reset.

On $\overline{\text{VCC\_POR}}$ Input reset, the WPC8763L performs the following:

- Puts pins with strap options into TRI-STATE mode and enables the internal pull-up/down resistors on the strap pins.
- Samples the values of the strap pins.
- Sets the High-Frequency Clock Generator (HFCG) to its default frequency.
- Resets the TAP controller of the Debugger Interface module.
- Loads default values to all the host-controlled registers retained by V_{CC}.
- Performs all actions done by a Watchdog Cold reset.

3.2.3 Watchdog Cold Reset and Debugger Cold Reset

Watchdog Cold reset is generated by the TWD module on detection of a Watchdog Cold condition. The occurrence of a Watchdog Cold reset is indicated by the WDRST_STS bit set to 1 in T0CSR register if WDRST_MODE bit in TWCFG register is set to 0; see Section 4.11.3 on page 135.

Debugger Cold reset is generated by the Debugger Interface module on receipt of a SYSRST_ON, JTAG command, if DBGRST_MODE bit in RSTCTL register is set to 0. The occurrence of a Debugger Cold reset is indicated by the DBGRST_STS bit set to 1 in RSTCTL register, if DBGRST_MODE bit in RSTCTL register is set to 0; see RSTCTL register on page 37.

The WPC8763L response to a host processor access during the internal reset time is the same as for V_{CC} Power-Up reset.

On Watchdog Cold reset or Debugger Cold reset, the WPC8763L performs the following:

- Loads default values to all core-controlled registers.
- Performs all the actions done by a Watchdog Warm reset or Debugger Warm reset.

3.2.4 Watchdog Warm Reset and Debugger Warm Reset

Watchdog Warm reset is generated by the TWD module on detection of a Watchdog Warm condition. The occurrence of a Watchdog Warm reset is indicated by the WDRST_STS bit set to 1 in T0CSR register, if WDRST_MODE bit in TWCFG register is set to 1; see Section 4.11.3 on page 135.

Debugger Warm reset is generated by the Debugger Interface module on receipt of a SYSRST_ON, JTAG command, if DBGRST_MODE bit in RSTCTL register is set to 1. The occurrence of a Debugger Warm reset is indicated by the DBGRST_STS bit set to 1 in RSTCTL register, if DBGRST_MODE bit in RSTCTL register is set to 1; see RSTCTL register on page 37.

On Watchdog Warm reset or Debugger Warm reset, the WPC8763L resets the core and the status registers in the Debugger Interface module. It also resets the PMC module to Active mode.

3.0 Power, Reset and Clocks (Continued)

3.2.5 V_{DD} Power-Up Reset

V_{DD} Power-Up reset is generated by an internal circuit. The WPC8763L performs a V_{DD} Power-Up reset when V_{CC} power is detected as "good" (i.e., V_{DD} is above V_{DDON}; see Section 8.1.5 on page 279). V_{DD} Power-Up reset timing is described in "Host Domain Reset" on page 289.

On V_{DD} Power-Up reset, the WPC8763L performs the following action:

- Performs all the actions done by a Host Hardware reset.

3.2.6 Host Hardware Reset

Host Hardware reset is generated by a low level at the $\overline{\text{LRESET}}$ pin. Host Hardware reset timing is described in "Host Domain Reset" on page 289. $\overline{\text{LRESET}}$ assertion generates a Host Hardware reset, regardless of the state of the $\overline{\text{LPCPD}}$ signal (i.e., the reset is generated also when the LPC interface is powered down).

Important Note: Every time a Core Domain reset is asserted, $\overline{\text{LRESET}}$ must be asserted too. $\overline{\text{LRESET}}$ must remain asserted until the internal reset (as specified in "Core Domain Reset" on page 288) is deasserted.
This requirement is inherently met if the V_{DD} power supply is controlled by the WPC8763L. Otherwise, after Core Domain reset is deasserted, $\overline{\text{LRESET}}$ must remain asserted for at least $t_{RSTDLY1}$ (see "Core Domain Reset" on page 288).

On Host Hardware reset, the WPC8763L performs the following:

- Loads default values to all the host-controlled registers retained by V_{DD}.
- Loads default values to the Host GPIO registers retained by V_{CC} if the respective VDDLOAD bit in their GPCFG2 configuration register is set to 1.

3.2.7 $\overline{\text{VCC\_POR}}$ Generation

The $\overline{\text{VCC\_POR}}$ signal (external V_{CC} Power-Up reset) is an open-drain I/O pin. It is internally pulled low when the WPC8763L powers up for a period of t_{PORW} (see "Core Domain Reset" on page 288) after V_{CC} rises above V_{CCON} level; see Section 8.1.5 on page 279. Then, the pin is floated. An external pull-up resistor is required to pull the $\overline{\text{VCC\_POR}}$ pin high.

$\overline{\text{VCC\_POR}}$ pin is also pulled low for t_{PORW} when a Watchdog Cold reset is generated by the TWD module, if the WDC2POR bit in TWCFG register (see Section 4.11.3 on page 135) is set to 1.

In addition, the $\overline{\text{VCC\_POR}}$ signal is used internally to reset the WPC8763L. Pulling this pin low, externally, resets the core and its modules; see Section 3.2.2 on page 49. The occurrence of a $\overline{\text{VCC\_POR}}$ Input reset is indicated by the VCC_POR_STS bit in RSTCTL register; see page 36 (System Configuration chapter).

3.3 CLOCK DOMAINS

The WPC8763L has three clock domains, as shown in Table 7.

Table 7. WPC8763L Clock Domains

Clock Domain	Frequency	Source	Usage
Core (CLK, FCLK, FMCLK)	See Section 4.19 on page 181	HFCG	Core domain modules, Shared Memory, "Core Access to Host Modules" and MSWC
LPC (LCLK)	0 or 33 MHz	LPC clock input	LPC Bus interface
32K (LFCLK)	32.768 KHz	Clock input or on-chip oscillator[1]	HFCG, TWD, PMC, MSWC

1. See Section 3.3.1.

Core Clock Domain

The core clock domain is sourced by the HFCG. On V_{CC} Power-Up reset and $\overline{\text{VCC\_POR}}$ Input reset, the HFCG is set to generate a core clock (CLK) frequency of 6.67 MHz. The core clock frequency, as well as the frequency of the other clock domain signals (FMCLK, FCLK), can be modified by the firmware; see Section 4.19 on page 181.

LPC Clock Domain

The LPC interface is driven by the LCLK input. LCLK frequency can be 0 or 33 MHz. The $\overline{\text{CLKRUN}}$ signal may be used as part of the power management mechanism that stops the LPC clock; see "CLKRUN Functionality" on page 253.

3.0 Power, Reset and Clocks (Continued)

32K Clock Domain

The 32K clock domain (LFCLK) is the reference for the generation of other clocks in the WPC8763L. LFCLK is used as a reference clock for the HFCG generator and as an “always-running” clock (LFCLK is running in Deep Idle) in TWD, PMC and MSWC.

3.3.1 32 KHz Clock Domain Generation

The WPC8763L uses a 32.768 KHz clock signal as the basic clock for the generation of the other internal clocks. The 32.768 KHz clock is generated either by the internal oscillator circuit or by an external oscillator.

Internal Oscillator

The internal oscillator employs an external crystal connected to the on-chip amplifier. The on-chip amplifier is accessible on the 32KX1 input pin and the 32KX2 output pin. See Figure 3 for the recommended external circuit and Table 8 for a listing of the circuit components. The oscillator is disabled if the AV_{CC} supply is below the minimum values specified in Section 8.1.1 on page 278.

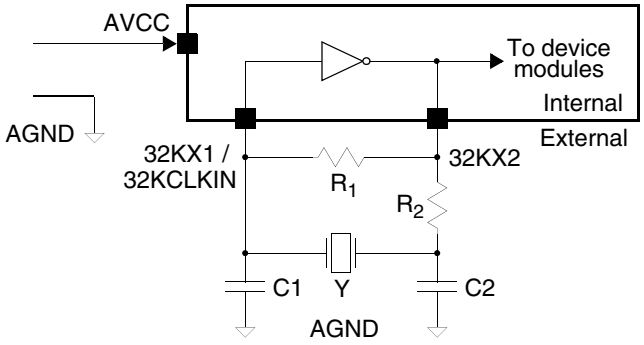


Figure 3. Recommended Oscillator External Circuitry

Table 8. Crystal Oscillator Circuit Components

Component	Parameters	Values	Tolerance
Crystal	Resonance Frequency	32.768 KHz Parallel Mode	User defined
	Type	N-Cut or XY-bar	
	Drive Level	0.5 µW	Max
	Serial Resistance	40 KΩ	Max
	Quality Factor, Q	35000	Min
	Shunt Capacitance	2 pF	Max
	Load Capacitance, C_L	9-13 pF	
	Temperature Coefficient	User defined	
Resistor R_1	Resistance	20 MΩ	5%
Resistor R_2	Resistance	33 KΩ	5%
Capacitor C_1	Capacitance	5.6 pF - 6.8 pF	10% - 20%
Capacitor C_2	Capacitance	5.6 pF - 6.8 pF	10% - 20%

External Elements

Choose C1 and C2 capacitors (see Figure 3) to match the crystal’s load capacitance. The load capacitance C_L “seen” by crystal Y is comprised of C_1 in series with C_2 and in parallel with the parasitic capacitance of the circuit. The parasitic capacitance is caused by the chip package, board layout and socket (if any), and can vary from 0 to 8 pF. The rule of thumb for choosing these capacitors is:

$$C_L = (C_1 * C_2) / (C_1 + C_2) + C_{PARASITIC}$$

3.0 Power, Reset and Clocks (Continued)

To provide an accurate 32.768 KHz frequency, the value of C_1 and C_2 capacitors may vary (within the range defined in Table 8) according to the specific board implementation. See the TBD Application Note for details on how to select the capacitors' value and how to lay out the PCB.

Oscillator Start-Up

The oscillator starts to generate 32.768 KHz pulses after approximately t_{32KVAL} from when AV_{CC} is turned on; see "Low-Frequency Clock Timing" on page 290.

C_1 can be trimmed to achieve precisely 32.768 KHz. For high accuracy, use crystal and capacitors with low tolerance and temperature coefficients.

External Oscillator

32.768 KHz can be applied from an external clock source, as shown in Figure 4.

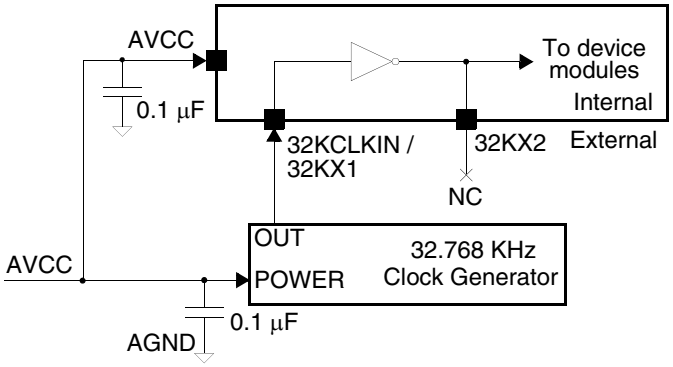


Figure 4. External Oscillator Connections

Connections

Connect the clock to the 32KCLKIN pin, leaving the oscillator output, 32KX2, unconnected.

Signal Parameters

The signal levels should conform to the voltage level requirements for 32KCLKIN/32KX1 stated in Section 8.2.3 on page 280. The signal should have a duty cycle of approximately 50%.

3.0 Power, Reset and Clocks (Continued)

3.4 TESTABILITY SUPPORT

The WPC8763L supports two testing techniques:

- In-Circuit Testing (ICT)
- XOR-Tree Testing

Table 9 shows the strap values required to select each Test mode.

Table 9. Test Mode Selection

Test Mode	$\overline{\text{TRIS}}$	BADDR1	BADDR0
No Test mode selected	1	Other than 00	
XOR Tree	1	0	0
ICT	0	1	1
Reserved exclusively for Winbond use	0	Other than 11	

3.4.1 ICT

The In-Circuit Testing (ICT) technique, also known as “bed-of-nails”, injects logic patterns to the input pins of the devices mounted on the tested board. It then checks their outputs for the correct logic levels.

The WPC8763L supports this testing technique by floating (TRI-STATE mode) all the device pins. This prevents back-driving the WPC8763L pins by the ICT tester when a device normally controlled by WPC8763L is tested (device inputs are driven by the ICT tester).

To enter TRI-STATE mode, the $\overline{\text{TRIS}}$ pin must be pulled down (by a 10 KΩ resistor to GND), the BADDR0 and BADDR1 pins must be left unconnected (they are internally pulled to V_{CC}). The straps are sampled by a V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset. When the internal reset (see “Core Domain Reset” on page 288) is completed, all the device output and I/O pins are floated (TRI-STATE mode). Exceptions to this are: power supply pins (AVCC, VCC, VDD, VCORF, Reserved pin 80, AGND, GND), the ADC reference pin (VREF), the $\overline{\text{VCC\_POR}}$ pin and the 32KX2 pin, which do not float in TRI-STATE mode.

3.4.2 XOR-Tree Testing

The WPC8763L device mounted on the board can be tested using the XOR-Tree technique. This test also checks the correct connection of the device pins to the board.

To enter XOR-Tree mode, BADDR0 and BADDR1 pins must be pulled down (by a 10 KΩ resistor to GND) and the $\overline{\text{TRIS}}$ pin must be left unconnected (it is internally pulled to V_{CC}). The straps are sampled by a V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset. When the internal reset (see “Core Domain Reset” on page 288) is completed, the device pins (including BADDR0-1 and $\overline{\text{TRIS}}$ pins) are connected in a XOR-Tree configuration and are isolated from the internal WPC8763L functions.

In XOR-Tree mode, all WPC8763L device pins included in the XOR-Tree chain are configured as inputs except the last pin in the tree, which is the XOR_OUT output. The buffer type of the input pins included in the XOR tree is the type of the default signal for the respective pin, except for the following pins, which have IN_{PCI} buffer type: $\overline{\text{LDRQ}}$/GPIO24/HGPIO01, $\overline{\text{CLKRUN}}$/GPIO11/HGPIO02, $\overline{\text{LPCPD}}$/GPIO10/HGPIO00.

The input pins are chained through XOR gates, as shown in Figure 5. The power supply pins (AVCC, VCC, VDD, VCORF, Reserved pin 80, AGND, GND), ADC input pins (AD0-AD3), ADC reference pin (VREF), DAC output pins (DA0-DA1), GPI96-97 pins, $\overline{\text{VCC\_POR}}$ pin and crystal oscillator pins (32KX1, 32KX2) are excluded from the XOR tree. In XOR-Tree mode, all pins in the XOR tree have their internal pull-down resistors enabled, with the following exceptions:

- KBSIN0-KBSIN7 pins, which have internal pull-up resistors enabled.
- LCLK, $\overline{\text{LRESET}}$, $\overline{\text{LFRAME}}$ pins, which are floating (i.e., they do not have an internal pulling resistor).

The enabling of the internal pulling resistors allows the pins that are not used to be left unconnected during XOR-Tree testing without affecting XOR-Tree functionality.

The XOR-tree chain starts with pin 36 (see Figure 5), continues with pins 37 (the next pin in ascending order) through 128, goes to pin 1, and ends with XOR_OUT pin (35).

3.0 Power, Reset and Clocks (Continued)

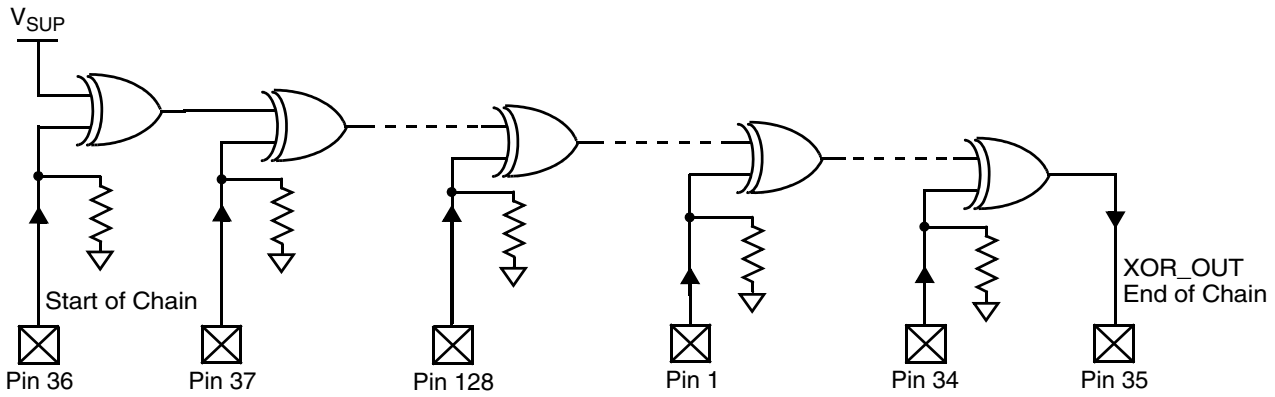


Figure 5. XOR-Tree Chain (Simplified Diagram)

The maximum propagation delay through the XOR tree, from the start of the chain to the end of the chain (XOR_OUT), is 125 ns (not tested; based on design simulation).

4.0 Embedded Controller Modules

4.1 FLASH INTERFACE UNIT (FIU)

4.1.1 Overview

The FIU interfaces between the core bus and the external flash memory. It supports SPI interface flash devices with up to 4 Mbytes of memory. The FIU maps the flash memory into the core address space and provides a write protection mechanism.

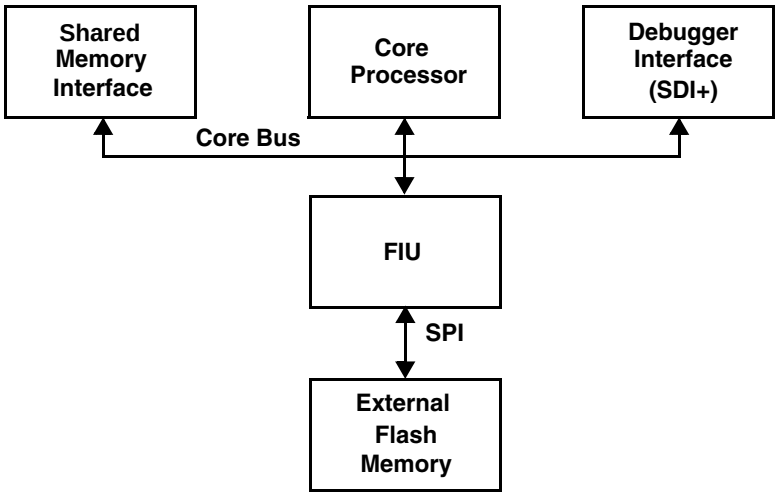


Figure 6. FIU System Block Diagram

Features:

- SPI-Based External Memory
  - Interfaces with SPI memory up to 32 Mbits
  - Interfaces with one SPI device
  - Configurable, maximum block size:
    - Read:1 byte or 16 bytes
    - Write:1 byte or 4 bytes
  - Page programing support
  - Configurable SPI interface clock rate: 8 MHz to 50 MHz
  - Configurable SPI Fast Read mode with fast clock rate up to 50 MHz
- Memory Access Protection
  - Three configurable memory write windows
  - Protects the lowest 128 Kbytes of flash memory (Core Boot Block area)

4.1.2 FIU Core Bus Interface

The transactions on the core bus are initiated by one of the following:

- Core
- Shared Memory module
- Debugger interface (SDI+ module)

Core Bus Transactions. The size of a core-to-flash write transaction should not exceed the burst length, as configured by W_BURST field; otherwise, the transaction is ignored. During the execution of a write access, the flash interface is busy and therefore stalls any new flash read or write transaction. During a read, the core bus may use a byte, word or double-word read transaction regardless of the burst size value configured by R_BURST field in BURST_CFG register.

Unsupported Flash Address. The FIU supports flash devices ranging from 128 Kbytes to 4 Mbytes in steps of 128 Kbytes, the configuration of the flash size is done via FL_SIZE_P1 field in FIU_CFG register. In the WPC8763L the size of the core address space allocated for the flash memory is 4 Mbytes. If the flash device is smaller than 4 Mbytes, only part of the flash allocated memory space is in use (i.e., supported). The FIU ignores any access to unsupported addresses and, if IAD_EN bit in RESP_CFG register is set to 1, an IAD trap is generated by the FIU.

Flash Base Address. See the memory map in Table 1 on page 19.

4.0 Embedded Controller Modules (Continued)

4.1.3 SPI Flash Interface

Flash Configuration. The FIU supports SPI flash devices with a total size ranging from 128 Kbytes to 4 Mbytes. Total flash size can be any multiple of 128 Kbytes within this range. FL_SIZE_P1 field in FIU_CFG register and DEV_SIZE field in SPI_FL_CFG register control the flash size as follows:

— The device must be connected to $\overline{\text{F\_CS0}}$.

— The total flash size, defined by FL_SIZE_P1, must be ≤ the device size, defined by DEV_SIZE.

After reset, the Booter, running from internal ROM, detects the actual size of the flash device(s). To do this, the Booter reads the size of the flash from the Header area, located at the bottom of the flash, and updates FL_SIZE_P1 field in FIU_CFG register, FL_DEV bit and DEV_SIZE field in SPI_FL_CFG register.

Note: If the flash is not programed, the Booter must receive the flash size from the Flash Loader via the debugger interface.

Address Translation. The SPI Flash Interface block translates the core address to an SPI address, as shown in Figure 7.

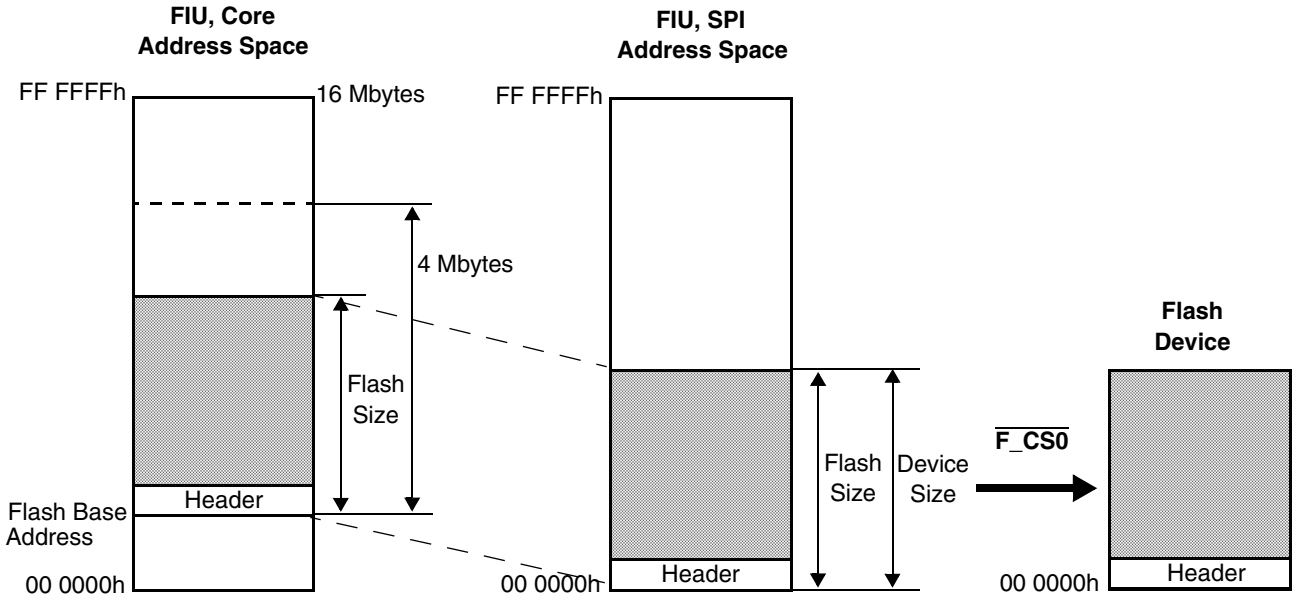


Figure 7. Address Translation

Flash Transactions. The SPI burst length is limited according to the number of bytes defined either in R_BURST or W_BURST fields in BURST_CFG register, or by the number of bytes in the read/write core transaction, if the latter is shorter. To enhance performance it is recommended to set R_BURST so that a minimum number of separate transactions will be generated on the FIU interface. To do this, set R_BURST and W_BURST to the maximum read/write burst length supported by the flash device.

Read Access. Normal read transaction is selected when the F_READ bit in SPI_FL_CFG register is set to 0. Fast read is selected when the F_READ bit is set to 1.

User Mode Access (UMA). UMA allows the execution of special SPI transactions, such as write control, status read, sector/block/chip erase and chip ID read, in addition to the standard SPI transactions (write, read and fast read). It provides flash access by using an indirect address/data mechanism. In this mode no protection is supported.

UMA contains the following registers:

- Transaction Code register (UMA_CODE): contains the command code (i.e., for flash device commands) for the SPI transaction.
- Address registers (UMA_AB2-0): hold address bits 23-0 of the SPI transaction (UMA_AB2 holds the MSByte).
- Data registers (UMA_DB3-0): hold up to four data bytes of the SPI transaction; the data byte in UMA_DB0 holds the LSB.
- Control/Status register (UMA_CTS): contains various control, configuration and status fields.
- Extended Control/Status register (UMA_ECTS): contains the software control bits for flash device selection.

UMA read transactions ignore the setting of F_READ bit in SPI_FL_CFG register. Instead, a UMA fast read is performed when the UMA_CODE register value is 0Bh and RD_WR bit in UMA_CTS register is 0.

4.0 Embedded Controller Modules (Continued)

UMA write transactions may contain more than four data bytes. To perform extended transactions, the first UMA write transaction must contain the valid code, address bytes and optional data bytes; the next transactions must contain only data bytes (UMA_CODE and UMA_AB2-0 registers are also interpreted as data bytes). These data bytes are sent via F_SDO in the following order: UMA_CODE, UMA_AB2, UMA_AB1, UMA_AB0, UMA_DB0, UMA_DB1, UMA_DB2, UMA_DB3. The SW_CS0 bit in UMA_ECTS register must be set to 0 before the first transaction and it must be returned to 1 after the last transaction. Only write transactions can be performed using this feature; performing read transactions might cause unpredictable behavior.

Flash transactions should not be performed concurrently with UMA transactions. Therefore, flash transactions initiated by the core (such as instruction fetch), Shared Memory or Debugger Interface, must be disabled before the execution of a UMA transaction and can be re-enabled after the UMA transaction is completed.

4.1.4 Access Protection

Access to a Write-Protected Flash Address. The FIU provides two protection mechanisms against inadvertent flash programing (write access to the flash):

- Core Firmware Boot Block (CFBB) protection
- Bulk flash protection

To provide protection for the CFBB, the lower 128 Kbytes of the flash is divided into eight 16-Kbyte blocks; see Figure 8 on page 57. Each block is protected by the respective bit (CFB_P7-0) in CFBB_PROT register.

Bulk flash protection is based on the following:

- The 4 Mbytes allocated for the flash is write protected by default.
- Write access to the flash is allowed only within a programmable write window.

The FIU supports up to three independent write windows. Each write window has programmable low and high limits (see Figure 8 on page 57), which are programed via the FWINn_LOW and FWINn_HIGH registers (where n=1, 2 or 3). Any write window may be set to any address on a boundary of 4 Kbytes, and its size may vary from 4 Kbytes to 4 Mbytes. When either WIN_LOW or WIN_HIGH is set to 000h, the respective write window is disabled.

To write into any flash address, the address must be within at least one write window; otherwise, the FIU ignores the write access. Also, in the CFBB area (the lower 128 Kbytes of the flash address space), a write access must be within an enabled write window and the protection bit for the corresponding CFBB block must be set to 0 (i.e., the block is not protected).

When NMI_EN bit in RESP_CFG register is set to 1, write access to a protected address generates an NMI to the ICU and sets (1) the PROT_ACC status bit in NMISTAT register.

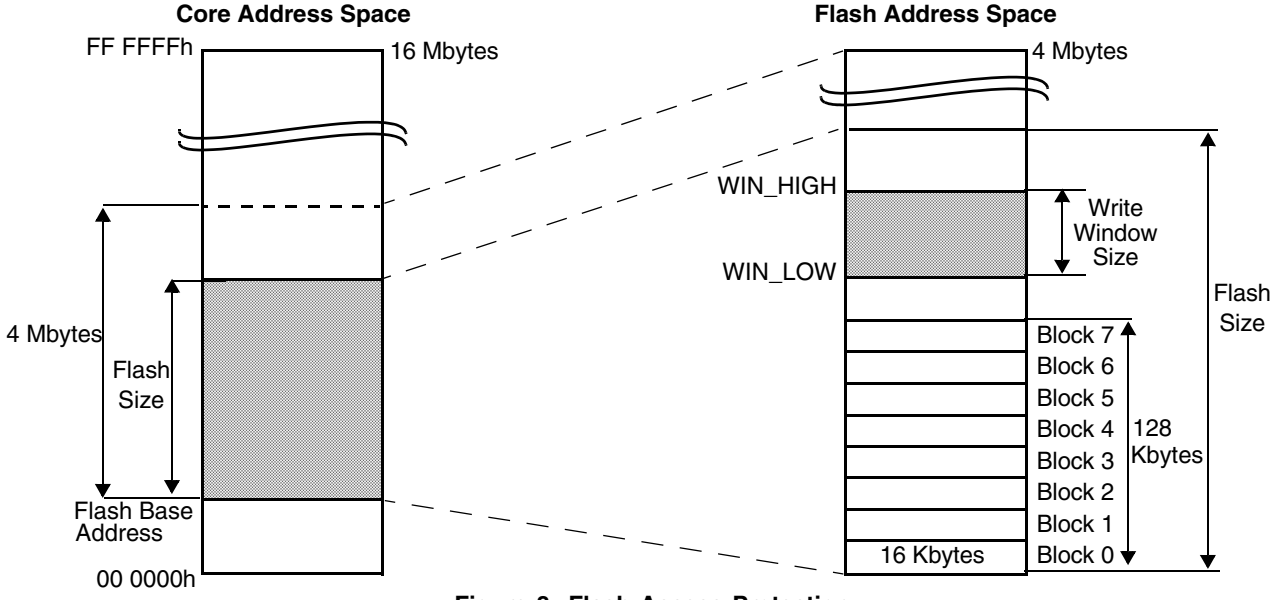


Figure 8. Flash Access Protection

4.0 Embedded Controller Modules (Continued)

Access Protection Locking. To prevent inadvertent changes of the protection settings, a lock mechanism is implemented, as follows:

- FL_SIZE_P1 field in the FIU_CFG register is protected by setting FL_PRM_LK bit in PROT_LOCK register to 1.
- CFB_P7-0 bits in CFBB_PROT register are locked when they are set to 1.
- Each set of FWINn_LOW and FWINn_HIGH registers is locked by setting the respective FWINn_LK bit in PROT_LOCK register to 1.
- All the lock bits above are cleared either by Core Domain reset or by writing an “unlock” data sequence to the PROT_CLEAR register; see “PROT_CLEAR Register” on page 62.

4.1.5 Low Power Support

The FIU supports low power operation by controlling the clock signal generated by the flash interface.

The F_SCK signal toggles only during transactions (i.e., it is low when the bus is idle).

4.1.6 FIU Registers

For a summary of the abbreviations used for Register Type, see “Register Abbreviations and Access” on page 20.

Register Map

All FIU registers are powered by V_{CC} and are reset by Core Domain reset.

Location	Mnemonic	Register Name	Size	Type
FF 0000h	FIU_CFG	FIU Configuration	Byte	R/W or RO
FF 0001h	BURST_CFG	Burst Configuration	Byte	R/W
FF 0002h	RESP_CFG	FIU Response Configuration	Byte	R/W
FF 0003h	CFBB_PROT	Core Firmware Boot Block Protection	Byte	R/W1S
FF 0004h	FWIN1_LOW	Flash Access Window 1, Low Limit	Word	R/W or RO
FF 0006h	FWIN1_HIGH	Flash Access Window 1, High Limit	Word	R/W or RO
FF 0008h	FWIN2_LOW	Flash Access Window 2, Low Limit	Word	R/W or RO
FF 000Ah	FWIN2_HIGH	Flash Access Window 2, High Limit	Word	R/W or RO
FF 000Ch	FWIN3_LOW	Flash Access Window 3, Low Limit	Word	R/W or RO
FF 000Eh	FWIN3_HIGH	Flash Access Window 3, High Limit	Word	R/W or RO
FF 0010h	PROT_LOCK	Protection Lock	Byte	R/W1S
FF 0011h	PROT_CLEAR	Protection and Lock Clear	Byte	WO
FF 0012h-FF 0013h	Reserved			
FF 0014h	SPI_FL_CFG	SPI Flash Configuration	Byte	R/W or RO
FF 0015h	Reserved			
FF 0016h	UMA_CODE	UMA Code Byte	Byte	R/W
FF 0017h	UMA_AB0	UMA Address Byte 0	Byte	R/W
FF 0018h	UMA_AB1	UMA Address Byte 1	Byte	R/W
FF 0019h	UMA_AB2	UMA Address Byte 2	Byte	R/W
FF 001Ah	UMA_DB0	UMA Data Byte 0	Byte	R/W
FF 001Bh	UMA_DB1	UMA Data Byte 1	Byte	R/W
FF 001Ch	UMA_DB2	UMA Data Byte 2	Byte	R/W
FF 001Dh	UMA_DB3	UMA Data Byte 3	Byte	R/W
FF 001Eh	UMA_CTS	UMA Control and Status	Byte	R/W
FF 001Fh	UMA_ECTS	UMA Extended Control and Status	Byte	R/W

4.0 Embedded Controller Modules (Continued)

FIU Configuration Register (FIU_CFG)

Location: FF 0000h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**FL_SIZE_P1**					
Reset	0	0	1	0	0	0	0	1

Bit	Type	Description
7-6		**Reserved.**
5-0	R/W or RO	**FL_SIZE_P1 (Flash Size Plus 1).** Selects the size of the connected flash. Updated by the Booter from the flash after reset. FL_SIZE_P1 = (Number of 128 Kbyte blocks contained by the flash) + 1. Valid values for FL_SIZE_P1 range from 02h to 21h (default); all other values are reserved.

Burst Configuration Register (BURST_CFG)

Location: FF 0001h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**W_BURST**		**Reserved**		**R_BURST**	
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-6		**Reserved.**
5-4	R/W	**W_BURST (Flash Write Burst).** Sets the maximum length of a legal write burst. This value must be set according to the specification of the flash connected to the WPC8763L. E.g., if a core bus word transaction is initiated while W_BURST = 0 or if a double-word transaction is initiated while W_BURST < 2, the FIU ignores the transaction. The value of this field is updated by the Booter from the flash after reset. **Bits** **5 4 Write Burst Size** 0 0: 1 byte (default). 1 0: 4 bytes. Other: Reserved.
3-2		**Reserved.**
1-0	R/W	**R_BURST (Flash Read Burst).** Sets the maximum length of a legal read burst. This value must be set according to the specification of the flash connected to the WPC8763L.The value of this field is updated by the Booter from the flash after reset. **Bits** **1 0 Read Burst Size** 0 0: 1 byte (default). 1 1: 16 bytes. Other: Reserved.

4.0 Embedded Controller Modules (Continued)

FIU Response Configuration Register (RESP_CFG)

Location: FF 0002h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**NMI_EN**	**IAD_EN**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-2		**Reserved.**
1	R/W	**NMI_EN (NMI Generation Enable).** When set to 1, core bus access to a write-protected flash address generates an NMI to the ICU. 0: Disabled (default). 1: Enabled.
0	R/W	**IAD_EN (IAD Trap Generation Enable).** When set to 1, core bus access to unsupported flash address generates an IAD Trap to the core. 0: Disabled (default). 1: Enabled.

Core Firmware Boot Block Protection Register (CFBB_PROT)

CFBB_PROT protects the lower 128 Kbytes of flash address space (CFBB). It divides the address range into eight 16 Kbyte memory blocks; see Figure 8 on page 57. Bit 0 protects the first memory block, bit 1 protects the next memory block, and so on.

Location: FF 0003h

Type: R/W1S

Bit	7	6	5	4	3	2	1	0
Name	**CFB_P7**	**CFB_P6**	**CFB_P5**	**CFB_P4**	**CFB_P3**	**CFB_P2**	**CFB_P1**	**CFB_P0**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W1S	**CFB_P7-0 (Core Firmware Boot Block 7-0 Protect).** Each bit write protects the respective CFBB block. Once set to 1, those bits can be cleared to 0 by reset or by writing the open lock sequence to the PROT_CLEAR register; see page 62. 0: Writing into the respective CFBB block is enabled (default). 1: Writing into the respective CFBB is disabled.

4.0 Embedded Controller Modules (Continued)

Flash Access Window 3-1, Low Limit Registers (FWIN3-1_LOW)

FWIN3-1_LOW selects the lower limit of flash access Windows 3-1. It can be set anywhere within the area allocated for the flash in the core address space; see Figure 8 on page 57.

Location: FWIN1_LOW: FF 0004h; FWIN2_LOW: FF 0008h; FWIN3_LOW: FF 000Ch

Type: R/W or RO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Res.**	**WIN_LOW**											**Reserved**			
Reset **FWIN2-1_LOW**	0	0	0	0	0	0	**0**	0	0	0	0	0	0	0	0	0
Reset **FWIN3_LOW**	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15		**Reserved.**
14-4	R/W or RO	**WIN_LOW (Window Low Limit).** Selects the low limit address of the write access window. Setting either WIN_LOW or WIN_HIGH field to 000h disables the window. WIN_LOW = (Lower limit for window address) / 4 Kbytes. 000h: Window is disabled (default). 020h - 420h: Valid values for WIN_LOW. Other: Reserved.
3-0		**Reserved.**

Flash Access Window 3-1, High Limit Register (FWIN3-1_HIGH)

FWIN3-1_HIGH selects the upper limit of flash access Windows 3-1. It can be set anywhere within the area allocated for the flash in the core address space; see Figure 8 on page 57.

Location: FWIN1_HIGH: FF 0006h; FWIN2_HIGH: FF 000Ah; FWIN3_HIGH: FF 000Eh

Type: R/W or RO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Res.**	**WIN_HIGH**											**Reserved**			
Reset **FWIN2-1_HIGH**	0	0	0	0	0	0	**0**	0	0	0	0	0	0	0	0	0
Reset **FWIN3_HIGH**	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15		**Reserved.**
14-4	R/W or RO	**WIN_HIGH (Window High Limit).** Selects the high limit address of the write access window. Setting either WIN_LOW or WIN_HIGH field to 000h disables the window. WIN_HIGH = (Upper limit for window address) / 4 Kbytes + 1. 000h: Window is disabled (default). 020h - 420h: Valid values for WIN_HIGH (see Note). Other: Reserved. Note: The value of WIN_HIGH must be higher than the value of WIN_LOW; otherwise, the device might cause unpredictable results.
3-0		**Reserved.**

4.0 Embedded Controller Modules (Continued)

Protection Lock Register (PROT_LOCK)

Location: FF 0010h

Type: R/W1S

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**FL_PRM_LK**	**Reserved**	**FWIN3_LK**	**FWIN2_LK**	**FWIN1_LK**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-5		**Reserved.**
4	R/W1S	**FL_PRM_LK (Flash Parameter Lock).** When set to 1, locks itself and the following: FL_SIZE_P1 field in FIU_CFG register and all bits of SPI_FL_CFG register. Once set to 1, this bit can be cleared to 0 by reset or by writing the open lock sequence to the PROT_CLEAR register; see page 62. 0: Flash parameters can be changed (default). 1: Flash parameters and FL_PRM_LK bit are locked.
3		**Reserved.**
2	R/W1S	**FWIN3_LK (Flash Access Window 3 Limits, Lock).** When set to 1, locks itself and the contents of the FWIN3_LOW and FWIN3_HIGH registers. Once set to 1, it can be cleared to 0 by reset or by writing the open lock sequence to the PROT_CLEAR register; see page 62. 0: FWIN3_LOW and FWIN3_HIGH registers are R/W (default). 1: FWIN3_LOW and FWIN3_HIGH registers and FWIN3_LK bit are read-only.
1	R/W1S	**FWIN2_LK (Flash Access Window 2 Limits, Lock).** Same as FWIN3_LK bit for FWIN2_LOW and FWIN2_HIGH registers.
0	R/W1S	**FWIN1_LK (Flash Access Window 1 Limits, Lock).** Same as FWIN3_LK bit for FWIN1_LOW and FWIN1_HIGH registers.

Protection and Lock Clear Register (PROT_CLEAR)

Location: FF 0011h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**Open Lock Sequence (see description)**							
Reset	X	X	X	X	X	X	X	X

Bit	Type	Description
7-0	WO	**Open Lock Sequence.** Writing a data sequence to this register clears (00h) the contents of CFBB_PROT and PROT_LOCK registers. Open lock sequence: 87h, 61h, 63h. A different data value or a out-of-sequence value aborts the whole sequence. Read returns unpredictable results.

4.0 Embedded Controller Modules (Continued)

SPI Flash Configuration Register (SPI_FL_CFG)

Location: FF 0014h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**F_READ**	**DEV_SIZE**					
Reset	0	0	1	0	0	0	0	0

Bit	Type	Description
6	R/W or RO	**F_READ (Fast Read Select).** Selects the SPI read access type (normal or fast). F_READ is updated by the Booter from the flash after reset. 0: Normal read (default). 1: Fast read.
5-0	R/W or RO	**DEV_SIZE (Flash Device Size).** Sets the size of the flash device connected to the SPI bus. DEV_SIZE is updated by the Booter from the flash after reset. **Bits 5 4 3 2 1 0 — Device Size** 0 0 0 0 0 1: 128 Kbytes. 0 0 0 0 1 0: 256 Kbytes. 0 0 0 1 0 0: 512 Kbytes. 0 0 1 0 0 0: 1 Mbyte. 0 1 0 0 0 0: 2 Mbytes. 1 0 0 0 0 0: 4 Mbytes (default). Other: Reserved; if used, causes unpredictable behavior.

UMA Code Byte Register (UMA_CODE)

Location: FF 0016h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Transaction Code**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Transaction Code.** Holds the transaction code byte used in an UMA transaction to the flash; see "User Mode Access (UMA)." on page 56).

UMA Address Byte 2-0 Register (UMA_AB2-0)

UMA_AB0: Holds address bits 7-0 used in a UMA transaction to the flash (Address Byte 0).
UMA_AB1: Holds address bits 15-8 used in a UMA transaction to the flash (Address Byte 1).
UMA_AB2: Holds address bits 23-16 used in a UMA transaction to the flash (Address Byte 2).

Location: UMA_AB0: FF 0017h; UMA_AB1: FF 0018h; UMA_AB2: FF 0019h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Address Byte 2-0**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Address Byte 2-0.** Holds the respective address bits used in an UMA transaction to the flash.

4.0 Embedded Controller Modules (Continued)

UMA Data Byte 3-0 Register (UMA_DB3-0)

UMA_DB0: Holds data byte 0 used in an UMA transaction to the flash (Data Byte 0).
UMA_DB1: Holds data byte 1 used in an UMA transaction to the flash (Data Byte 1).
UMA_DB2: Holds data byte 2 used in an UMA transaction to the flash (Data Byte 2).
UMA_DB3: Holds data byte 3 used in an UMA transaction to the flash (Data Byte 3).

Location: UMA_DB0: FF 001Ah; UMA_DB1: FF 001Bh; UMA_DB2: FF 001Ch; UMA_DB3: FF 001Dh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Data Byte 3-0**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Data Byte 3-0.** Holds the respective data byte used in an UMA transaction to the flash.

UMA Control and Status Register (UMA_CTS)

Location: FF 001Eh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**EXEC_ DONE**	**Reserved**	**RD_WR**	**Reserved**	**A_SIZE**	**D_SIZE**		
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W1S	**EXEC_DONE (Operation Execute/Done).** Writing 1 triggers a UMA flash transaction. Writing 0 is ignored. This bit is set to 0 on Core Domain reset or at the end of a UMA transaction. The bit remains set (high) while any UMA transaction is in progress. Note:Modifying the contents of UMA_CTS, UMA_CODE, UMA_AB2-0, UMA_DB3-0 registers while EXEC_DONE is set to 1 causes unpredictable results. 0: Writing 0 is ignored; reading 0 indicates that a UMA transaction is not in progress (default). 1: Writing 1 triggers a UMA flash transaction; reading 1 indicates that a UMA transaction is in progress.
6		**Reserved.**
5	R/W	**RD_WR (Read/Write Select).** Selects the type (read or write) of the UMA flash transaction to be executed. 0: Read - flash data is read into UMA_DB3-0 registers; the first byte goes into UMA_DB0. (default). 1: Write - the flash is written with data from UMA_DB3-0 registers; the first byte comes from UMA_DB0.
3	R/W	**A_SIZE (Address Field Size Select).** Selects the size of the address field (0 or 3 bytes) of the UMA flash transaction to be executed. 0: No address field used (default). 1: Three-byte address field from UMA_AB2-0 registers.
2-0	R/W	**D_SIZE (Data Field Size Select).** Selects the size of the data field (0 to 4 bytes) of the UMA flash transaction to be executed. **Bits** **2 1 0 Data Field Size** 0 0 0: No data field (default). 0 0 1: 1 byte data field to/from UMA_DB0 register. 0 1 0: 2 byte data field to/from UMA_DB1-0 register. 0 1 1: 3 byte data field to/from UMA_DB2-0 register. 1 0 0: 4 byte data field to/from UMA_DB3-0 register. Others: Reserved.

4.0 Embedded Controller Modules (Continued)

UMA Extended Control and Status Register (UMA_ECTS)

Location: FF 001Fh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**Reserved (must be 1)**	**SW_CS0**
Reset	0	0	0	0	0	0	1	1

Bit	Type	Description
7-2		**Reserved.**

4.1.7 Usage Hints

1. After reset, the flash interface signals are as follows:
   - F_SCK drives low level.
   - F_SDO drives an undefined level (either low or high).
   - F_SDI input is ignored.
   - $\overline{\text{F\_CS0}}$ drives high level.
2. To initiate UMA transactions:
   1. Configure the required UMA address, data and code in the respective registers.
   2. Set the correct transaction parameters in the UMA_CTS register.
   3. Initiate a UMA transaction by writing 1 to the EXEC_DONE bit.
4. When UMA transactions are used, direct flash access (i.e., non-UMA: core code fetch, core data access, shared memory access and Debugger Interface access) is not allowed. Therefore, the following must be performed:
   1. Stall any host access to shared BIOS memory in the flash, using STALL_HOST bit in FLASH_SIZE register (see page 225).
   2. Execute core firmware from the internal RAM.
   3. Initiate the UMA transaction.
   4. During UMA transaction execution, limit data access by either the core firmware, or the Debugger Interface, to internal RAM and WPC8763L registers only.
   5. After EXEC_DONE bit becomes 0, return to normal operation.

4.0 Embedded Controller Modules (Continued)

4.2 MULTI-INPUT WAKE-UP (MIWU)

The Multi-Input Wake-Up Unit (MIWU) allows the WPC8763L to exit Idle and Deep Idle modes. In addition, it provides signal conditioning and grouping of external interrupt sources. It supports a total of 40 internal and/or external wake-up sources.

4.2.1 Features

- Supports up to 40 internal and/or external wake-up inputs
- Generates a wake-up signal
- Generates interrupt signals for:
  - — each of the 40 inputs
  - — one interrupt for each group of eight inputs, e.g., I/O ports
- User-selectable trigger condition on each input:
  - — positive edge
  - — negative edge
- Individual enable and pending bits for each input
- Programmable bypass mode connects inputs to ICU without MIWU involvement

4.2.2 Operation

Overview

The MIWU detects a valid software-selectable trigger condition on any of its inputs. On detection of a valid trigger event, the MIWU generates a wake-up request and/or an interrupt request. The wake-up request is connected to the Power Management module (PMC) and may be utilized to exit Idle and Deep Idle modes and return to Active mode. The interrupt requests are used to signal to the Interrupt Control Unit (ICU) that an edge-triggered (either external or internal) interrupt condition occurred. Figure 9 shows a block diagram of the Multi-Input Wake-Up module.

The MIWU is active while in Idle and Deep Idle modes. Note, however, that because all device clocks are stopped in these modes, the detection of a trigger event on an input and the resulting setting of the pending flag are not synchronous to the core clock.

Table 10 lists the MIWU sources and interrupts used in the WPC8763L.

Table 10. Input Assignments

Source		Destination	
Name	**MIWU Input**	**MIWU Output**	**Interrupt Name**
PSCLK1	WUI10	WKINTA	PSWUINT (INT7)
PSCLK2	WUI11		
PSCLK3	WUI12		
PSDAT1	WUI13		
PSDAT2	WUI14		
PSDAT3	WUI15		
Reserved	WUI16		
Reserved	WUI17		
Reserved	WUI20	WKO20	INT1
GPIO01	WUI21	WKO21	INT30
Reserved	WUI22	WKO22	INT22
GPIO03	WUI23	WKO23	INT8
GPIO04	WUI24	WKO24	INT20
GPIO05	WUI25	WKO25	INT19
MSWC Wake-Up	WUI26	WKO26	INT16
T0OUT[1]	WUI27	WKO27	INT31

4.0 Embedded Controller Modules (Continued)

Table 10. Input Assignments (Continued)

Source		Destination	
Name	**MIWU Input**	**MIWU Output**	**Interrupt Name**
KBSIN0	WUI30	WKINTC	KBSINT (INT15)
KBSIN1	WUI31		
KBSIN2	WUI32		
KBSIN3	WUI33		
KBSIN4	WUI34		
KBSIN5	WUI35		
KBSIN6	WUI36		
KBSIN7	WUI37		
GPIO06	WUI40	WKINTD	MIWU2 (INT7)
SensorPath Event[1]	WUI41		
Reserved	WUI42		
SMBus1 Wake-Up[1]	WUI43		
SMBus2 Wake-Up[1]	WUI44		
GPIO07	WUI45		
GPIO10 or GPI90[2]	WUI46		
GPIO11 or GPI91[2]	WUI47		
GPIO12 or GPI92[2]	WUI50	WKINTE	MIWU3 (INT11)
GPIO13 or GPI93[2]	WUI51		
GPIO14 or GPI94[2]	WUI52		
GPI95	WUI53		
GPIO16 or GPI96[2]	WUI54		
GPI97	WUI55		
Host Access Wake-Up[3]	WUI56		
$\overline{\text{LRESET}}$ Active Wake-Up[4]	WUI57		

1. Configure WUIxx for rising edge detection of the input event.
2. Selected by GES1 configuration register (see page 43).
3. The wake-up input is triggered by host access to one of the logical devices, listed in "Core Interrupt" on page 253. Configure the WUI56 for rising edge detection.
4. Set WKEDG57 bit to 0 (see WKEDG5 register on page 70) to generate a wake-up event on a high-to-low transition of $\overline{\text{LRESET}}$; set WKEDG57 bit to 1 to generate a wake-up event on a low-to-high transition of $\overline{\text{LRESET}}$ (see also LPCRSTA bit in MSWCTL1 register, on page 241).

Trigger Conditions

Through the WKEDGx registers, the trigger condition on the selected input signal can be selected as either positive edge (low-to-high transition) or negative edge (high-to-low transition).

Pending Flags

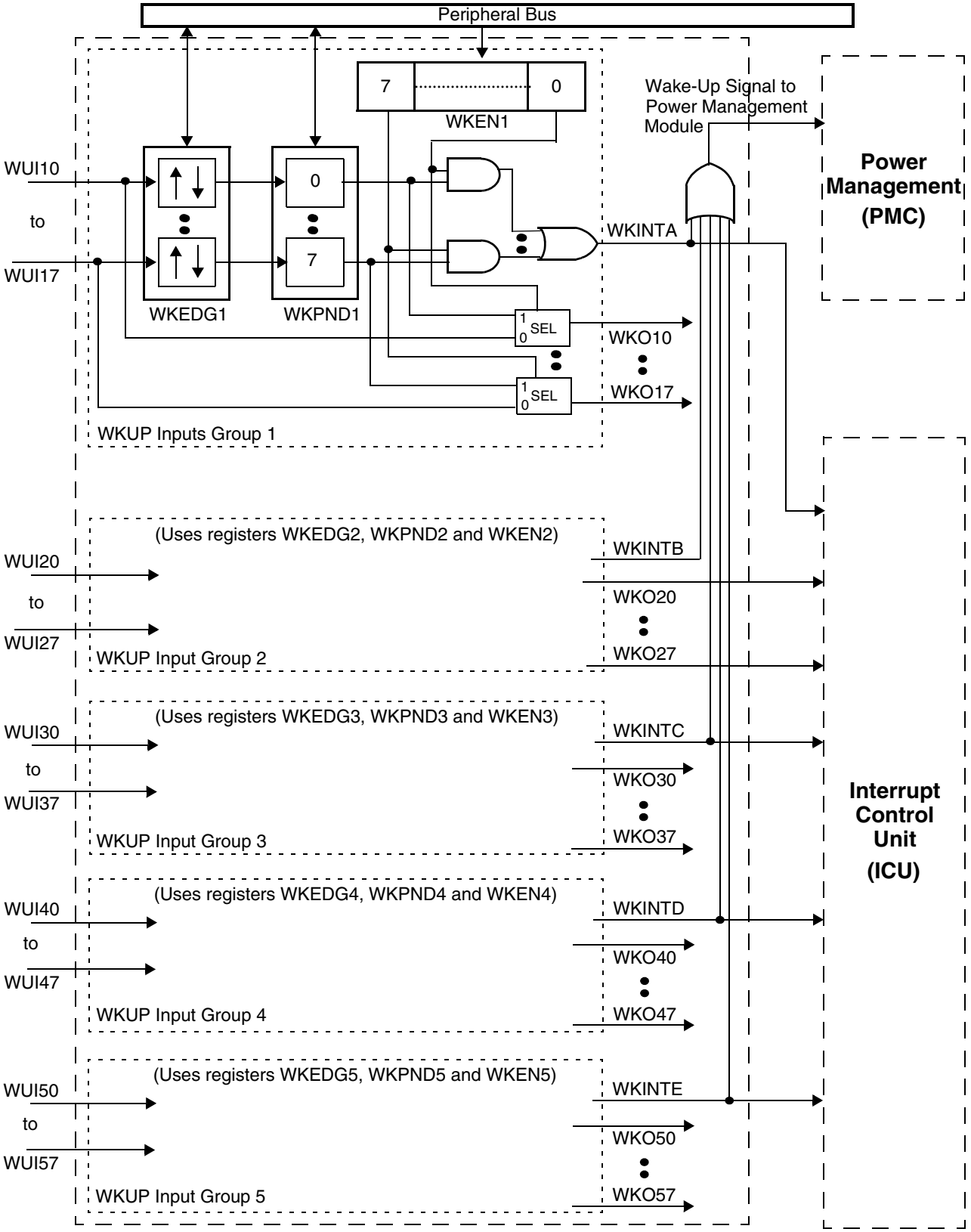
An trigger event at the MIWU input sets to 1 the respective pending bit in WKPNDx register, for the corresponding input signal.

Since the WKPNDx register holds a pending wake-up condition until it is cleared, the device does not enter Idle or Deep Idle mode if any wake-up bit is both enabled and pending (set to 1). Consequently, the pending flags must be cleared before attempting to enter Idle and Deep Idle modes.

Input Enable

The MIWU handles multiple wake-up signals. Set the appropriate bits in the WKENx registers to select which wake-up signal causes the device to exit Idle and Deep Idle modes.

4.0 Embedded Controller Modules (Continued)



Note: Not all MIWU-ICU connections are implemented; for details, see ICU interrupt assignments in Table 10 on page 66.

Figure 9. Multi-Input Wake-Up, Generic Block Diagram

4.0 Embedded Controller Modules (Continued)

Interrupts

The combined output of all pending and enabled channels of the MIWU generates the wake-up signal, which is fed into both the Power Management Control module (PMC) and the Interrupt Control Unit (ICU). Therefore, each wake-up of the device can be followed by a wake-up interrupt. Since the device cannot enter Idle and Deep Idle modes without having the core execute a “WAIT” instruction, a wake-up interrupt is needed to terminate the “WAIT” instruction on wake-up.

MIWU outputs WKO10–WKO57 are connected to the Interrupt Control Unit (ICU) to generate an interrupt associated with the specific MIWU output. The WKOxx output behaves as follows:

- When the respective WKENxx bit is cleared, the WUIxx is connected to the ICU directly (bypassing the edge detectors and pending bits). The ICU can be configured to use the signal as a level or edge triggered interrupt.
- When the respective WKENxx bit is enabled, the output of the pending bit, WKPDxx, is connected to WKOxx.s

Note: To enable and disable ICU interrupts generated by their associated MIWU WKOxx output, use the ICU mask register in the ICU.

In addition, the MIWU provides five interrupt request lines: WKINTA, WKINTB, WKINTC, WKINTD and WKINTE; see Figure 9 on page 68. These are routed to the ICU (except WKINTB) and can request an interrupt if a valid trigger event occurred at any of the **enabled** input sources within the group of eight inputs associated with the interrupt line. For a detailed summary of the interrupts available, see Table 10 on page 66.

4.2.3 MIWU Registers

For a summary of the abbreviations used for Register Type, see “Register Abbreviations and Access” on page 20.

MIWU Register Map

All MIWU registers are powered by V_{CC} and are reset by Core Domain reset.

Location	Mnemonic	Register Name	Size	Type
FF F1C0h	WKEDG1	Edge Detection Register	Byte	R/W
FF F1C2h	WKEDG2	Edge Detection Register	Byte	R/W
FF F1C4h	WKEDG3	Edge Detection Register	Byte	R/W
FF F1C6h	WKEDG4	Edge Detection Register	Byte	R/W
FF F1C8h	WKEDG5	Edge Detection Register	Byte	R/W
FF F1CAh	WKPND1	Pending Register	Byte	R/W1S
FF F1CCh	WKPCL1	Pending Clear Register	Byte	WO
FF F1CEh	WKPND2	Pending Register	Byte	R/W1S
FF F1D0h	WKPCL2	Pending Clear Register	Byte	WO
FF F1D2h	WKPND3	Pending Register	Byte	R/W1S
FF F1D4h	WKPCL3	Pending Clear Register	Byte	WO
FF F1D6h	WKPND4	Pending Register	Byte	R/W1S
FF F1D8h	WKPCL4	Pending Clear Register	Byte	WO
FF F1DAh	WKPND5	Pending Register	Byte	R/W1S
FF F1DCh	WKPCL5	Pending Clear Register	Byte	WO
FF F1DEh	WKEN1	Enable Register	Byte	R/W
FF F1E0h	WKEN2	Enable Register	Byte	R/W
FF F1E2h	WKEN3	Enable Register	Byte	R/W
FF F1E4h	WKEN4	Enable Register	Byte	R/W
FF F1E6h	WKEN5	Enable Register	Byte	R/W

4.0 Embedded Controller Modules (Continued)

Edge Detection Register (WKEDG1)

WKEDG1 configures the trigger condition of the input signals WUI10 to WUI17. The register is cleared on reset; this configures all associated input signals to be triggered on a rising edge.

Location: FF F1C0h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WKED17-WKED10**							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**Edge Selection.** For inputs WUI10 through WUI17. Each bit is associated with one of eight inputs. 0: Low-to-High transition (default). 1: High-to-Low transition.

Edge Detection Register (WKEDG2)

WKEDG2 configures the trigger condition of the input signals WUI20 to WUI27. The functionality of this register is identical to the WKEDG1 register described above.

Location: FF F1C2h

Type: R/W

Edge Detection Register (WKEDG3)

WKEDG3 configures the trigger condition of the input signals WUI30 to WUI37. The functionality of this register is identical to the WKEDG1 register described above.

Location: FF F1C4h

Type: R/W

Edge Detection Register (WKEDG4)

WKEDG4 configures the trigger condition of the input signals WUI40 to WUI47. The functionality of this register is identical to the WKEDG1 register described above.

Location: FF F1C6h

Type: R/W

Edge Detection Register (WKEDG5)

WKEDG5 configures the trigger condition of the input signals WUI50 to WU57. The functionality of this register is identical to the WKEDG1 register described above.

Location: FF F1C8h

Type: R/W

Pending Register (WKPND1)

WKPND1 latches the selected trigger event associated with input signals WUI10 to WUI17. On reset, the value of WKPND1 register is undefined.

Note: Software can only set the register bits; only the WKPCL1 register can clear them. Writing 0 to any of the bits leaves its value unchanged. The WKPND1 register format is shown below:

Location: FF F1CAh

Type: R/W1S

Bit	7	6	5	4	3	2	1	0
Name	**WKPD17-WKPD10**							
Reset	X	X	X	X	X	X	X	X

Bit	Description
7-0	**Wake-Up Pending.** A set bit (1), indicates that a valid trigger event occurred on the associated input.

4.0 Embedded Controller Modules (Continued)

Pending Register (WKPND2)

WKPND2 latches the selected trigger event associated with the input signals WUI20 to WUI27. For a detailed description of the register, see the description of the WKPND1 register, above.

Location: FF F1CEh

Type: R/W1S

Pending Register (WKPND3)

WKPND3 latches the selected trigger event associated with the input signals WUI30 to WUI37. For a detailed description of the register, see the description of the WKPND1 register, above.

Location: FF F1D2h

Type: R/W1S

Pending Register (WKPND4)

WKPND4 latches the selected trigger event associated with the input signals WUI40 to WUI47. For a detailed description of the register, see the description of the WKPND1 register, above.

Location: FF F1D6h

Type: R/W1S

Pending Register (WKPND5)

WKPND5 latches the selected trigger event associated with the input signals WUI50 to WU57. For a detailed description of the register, see the description of the WKPND1 register, above.

Location: FF F1DAh

Type: R/W1S

Enable Register (WKEN1)

WKEN1 enables the wake-up function of the associated input signal, WUI10 to WUI17. On reset, WKEN1 is cleared (0); this disables the associated input signals. The WKEN1 register format is shown below:

Location: FF F1DEh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WKEN17-WKEN10**							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**Wake-Up Enable.** If the bit is set (1), a valid trigger event on the associated input generates a wake-up signal or interrupt request.

Enable Register (WKEN2)

WKEN2 enables the wake-up function of the associated input signal, WUI20 to WUI27. For a detailed description of the register, see the description of the WKEN1 register, above.

Location: FF F1E0h

Type: R/W

Enable Register (WKEN3)

WKEN3 enables the wake-up function of the associated input signal, WUI30 to WUI37. For a detailed description of the register, see the description of the WKEN1 register, above.

Location: FF F1E2h

Type: R/W

4.0 Embedded Controller Modules (Continued)

Enable Register (WKEN4)

WKEN4 enables the wake-up function of the associated input signal, WUI40 to WUI47. For a detailed description of the register, see the description of the WKEN1 register, above.

Location: FF F1E4h

Type: R/W

Enable Register (WKEN5)

WKEN5 enables the wake-up function of the associated input signal, WUI50 to WU57. For a detailed description of the register, see the description of the WKEN1 register, above.

Location: FF F1E6h

Type: R/W

Pending Clear Register (WKPCL1)

WKPCL1 controls the clearing (0) of the pending bits associated with the WUI10 through WUI17 inputs. This prevents potential hardware/software collisions during read-modify-write. The WKPCL1 register format is shown below:

Location: FF F1CCh

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**WKCL17-WKCL10**							

Bit	Description
7-0	**Clear Pending Flag.** If 1 is written to a bit, the associated pending flag located in WKPND1 is cleared (0). Writing 0 to a bit leaves the value of the corresponding pending flag unchanged.

Pending Clear Register (WKPCL2)

WKPCL2 controls the clearing (0) of the pending bits associated with the WUI20 through WUI27 inputs. For a detailed description of the register, see the description of the WKPCL1 register, above.

Location: FF F1D0h

Type: WO

Pending Clear Register (WKPCL3)

WKPCL3 controls the clearing (0) of the pending bits associated with the WUI30 through WUI37 inputs. For a detailed description of the register, see the description of the WKPCL1 register, above.

Location: FF F1D4h

Type: WO

Pending Clear Register (WKPCL4)

WKPCL4 controls the clearing (0) of the pending bits associated with the WUI40 through WUI47 inputs. For a detailed description of the register, see the description of the WKPCL1 register, above.

Location: FF F1D8h

Type: WO

Pending Clear Register (WKPCL5)

WKPCL5 controls the clearing (0) of the pending bits associated with the WUI50 through WUI57 inputs. For a detailed description of the register, see the description of the WKPCL1 register, above.

Location: FF F1DCh

Type: WO

4.0 Embedded Controller Modules (Continued)

4.2.4 Usage Hints

1. On reset, the WKEDGx registers are configured to select positive edge detection for all wake-up inputs. To change the edge detection of an input signal while preventing the false triggering of a wake-up/interrupt condition:
   a. Clear the WKENxx bit associated with the WUIxx input to disable that input.
   b. Write to the WKEDGx register to select the new type of edge detection for the specific input.
   c. Clear the WKPNDxx bit associated with the WUIxx input.
   d. Set the WKENxx bit associated with the WUIxx input to re-enable it.
2. After a reset (in this case, the wake-up inputs are left floating, producing unknown data at the MIWU input signals), and to prevent the false triggering of a wake-up condition, perform the following sequence:
   a. If the input originates from an GPIO pin, configure the pin as input.
   b. Clear the WKENx register, or if a WKOxx interrupt is used, disable the interrupt via the ICU.
   c. Write the WKEDGx register to select the desired type of edge detection for each of the pins used.
   d. Clear the WKPNDx register to cancel any pending bits.
   e. Either set the WKENxx bits associated with the pins to be used (thus enabling them for the wake-up/interrupt function) or re-enable the interrupt via the ICU.
3. To prevent spurious wake-up events from GPIO pins, which are either powered by V_{DD} or configured for “Enable by V_{DD}”, follow these guidelines:
   a. Before V_{DD} is turned off, disable both wake-up and interrupt from the GPIO pin by setting the relevant bit of WKEN2 and WKEN4-5 registers to 0.
   Note that for GPIO01, GPIO03-05, the interrupt is disabled separately by setting the relevant bit of IENAM0-1 registers to 0.
   b. After V_{DD} power is on:
      i. Clear the pending wake-up event by writing 1 to the relevant bit of WKPCL2 and WKPCL4-5 registers.
      ii. Enable wake-up and interrupt from the GPIO pin by setting the relevant bit of WKEN2 and WKEN4-5 registers to 1.

   Note that for GPIO01, GPIO03-05, the interrupt is enabled separately by setting the relevant bit of IENAM0-1 registers to 1.

4.0 Embedded Controller Modules (Continued)

4.3 INTERRUPT CONTROL UNIT (ICU)

The ICU has 31 channels. It interfaces between the different modules' interrupt requests and external interrupt requests and also generates the core interrupt. It generates both maskable and non-maskable interrupts. The ICU has a predetermined mechanism that allocates priority.

4.3.1 Features

Non-Maskable Interrupts (NMI)

- Gathers all edge-triggered non-maskable interrupt sources
  - External GPIO04 interrupt source
  - Access to write protected flash addresses
- Holds the status of the current pending NMI requests
- Generates non-maskable interrupt (NMI) to the core

Maskable Interrupts

- 31 active-high level or edge-triggered interrupt sources
- Core vectored interrupt mode
- Fixed priority among interrupt sources
- Individual enable/disable for each interrupt source
- Polling support by an Interrupt Status register
- Clear registers for edge-triggered interrupts

4.3.2 Non-Maskable Interrupt (NMI)

The Interrupt Control Unit (ICU) gathers internal and external non-maskable interrupt (NMI) sources and generates an NMI interrupt to the core when required.

External NMI

The ICU processes the GPIO04 signal to assert the core NMI input.

The GPIO04 signal is an asynchronous input with Schmitt trigger characteristics and an internal synchronization circuit; therefore, no external synchronizing circuit is needed. The GPIO04 signal generates an NMI on its falling edge.

Internal NMI

The ICU processes an access to write protected flash addresses (see Section 4.1 on page 55) to assert the core NMI input.

Non-Maskable Interrupt Processing

The core performs an "Interrupt Acknowledge" bus cycle when it starts to process a non-maskable interrupt.

After reset, NMI is disabled and must remain disabled until the software initializes the interrupt table, interrupt base and interrupt mode.

Internal NMI generation is always enabled in the ICU module. Therefore, NMI generation that is caused by an access to write protected flash addresses can be disabled by NMI_EN bit in RESP_CFG register; see page 60.

The external GPIO04 interrupt is enabled by setting ENLCK bit and remains enabled until a reset occurs. This allows the external NMI feature to be enabled only after the Interrupt Base Register (IMASK) and the Interrupt Stack Pointer (ISP), both in the core, have been set up.

Alternatively, the external GPIO04 interrupt can be enabled by setting EN bit, which remains enabled until an interrupt event or a reset occurs. This is useful when the pin toggles frequently but nested NMI interrupts are not desired. In such cases, the EN bit must be re-enabled before exiting the interrupt service routine.

To prevent a spurious NMI, set either EN bit or ENLCK bit to 1 only after GPIO04 input has been configured for operation.

The NMISTAT register holds the status of the current pending NMI requests. When one of the bits in NMISTAT is set to 1, an NMI request to the core is issued. The NMI handler should read the NMI Status register (NMISTAT) to decide which request to respond to. NMISTAT is cleared each time its contents are read, allowing new NMI events to occur.

Note that more than one of the bits in NMISTAT register might be set to 1. In addition, if a second NMI occurs while an NMI is in process, the second NMI might read NMISTAT and clear it; therefore, the first NMI reads a value of 0 from NMISTAT. For proper operation, the NMI handler must read NMISTAT. If more than one bit is set to 1, the NMI handler must handle all the sources.

4.0 Embedded Controller Modules (Continued)

4.3.3 Maskable Interrupts

The ICU receives level or edge-triggered interrupt request signals from 31 sources and generates a vectored interrupt to the core, when required. Priority among the interrupt sources (named INT1 to INT31) is fixed. Each interrupt source can be individually enabled or disabled by software control through:

- ICU interrupt enable registers
- Interrupt enable bits in the modules that request the interrupts

Pending interrupts, enabled or disabled, can be polled using the Status registers. The core supports INT0, but the ICU reserves INT0 so that it is not connected to any interrupt source.

Maskable Interrupt Vectors

Interrupt vector numbers are always positive and are in the range 10h to 2Fh. The IVCT register contains the interrupt vector of the enabled and pending interrupt with the highest priority. Interrupt vector 10h corresponds to INT0 with the lowest priority; vector 2Fh corresponds to INT31 with the highest priority.

The core performs an “Interrupt Acknowledge” bus cycle on receiving an enabled maskable interrupt request from the ICU. During the interrupt acknowledge cycle, a byte is read from address FF FE00h (IVCT register). The byte read is used as an index in the Dispatch Table to determine the address of the interrupt handler.

Although INT0 is not connected to any interrupt source, the IVCT register can return the value 10h. This happens, for example, when the interrupt request is removed before the interrupt acknowledge cycle. The entry in the Dispatch Table should point to a default interrupt handler that handles this error condition.

Pending Interrupts

Edge-triggered interrupts are latched by the Interrupt Status register. A pending edge-triggered interrupt is cleared by writing a 1 to the corresponding bit in one of the Edge Interrupt Clear registers (IECLR0 or IECLR1).

A pending level-triggered interrupt is cleared when the interrupt source is not active; note that the interrupt should be cleared in the device/module that drives it by clearing the event status bit.

Interrupt mask bits (IENAM register bits) and pending interrupt bits (ISTAT register bits) should be cleared to 0 only when interrupts are disabled; i.e., when bits I and/or E in PSR register (a core register) are 0. Bits in IENAM may be set at any time.

Interrupt Priorities

The priorities of the maskable interrupts are hard-wired and thus fixed. The interrupts are named INT0 to INT31, where INT0 has the lowest priority and INT31 has the highest priority.

External Interrupt Inputs

When a MIWU input is disabled and the respective WKOxx output of the MIWU is connected to the ICU, the MIWU input is fed directly to the ICU. In this case, the interrupt inputs are asynchronous. They are recognized by the WPC8763L during cycles in which the input setup and hold time requirements are satisfied.

Interrupt Assignment

Table 11 shows the mapping of the ICU maskable interrupts to different functions. For information on mask bits and the clear mechanism for the status bit in internal level interrupts, refer to descriptions of the module that is the interrupt source.

Table 11. ICU Interrupt Assignments

INT Number	Source	Type	Details	Priority
INT0	-	-	Error condition occurred (spurious interrupt)	Lowest
INT1	Reserved	Level-High	Reserved	
INT2	Internal	Level-High	Host I/F Keyboard/Mouse channel Output Buffer Empty	
INT3	Internal	Level-High	Host I/F Power Management Channel 1 **or** Channel 2 Output Buffer Empty	
INT4	Reserved	Level-High	Reserved	
INT5	Reserved	Level-High	Reserved	
INT6	Internal	Level-High	Core Bus Watcher interrupt	
INT7	Internal and External/MIWU	Level-High	MIWU PSWUINT or MIWU2	

4.0 Embedded Controller Modules (Continued)

Table 11. ICU Interrupt Assignments (Continued)

INT Number	Source	Type	Details	Priority
INT8	External/MIWU	Level-High	External Interrupt GPIO03 through the MIWU	
INT9	Internal	Level-High	MFT16-1 Interrupt (MFT16_INT1 **or** MFT16_INT2)	
INT10	Internal	Level-High	ADC interrupt (ADCI)	
INT11	External/MIWU	Level-High	MIWU MIWU3	
INT12	Reserved	Level-High	Reserved	
INT13	Internal	Level-High	SMB1 module interrupt	
INT14	Internal	Level-High	SMB2 module interrupt	
INT15	External/MIWU	Level-High	Internal Keyboard Scan Interrupt (KBSINT from MIWU)	
INT16	Internal/MIWU	Level-High	MSWC interrupt through the MIWU	
INT17	Reserved	Level-High	Reserved	
INT18	Internal	Level-High	CR_UART (Tx Interrupt **or** Rx Interrupt)	
INT19	External/MIWU	Level-High	External Interrupt GPIO05 through the MIWU	
INT20	External/MIWU	Level-High	External Interrupt GPIO04 through the MIWU	
INT21	Internal	Level High	PS/2 shift mechanism (PSINT1)	
INT22	Reserved	Level-High	Reserved	
INT23	Internal	Level-High	MFT16-2 Interrupt (MFT16_INT1 **or** MFT16_INT2)	
INT24	Internal	Level-High	Shared Memory Interrupt	
INT25	Internal	Level-High	Host I/F Keyboard/Mouse channel Input Buffer Full	
INT26	Internal	Level-High	Host I/F Power Management Channel 1 **or** Channel 2 Input Buffer Full	
INT27	Reserved	Level-High	Reserved	
INT28	Reserved	Level-High	Reserved	
INT29	Reserved	Level-High	Reserved	
INT30	External/MIWU	Level-High	External Interrupt GPIO01 through the MIWU	
INT31	Internal/MIWU	Edge Rising	TWD system tick (T0OUT), through the MIWU	Highest

4.0 Embedded Controller Modules (Continued)

4.3.4 ICU Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

ICU Register Map

All ICU registers are powered by V_{CC} and are reset by Core Domain reset.

Location	Mnemonic	Register Name	Size	Type
FF FE00h	IVCT	Interrupt Vector	Byte	RO
FF FE02h	NMISTAT	NMI Status	Byte	ROC
FF FE04h	EXNMI	External NMI Interrupt Control and Status	Byte	Varies per bit
FF FE0Ah	ISTAT0	Interrupt Status 0	Word	RO
FF FE0Ch	ISTAT1	Interrupt Status 1	Word	RO
FF FE0Eh	IENAM0	Interrupt Enable and Mask 0	Word	R/W
FF FE10h	IENAM1	Interrupt Enable and Mask 1	Word	R/W
FF FE12h	IECLR0	Edge Interrupt Clear 0	Word	WO
FF FE14h	IECLR1	Edge Interrupt Clear 1	Word	WO

Interrupt Vector Register (IVCT)

IVCT holds the vector number of the interrupt vector. It is set to 10h on reset.

Location: FF FE00h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**0**	**0**	**INTVECT**					
Reset	0	0	0	1	0	0	0	0

Bit	Description
7-6	These bits return 0 when read.
5-0	**INTVECT (Interrupt Vector).** Contains the encoded value of the enabled pending interrupt with the highest priority; the valid values range from 10h to 2Fh. Valid during an interrupt acknowledge core bus cycle in which IVCT is read. It may contain invalid data when INTVECT is updated.

NMI Status Register (NMISTAT)

NMISTAT holds the status of the current pending Non-Maskable Interrupt (NMI) request. It is cleared on reset and each time its contents are read. See the description of the EXNMI register below for additional details.

Location: FF FE02h

Type: ROC

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**PROT_ACC**	**EXT**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-2	**Reserved.**
1	**PROT_ACC (Flash Write Protected Access Non-Maskable Interrupt Request).** A PROT_ACC non-maskable interrupt request is caused by a write to protected flash addresses; see Section 4.1 on page 55. 0: No PROT_ACC non-maskable interrupt request occurred. 1: PROT_ACC non-maskable interrupt request occurred.

4.0 Embedded Controller Modules (Continued)

Bit	Description
0	**EXT (External Non-Maskable Interrupt Request).** An external non-maskable interrupt is caused (EXT bit is set to 1) by a falling edge of the signal input at the GPIO04 pin. 0: No external non-maskable interrupt request occurred (default). 1: External non-maskable interrupt request occurred.

External NMI Interrupt Control and Status Register (EXNMI)

EXNMI holds the current value of the GPIO04 signal (PIN bit) and controls the NMI interrupt generation based on a falling edge of the GPIO04 signal. EN and ENLCK are cleared on reset. When writing to this register, all reserved bits must be written with 0 for the device to function properly.

Location: FF FE04h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**ENLCK**	**PIN**	**EN**
Reset	x	x	x	x	0	0	x	0

Bit	Type	Description
7-3		**Reserved.**
2	R/W1S	**ENLCK (EXNMI Interrupt Enable Lock).** When set to 1, the external NMI feature is enabled and locked; it is cleared by Core Domain reset only (i.e., it cannot be cleared by software). After setting this bit, an NMI interrupt is generated every time the GPIO04 signal changes its value from high to low. Note that when ENLCK is set, EN bit is ignored. 0: External NMI feature disabled (default). 1: External NMI feature enabled and locked.
1	RO	**PIN (EXNMI Pin Value).** Contains the current (non-inverted) GPIO04 signal value. Data written to it is ignored.
0	R/W	**EN (EXNMI Interrupt Enable).** If this bit is set to 1, an NMI interrupt is generated when the GPIO04 signal changes its value from high to low. Cleared by hardware on reset and when an interrupt occurs (i.e., when EXT bit in NMISTAT register is set). It can be set and cleared by software; however, software can set this bit only when EXT is cleared. This bit is ignored when ENLCK is set. 0: External NMI interrupt disabled (default). 1: External NMI interrupt enabled.

Interrupt Status Register 0 (ISTAT0)

ISTAT0 indicates which maskable interrupts are pending regardless of the state of the corresponding IENA bits. It is cleared on reset.

Location: FF FE0Ah

Type: RO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IST15-0**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-0	**IST15-0 (Interrupt Status).** Each bit indicates if an interrupt event was sent to the ICU; IST15 to IST0 correspond to INT15 to INT0, respectively. Since INT0 is not used, IST0 always reads 0. Each bit is encoded as follows: 0: Interrupt input to ICU not pending (default). 1: Interrupt input to ICU pending.

4.0 Embedded Controller Modules (Continued)

Interrupt Status Register 1 (ISTAT1)

ISTAT1 indicates which maskable interrupts are pending regardless of the state of the corresponding IENA bits.

Location: FF FE0Ch

Type: RO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IST31-16**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-0	**IST31-16 (Interrupt Status).** Each bit indicates if an interrupt event was sent to the ICU. IST31 to IST16 correspond to INT31 to INT16, respectively. Each bit is encoded as follows: 0: Interrupt input to ICU not pending (default). 1: Interrupt input to ICU pending.

Interrupt Enable and Mask Register 0 (IENAM0)

IENAM0 controls the enable/disable of the maskable interrupt sources INT0 to INT15. It is cleared (0000h) on reset.

Location: FF FE0Eh

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IENA15-0**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-0	**IENA15-0 (Interrupt Enable).** Each bit enables or disables the corresponding interrupt request INT0 to INT15; e.g., IENA15 controls INT15. Since INT0 is not used, IENA0 has no effect on the operation of the ICU. 0: Interrupt disabled (default). 1: Interrupt enabled.

Interrupt Enable and Mask Register 1 (IENAM1)

IENAM1 controls the enable/disable of the maskable interrupt sources INT16 to INT31. It is cleared (0000h) on reset.

Location: FF FE10h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IENA31-16**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-0	**IENA31-16 (Interrupt Enable).** Each bit enables or disables the corresponding interrupt request INT16 to INT31; e.g., IENA16 controls INT16. 0: Interrupt disabled (default). 1: Interrupt enabled.

4.0 Embedded Controller Modules (Continued)

Edge Interrupt Clear Register 0 (IECLR0)

IECLR is used to clear pending, edge-triggered interrupts.

Location: FF FE12h

Type: WO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IEC15-1**															**Res.**

Bit	Description
15-1	**IEC15-1 (Edge Interrupt Clear).** Each bit clears the corresponding bit in ISTAT0 register. Writing to the bit positions of level-triggered interrupts has no effect. Read always returns FFFFh. IEC15 to IEC1 correspond to INT15 to INT1, respectively. Each bit is encoded as follows: 0: No effect. 1: Pending, edge-triggered interrupt cleared.
0	**Reserved.**

Edge Interrupt Clear Register 1 (IECLR1)

IECLR is used to clear pending, edge-triggered interrupts.

Location: FF FE14h

Type: WO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IEC31-16**															

Bit	Description
15-0	**IEC31-16 (Edge Interrupt Clear).** Each bit clears the corresponding bit in ISTAT1 register. Writing to the bit positions of level-triggered interrupts has no effect. Read always returns FFFFh. IEC31 to IEC16 correspond to INT31 to INT16, respectively. Each bit is encoded as follows: 0: No effect. 1: Pending, edge-triggered interrupt cleared.

4.3.5 Usage Hints

Initializing

The recommended initialization sequence is:

1. Initialize both the INTBASE register and the interrupt stack pointer of the core.
2. Prepare the interrupt routines of the interrupts used.
3. Clear the edge interrupt used.
4. Set the relevant bits of the modules.
5. Set the relevant bits in IENAM register.
6. Set EXNMI register.
7. Enable core interrupt.

Clearing

Clearing an interrupt request before it is serviced may cause a spurious interrupt (i.e., when the core detects an interrupt not reflected by IVCT). Clear interrupt requests only when interrupts are disabled. Clear IENAM bits and ISTAT bits while the core interrupts are disabled (i.e., bits I and/or E in PSR register are cleared).

Nesting

The IENAM registers can be used by interrupt handlers to allow interrupt nesting. When the core acknowledges an interrupt, it disables maskable interrupts by clearing bit I in PSR register and performs the interrupt service routine. This routine can enable nested interrupts by setting bit I in PSR register and can use the IENAM registers to control which interrupts are allowed.

4.0 Embedded Controller Modules (Continued)

4.4 GENERAL-PURPOSE I/O (GPIO) PORTS

The WPC8763L includes General-Purpose I/O (GPIO) ports that support input/output, input only and output only configurations. The programing model is the same for all port types. When a port has an unsupported feature, the register field controlling that feature is reserved. Modifying the reserved field may cause undetermined results.

The GPIO pins are organized in ports. Each port is 8-bits wide and includes up to eight GPIO pins (some ports may have fewer than eight GPIO pins). When some GPIO pins in a port are not implemented, the corresponding port registers bits are reserved.

4.4.1 Features

- General-Purpose Input/Output (GPIO) Port.
  - Each pin functions as input and/or output signal.
  - Direction register controls the direction for each bidirectional GPIO pin.
  - Weak pull-up and/or pull-down.
  - Configurable output drive disable, on V_{DD} powered off.
  - Back-drive protection on all GPIO pins except for pins noted in Section 2.4.4 on page 27.
- Some GPIO pins can be configured to echo the state of other pins; see Section 4.6.1 on page 88.
- Some GPIO pins have event detection capability, to generate wake-up/interrupt; see Section 4.2 on page 66.

4.4.2 GPIO Port Functional Description

Output Buffer

The output buffer is a TRI-STATE buffer. The output type for each pin is listed in the device pin list; see Section 2.4 on page 25. The output buffer characteristics are defined in the device electrical specifications; see Section 8.2 on page 279.

Input Buffer

The input type for each pin is listed in the device pin list. The input buffer characteristics are defined in the device electrical specifications. After reset, it is recommended that the firmware configures the not-connected GPIO pins to internal pulling; see "Weak Pull-Up and Pull-Down" on page 81.

Weak Pull-Up and Pull-Down

A port pin may support weak internal pull-up and/or pull-down. The pull up/down is enabled by P*x*PULL register bits. Selection between pull-up and pull-down is via the P*x*PUD register bits. When only one type of weak pulling is supported, setting the P*x*PUD register bit to an unsupported type may cause undetermined results. When weak pulling is not supported by a pin, setting the corresponding P*x*PULL register bit to logic 1 has no effect.

When the alternate pin functionality is selected, internal pulling is changed (enabled or disabled) according to the pulling requirements of the alternate function (see Section 2.5 on page 32). However, for alternate functions (listed in Section 2.5 on page 32) that do not specifically require internal pulling, the default pulling state is not changed and can be controlled via the corresponding P*x*PULL[*y*] bit.

Port Direction

The Port Direction register (P*x*DIR) controls the direction of the port. If set to 1, it enables the output buffer (i.e., the pin direction is output). If set to 0, it puts the output buffer in TRI-STATE mode (i.e., the pin direction is input); in this case, if the corresponding bit in P*x*PULL register is set, it also enables the pull-up or pull-down.

Data Output

The Data Output (P*x*DOUT) register holds the data to be driven onto the pin when the respective pin is configured as GPIO and its direction is set as output.

Data Input

The Data Input (P*x*DIN) register returns the current value/state of the pin.

Enable by V_{DD}

The Drive Enable by V_{DD} Present (P*x*ENVDD) register enables automatic shut-down of the output buffer and of the pull-up/down when V_{DD} supply is powered off (i.e., when V_{DD} is detected "off" by the WPC8763L). When an output pin does not support this functionality, the corresponding P*x*ENVDD has no effect on the buffer and on the pull-up/down disable.

4.0 Embedded Controller Modules (Continued)

Open Drain

To use a standard GPIO pin, or an open-drain GPIO pin (see Section 4.4.3) as an inverting open-drain output buffer, the firmware should set the corresponding bit in P*x*DOUT register to 0 and then use P*x*DIR register to set the value to the port pin.

When the signal direction is set as output (P*x*DIR register bit is set to 1), a value of 0 is driven by the pin. When the direction is set for input (P*x*DIR register bit is set to 0), the pin is in TRI-STATE mode (i.e., not driven).

Echo Function

Some GPIO pins can echo the value of a different GPIO pin. When echo is enabled (see Section 4.6.1 on page 88), the current level at the echoed GPIO input pin (and not the respective bit in the P*x*DOUT register) is used to drive the output buffer.

4.4.3 Non-Standard GPIO Ports

The following GPIO ports/pins are different from the standard GPIO model.

- GPI90-GPI97 pins are input only, without weak pull-up or weak pull-down. Therefore, setting the P*x*DIR, P*x*PULL and P*x*PUD bits has no effect.
- GPO47, GPO76, GPO82-84 pins are output only; therefore they do not have a Data Input register. Their weak pull-up or weak pull-down can be enabled if the output buffer is disabled (i.e., if the respective bit in P*x*DIR register is set to 0).
- GPIO03-GPIO05 and GPIO07 pins have an open-drain output buffer (i.e., they drive only low level).

4.4.4 GPIO Ports Memory Map

The GPIO port registers are arranged in a unified structure. Each port is allocated 16 bytes starting at base address xx xxx0h and ending at address xx xxxFh. All ports have the same address offsets for the various registers; therefore, moving functions from port to port only involves changing the base address of the port used.

Port Number	Base Address
GPIO0	FF F200h
GPIO1	FF F210h
GPIO2	FF F220h
GPIO3	FF F230h
GPIO4	FF F240h
GPIO5	FF F250h
GPIO6	FF F260h
GPIO7	FF F270h
GPIO8	FF F280h
GPI9	FF F290h

4.4.5 GPIO Port Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.
All GPIO ports registers are powered by V_{CC} and are reset by Core Domain reset.

GPIO Register Map

Location	Mnemonic	Register Name	Size	Type
GPIO*x* Base + 00h	P*x*DOUT	Port GPIO*x* Data Out	Byte	R/W
GPIO*x* Base + 01h	P*x*DIN	Port GPIO*x* Data In	Byte	RO
GPIO*x* Base + 02h	P*x*DIR	Port GPIO*x* Direction	Byte	R/W
GPIO*x* Base + 03h	P*x*PULL	Port GPIO*x* Pull-Up or Pull-Down Enable	Byte	R/W
GPIO*x* Base + 04h	P*x*PUD	Port GPIO*x* Pull-Up/Down Selection	Byte	R/W
GPIO*x* Base + 05h	P*x*ENVDD	Port GPIO*x* Drive Enable by V_{DD} Present	Byte	R/W

4.0 Embedded Controller Modules (Continued)

Port GPIO*x* Data Out Register (P*x*DOUT)

Writing to P*x*DOUT sets the output values of the GPIO pins. Reading from P*x*DOUT returns the last value written to the register. This register is reset to 00h.

Location: Offset 00h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**GPIO*x* Port Data Out**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**GPIO*x* Port Data Out.** Each of these bits controls the state of a GPIO pin when the GPIO pin is configured as output. When open-drain operation is required, the respective pin bit should be set to 0. In this case, the corresponding P*x*DIR register bit is used to control the output state.

Port GPIO*x* Data In Register (P*x*DIN)

Reading from P*x*DIN returns the current value of the GPIO pins in the port. Not-implemented bits return 0.

Location: Offset 01h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**GPIO*x* Port Data In**							

Bit	Type	Description
7-0	RO	**GPIO*x* Port Data In.** The value read from each of these bits reflects the actual status of the respective GPIO pin.

Port GPIO*x* Direction Register (P*x*DIR)

PxDIR configures the direction of the GPIO*x* pins. It is cleared on reset, which configures all the pins in port GPIO*x* as inputs.

Location: Offset 02h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**GPIO*x* Port Direction**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**GPIO*x* Port Direction.** Each of these bits controls the output buffer of a GPIO pin. This bit is also used in open-drain mode as the data out control. Setting this bit drives the respective GPIO output low. Clearing this bit sets the output in TRI-STATE mode. 0: The pin is configured as input (output buffer is disabled) (default). 1: The pin is configured as output (output buffer is enabled).

4.0 Embedded Controller Modules (Continued)

Port GPIO*x* Pull-Up or Pull-Down Enable Register (P*x*PULL)

PxPULL controls the pull-up or pull-down for the related pin. On reset, it is cleared, disabling all pull-ups except when otherwise noted in "Power-Up Configuration" on page 85.

Location: Offset 03h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**GPIO*x* Port Pull-Up or Pull-Down Enable**							
Reset	See Section 4.4.6 on page 85							

Bit	Type	Description
7-0	R/W	**GPIO*x* Port Pull-Up or Pull-Down Enable.** Each of these bits controls the pull-up or pull-down enable of a GPIO pin. The pulling type is selected in the P*x*PUD register. 0: The pin pull-up or pull-down are disabled. 1: The selected pull-up or pull-down for the pin is enabled.

Port GPIO*x* Pull-Up/Down Selection Register (P*x*PUD)

P*x*PUD elects between pull-up or pull-down for GPIO*x* port pins. It is reset to 00h unless otherwise noted in "Power-Up Configuration" on page 85.

Location: Offset 04h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**GPIO*x* Port Pull-Up/Down Select**							
Reset	See Section 4.4.6 on page 85							

Bit	Type	Description
7-0	R/W	**GPIO*x* Port Pull-Up/Down Select.** Each of these bits selects either pull-up or pull-down for a GPIO pin. 0: Pull-up selected for the pin. 1: Pull-down selected for the pin.

Port GPIO*x* Drive Enable by V_{DD} Present (PxENVDD)

PxENVDD controls the Drive Enable by V_{DD} Present option for the port GPIO*x* pins. On reset, it is set for all the port pins supporting the Drive Enable by V_{DD} Present option unless otherwise noted in "Power-Up Configuration" on page 85.

Location: Offset 05h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**GPIOx Port Drive Enable by V_{DD} Present Control**							
Reset	See Section 4.4.6 on page 85							

Bit	Type	Description
7-0	R/W	**GPIO*x* Port Drive Enable by V_{DD} Present Control**. Each of these bits selects the Drive Enable by V_{DD} Present functionality for a GPIO pin. 0: Pin output buffer and pull-up/pull-down are controlled only by P*x*DIR, P*x*PULL and P*x*PUD bits. 1: Pin output buffer and pull-up/pull-down are disabled when V_{DD} supply is powered-off, regardless of P*x*DIR state.

4.0 Embedded Controller Modules (Continued)

4.4.6 Power-Up Configuration

The following table lists the default values of the P*x*DIR, P*x*PULL, P*x*PUD and P*x*ENVDD registers for each GPIO*x* port, following Core Domain reset.

.

Port Number	P*x*DIR	P*x*PULL	P*x*PUD	P*x*ENVDD
GPIO0	0000 0X0X[1]b	0000 0X0Xb	0000 0X0Xb	0000 0X0Xb
GPIO1	X0X0 0000b	X1X0 0001b	X1X0 0000b	X1X0 00X[2]1b
GPIO2	0000 0X00b	0001 0X00b	0000 0X00b	000X[2] 0X00b
GPIO3	X0X0 0000b	X0X0 0000b	X0X0 0000b	X1X1 0000b
GPIO4	0000 00X0b	0000 00X0b	0000 00X0b	1111 11X1b
GPIO5	000X 0000b	000X 0000b	000X 0000b	000X 1111b
GPIO6	X0X0 0000b	X0X0 0000b	X0X0 0000b	X0X0 0000b
GPIO7	000X X000b	100X X010b	000X X010b	000X X111b
GPIO8	0XX0 000Xb	0XX0 000Xb	0XX0 000Xb	1XX1 110Xb
GPI9	N.A.	N.A.	N.A.	0000 0000b

1. X = N.A.
2. The GPIO pin is powered by V_{DD}, therefore the bit is not implemented.

4.0 Embedded Controller Modules (Continued)

4.5 KEYBOARD SCAN

4.5.1 Overview

Internal keyboard scanning is supported by up to 18 open-drain output signals and eight input signals. Switch-based keyboard matrices are supported by CMOS Schmitt trigger inputs that have internal pull-up resistors. For power efficiency, the inputs include interrupt and wake-up capability so that pressing/releasing keys can be identified without scanning the keyboard matrix in either Active, Idle or Deep Idle mode. The keyboard interrupt is controlled by the MIWU; see Table 10 on page 66.

4.5.2 Keyboard Scan Registers

All Keyboard Scan registers are powered by V_{CC}. They are set to their default values by Core Domain reset.

Keyboard Scan Register Map

Location	Mnemonic	Register Name	Size	Type
FF F6C4h	KBSIN	Keyboard Scan In	Byte	RO
FF F6C5h	KBSINPU	Keyboard Scan In Pull-Up Enable	Byte	R/W
FF F6C6h	KBSOUT0	Keyboard Scan Out 0	Word	R/W
FF F6C8h	KBSOUT1	Keyboard Scan Out 1	Word	R/W

Keyboard Scan In Register (KBSIN)

Location: FF F6C4h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**KBSIN7**	**KBSIN6**	**KBSIN5**	**KBSIN4**	**KBSIN3**	**KBSIN2**	**KBSIN1**	**KBSIN0**

Bit	Description
7-0	**KBSIN7-0 (Keyboard Scan In Bits 7-0).** When read, returns the level of the KBSIN7-0 signals.

Keyboard Scan In Pull-Up Enable Register (KBSINPU)

Location: FF F6C5h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**KBSINPU7**	**KBSINPU6**	**KBSINPU5**	**KBSINPU4**	**KBSINPU3**	**KBSINPU2**	**KBSINPU1**	**KBSINPU0**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**KBSINPU7-0 (Keyboard Scan In Pull-Up Enable Bits 7-0).** When set to 1, a pull-up resistor is connected to the respective KBSIN pin.

4.0 Embedded Controller Modules (Continued)

Keyboard Scan Out 0 Register (KBSOUT0)

Location: FF F6C6h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**KBSOUT15**	**KBSOUT14**	**KBSOUT13**	**KBSOUT12**	**KBSOUT11**	**KBSOUT10**	**KBSOUT9**	**KBSOUT8**
Reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	**KBSOUT7**	**KBSOUT6**	**KBSOUT5**	**KBSOUT4**	**KBSOUT3**	**KBSOUT2**	**KBSOUT1**	**KBSOUT0**
Reset	1	1	1	1	1	1	1	1

Bit	Description
15-0	**KBSOUT15-0 (Keyboard Scan Out Bits 15-0).** Controls the open-drain driver of the respective KBSOUT15-0 pin.

Keyboard Scan Out 1 Register (KBSOUT1)

Location: FF F6C8h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**							
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**KBSOUT17**	**KBSOUT16**
Reset	0	0	0	0	0	0	1	1

Bit	Description
15-2	**Reserved.**
1-0	**KBSOUT17-16 (Keyboard Scan Out Bits 17-16).** Controls the open-drain driver of the respective KBSOUT17-16 pin.

4.0 Embedded Controller Modules (Continued)

4.6 SYSTEM GLUE FUNCTIONS

The System Glue module includes the following functions:

- GPIO echo function control (IOEE register).
- SMBus wake-up support.

4.6.1 GPIO Echo Function

Some GPIO pins may be paired to generate an input to output echoing with software masking. The input GPIO ports are GPIO01 and GPIO03. The output ports are GPIO21 and GPIO23 respectively. The echo function outputs reflect the values of the echo inputs only when the respective Echo Enable bit in IOEE register is set and the respective GPIO output pin is configured to output. The input pin can also be configured for event detection, generating wake-up and/or interrupt; see Section 4.2 on page 66.

Table 12. GPIO Echo Functions Routing and Echo Enable Bit Assignments

Input Port	Echo Enable Bit	Output Port
GPIO01	EEP1 bit in IOEE register	GPIO21
GPIO03	EEP3 bit in IOEE register	GPIO23

4.6.2 SMBus Wake-Up Support

To manage SMBus wake-up efficiently, an indication of activity on the SMBus ports is provided. The bits of the SMBus Start Detection Register (SMB_SBD) indicate if a Start condition is detected at an SMBus port. If a Start condition is detected at an SMBus port **and** the respective bit in SMB_EEN register is set to 1, an event is generated to the MIWU. SMBus port 1 events are connected to WUI43, MIWU input (see Figure 10 on page 88). SMBus port 2 events are connected to WUI44 MIWU input.

WUI43 and WUI44, MIWU inputs also generate INT7 to the ICU via WKINTD, MIWU output.

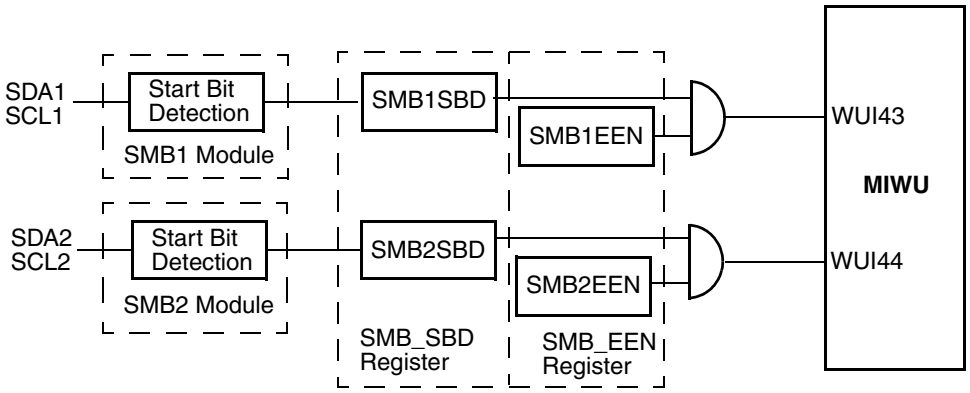


Figure 10. SMB Start Condition Events Connectivity to MIWU

For a connectivity table of SMBus port events to MIWU inputs, see Table 10 on page 66 in the MIWU.

4.6.3 System Glue Registers

All System Glue registers are powered by V_{CC}. They are set to their default values by Core Domain reset.

System Glue Register Map

Location	Mnemonic	Register Name	Size	Type
FF F6C0h	IOEE	Input to Output Echo Enable	Byte	R/W
FF F6C1h	Reserved			
FF F6C2h	SMB_SBD	SMBus Start Bit Detection	Byte	R/W1C
FF F6C3h	SMB_EEN	SMBus Event Enable	Byte	R/W

4.0 Embedded Controller Modules (Continued)

Input to Output Echo Enable Register (IOEE)

IOEE is read/write and byte-wide. It enables echoing of some General-Purpose Input ports to specified General-Purpose Output ports. The IOEE bits have an impact only when the output and Input GPIOs are configured as GPIO ports.

Location: FF F6C0h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**				**EEP3**	**Reserved**	**EEP1**	**Reserved**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-4	**Reserved.**
3	**EEP3 (Echo Enable Port 3).** Enables the echo of function of GPIO03 to GPIO23. Controls the respective GPIO input and output ports (see Table 12 on page 88). 0: Echo Disabled (default). 1: Echo Enabled.
2	**Reserved.**
1	**EEP1 (Echo Enable Port 1).** Enables the echo of function of GPIO01 to GPIO21. Controls the respective GPIO input and output ports (see Table 12 on page 88). 0: Echo Disabled (default). 1: Echo Enabled.
0	**Reserved.**

SMBus Start Bit Detection Register (SMB_SBD)

SMB_SBD indicates when a Start condition is detected at an SMBus port. In this case, an event is generated to the MIWU if the respective bit in SMB_EEN register is set.

Location: FF F6C2h

Type: R/W1C

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**SMB2SBD**	**SMB1SBD**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-2	**Reserved.**
1	**SMB2SBD (SMBus 2 Start Bit Detection).** Indicates that an SMBus Start condition was detected on SMBus 2 port of SMB 2 module. Write 1 to clear this bit after SMB2_SL bit in DEVALT2 register (see page 38) is modified. 0: No Start condition was detected (default). 1: Start condition was detected on SMBus 2 port (SDA2 and SCL2 pins).
0	**SMB1SBD (SMBus 1 Start Bit Detection).** Indicates that an SMBus Start condition was detected on SMBus 1 port of SMB 1 module. 0: No Start condition was detected (default). 1: Start condition was detected on SMBus 1 port (SDA1 and SCL1 pins).

4.0 Embedded Controller Modules (Continued)

SMBus Event Enable Register (SMB_EEN)

The detection of a Start condition at one of the SMBus ports sets one of the status bits of the SMB_SBD register and generates an event if the respective bit in the SMB_EEN register is 1 (see also SMB_SBD register).

Location: FF F6C3h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**SMB2EEN**	**SMB1EEN**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-2	**Reserved.**
1	**SMB2EEN (SMBus 2 Event Enable).** Enables an event to be asserted if SMB_SBD.SMB2SBD bit is set. 0: Event is disabled (default). 1: SMBus 2 Start condition event is enabled.
0	**SMB1EEN (SMBus 1 Event Enable).** Enables an event to be asserted if SMB_SBD.SMB1SBD bit is set. 0: Event is disabled (default). 1: SMBus 1 Start condition event is enabled.

4.0 Embedded Controller Modules (Continued)

4.7 PS/2 INTERFACE

The PS/2 protocol is an industry-standard, PC-AT-compatible interface for keyboards. It uses a two-wire bidirectional TTL interface for data transmission. Several vendors also supply PS/2 mouse products and other pointing devices that employ the same type of interface.

The WPC8763L provides three PS/2 data transfer channels. Each channel has two quasi-bidirectional signals that serve as direct interfaces to an external keyboard, mouse or any other PS/2-compatible pointing device. Since the three channels are identical, the connector ports are interchangeable.

4.7.1 Features

- Three PS/2 channels.
- Enable/Disable for each of the three channels.
- Automatic hardware shift mechanism.
- Hardware support for PS/2 auxiliary device protocol.
- Processor interrupts at the beginning and end of data transfer.
- Optional software-based PS/2 implementation.

4.7.2 General Description

In the previous generation of keyboard controllers, firmware executed the PS/2 device interface by toggling the interface signals. The WPC8763L supports this bit toggling mode via either polling or interrupt-driven clock edge detection.

The PS/2 device firmware is significantly simplified through the use of a hardware accelerator mechanism. The accelerator includes an 8-bit shift register, a state-machine, and control logic that handle both the incoming and outgoing data. The PS/2 module reduces the code overhead, performance requirements and overall interrupt latency of the core firmware. The hardware is designed to meet the PS/2 device interface as defined in *Keyboard and Auxiliary Device Controller (Types 1 and 2), August 1988.*

Naming Conventions

In this section, the following naming conventions are used:

- The term “channel” describes the interface to one of the PS/2 devices and its two associated signals (clock and data).
- The term “shift mechanism” refers to the hardware accelerator.
- The term “PS/2 interface” refers to the entire mechanism.

Interface Signals

The PS/2 interface includes six external signals (PSCLK3-1 and PSDAT3-1) and six registers.

Module Block Diagram

A schematic description of the PS/2 interface is shown in Figure 11. The interface to the three channels is symmetric; only channel 1 is detailed in the figure.

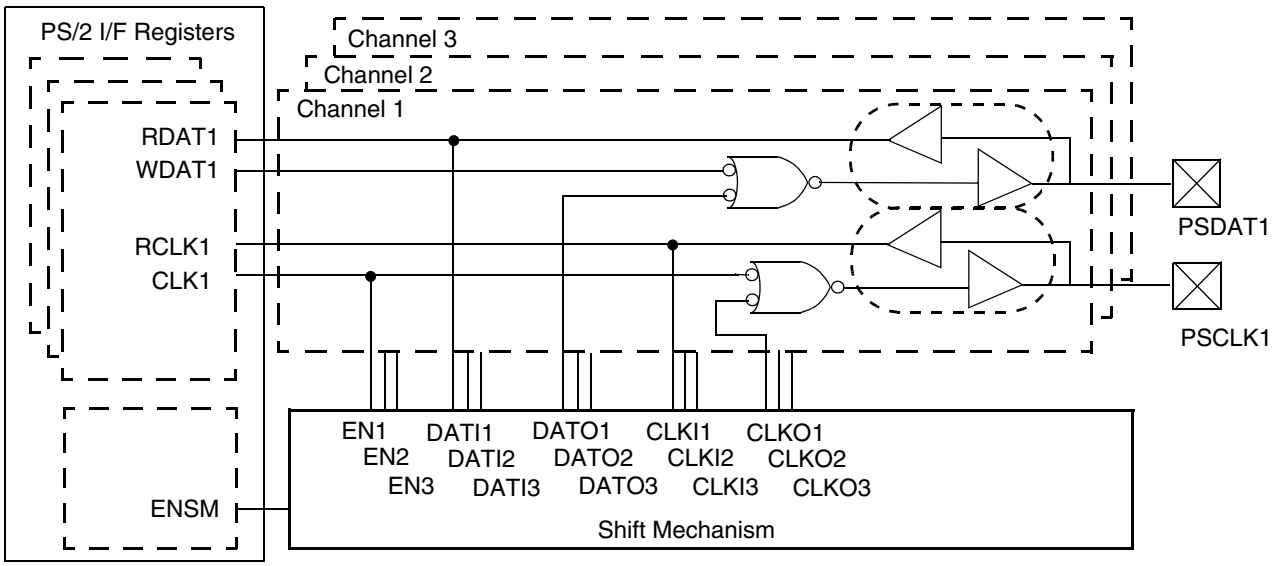


Figure 11. PS/2 Interface Functional Diagram

4.0 Embedded Controller Modules (Continued)

Quasi-Bidirectional Drivers

The quasi-bidirectional drivers have an open-drain output, an internal pull-up and a low-impedance pull-up. The open-drain output pulls the signal low when the output buffer data is 0. The weak pull-up is active when the output buffer data is 1 and WPUEN in PSCON register is set (1). The low impedance pull-up is active when the WPC8763L changes the output data buffer from 0 to 1, thereby reducing the low-to-high transition time. The length of time that the low-impedance pull-up is active is determined by HDRV field in PSCON register.

Interrupt Signal

The firmware can use an interrupt-driven mechanism to implement the PS/2 device interface via PSINT1. More details on the use of the interrupt are provided in "Interrupt Generation" on page 94. Figure 12 shows the interrupt mechanism with the associated enable bits.

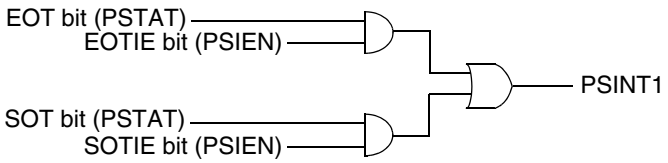


Figure 12. PS/2 Interface Interrupt Signal

Power Modes

The PS/2 interface is active only when the WPC8763L is in Active mode. The shift mechanism should be disabled before entering Idle or Deep Idle mode. In Idle and Deep Idle modes, the state of output signals cannot be changed (i.e., the firmware cannot write to PSOSIG register, and the shift mechanism does not function).

When the WPC8763L needs to wake up on a Start bit detection by the MIWU, the PS/2 channels that can serve as wake-up event sources must be enabled before entering Idle or Deep Idle mode. To enable them, set to 1 their corresponding CLK bits in PSOSIG register.

The MIWU can be used to identify a start bit in Idle or Deep Idle mode and to return the WPC8763L to Active mode. The MIWU receives PSCLK3-1 and PSDAT3-1 signals as inputs (see Table 10 on page 66). The MIWU should be programed to identify a falling edge on the clock or data lines of the enabled channels. In this configuration (clock or data), a start bit causes the WPC8763L to switch from Idle or Deep Idle mode to Active mode. Once Active mode is reached, the firmware should cancel the transaction that started and then enable retransmission of the information by the device.

PS/2 Interface Operation

The PS/2 interface has two basic operating methods:

- With the shift mechanism disabled
- With the shift mechanism enabled

The following sections describe how to use the PS/2 interface with each of these operating methods.

4.7.3 Operating With the Shift Mechanism Disabled

The shift mechanism is disabled when EN bit in PSCON register is cleared (0). In this state, the PS/2 clock and data signals are controlled by the firmware, which performs the PS/2 protocol by manipulating the PS/2 clock and data signals.

Clock Signal Control

CLK3-1 bits in PSOSIG register control the value of the respective clock signals (PSCLK3-1). When one of these bits is cleared (0), the relevant pin is held low. When set (1), the open-drain output is open and the respective clock signal is either floating or held high by the pull-up. In this case, an external device can force the respective clock signal low.

When reading PSISIG register, bits RCLK3-1 indicate the current state of the corresponding clock signal.

Data Signal Control

WDAT3-1 bits in PSOSIG register control the value of the respective data signals (PSDAT3-1). When one of these bits is cleared (0), the relevant data signal is held low. When set (1), the open-drain output is open and the respective data signal is held high by the pull-up. In this case, an external device can force the respective data signal low.

When reading PSISIG register, bits RDAT3-1 indicate the current state of the corresponding data signal.

4.0 Embedded Controller Modules (Continued)

4.7.4 Operating With the Shift Mechanism Enabled

The shift mechanism is designed to offload the bit level handling of the data transfer from the firmware to a hardware mechanism; this improves system tolerance to interrupt latency. The mechanism includes a shift register and a state machine that controls the PS/2 protocol.

Figure 13 shows the shift mechanism PS/2 data transfer sequence. There are three basic modes: Disabled, Receive and Transmit. Different states in each mode define the progress of the data transfer. The rest of this section details the use of the shift mechanism for implementing a PS/2 data transfer.

Reset the Shift Mechanism

Clearing either the shift mechanism enable bit (EN = 0 in PSCON register) or all channel clock bits (CLK3-1 = 0) resets the shift mechanism. Setting CLK3-1 to 0 is recommended to prevent glitches on the clock signals. In reset state, PSTAT register is cleared (00h), and the state of the PS/2 clock and data signals (PSCLK3-1 and PSDAT3-1) is set according to the value of their control bits (CLK3-1 and WDAT3-1, respectively).

When the shift mechanism is reset while in an unknown state or in Transmit Idle state, the firmware should set (1) WDAT3-1 before the shift mechanism is reset.

Enable the Shift Mechanism

To enable the shift mechanism, verify that PSOSIG register is set to 47h and then set (1) EN bit in PSCON register. This puts the shift register state machine in Receive Inactive or Transmit Inactive state (XMT is 0 or 1, respectively, in PSCON register). In either of these states, the clock signals (PSCLK3-1) are low and the data signals PSDAT3-1 are either floating or pulled high.

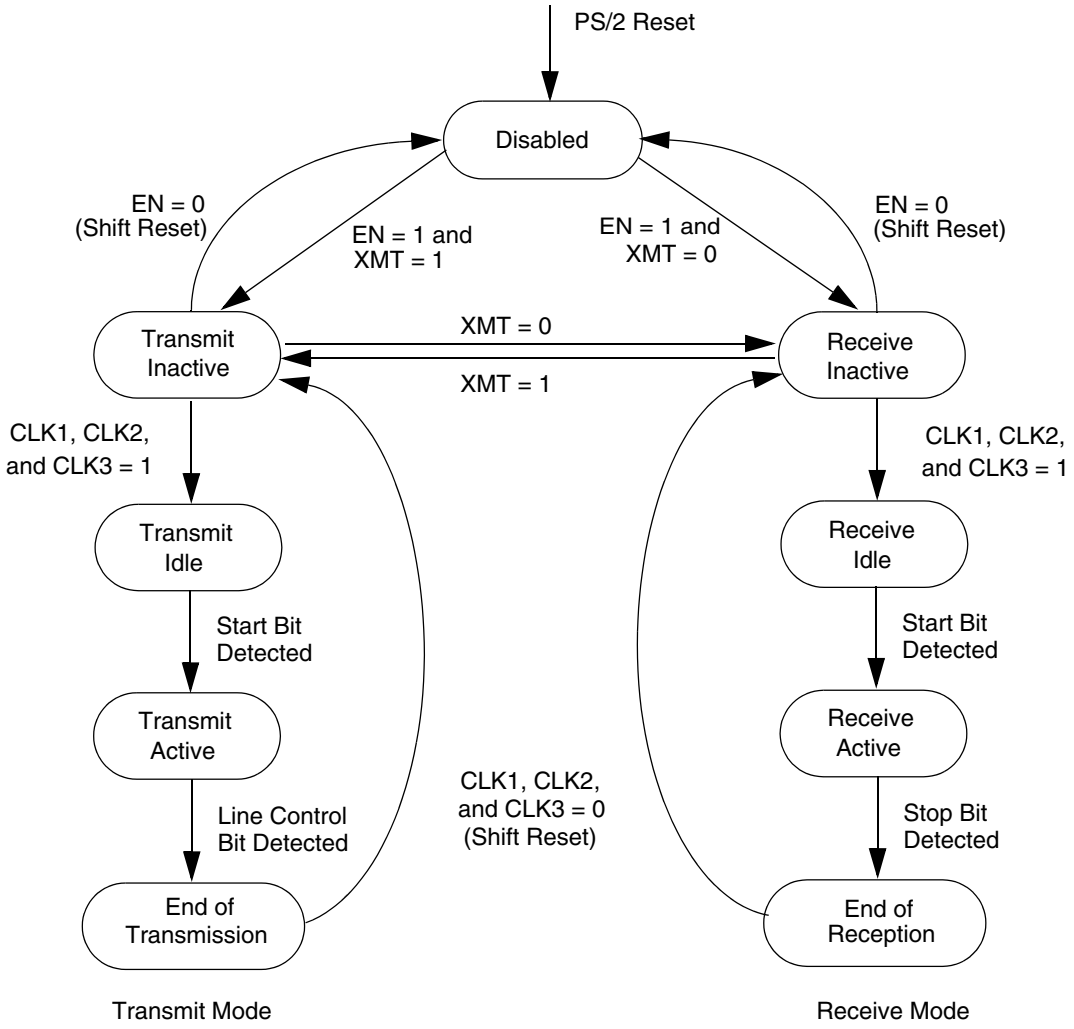


Figure 13. Shift Mechanism State Diagram

4.0 Embedded Controller Modules (Continued)

Shift Status

The PSTAT register indicates the current status of the shift mechanism. The data transfer process can be in one of three states:

- Shifter Empty:
  The shift mechanism is in Receive Inactive, Receive Idle, Transmit Inactive or Transmit Idle state. The PSTAT is cleared because none of the enabled devices sent a start bit.
- Start Bit Detected:
  The shift mechanism is in Receive Active or Transmit Active state. This indicates that a start bit was identified for at least one of the channels and the shift process has begun. SOT bit in PSTAT register indicates the detection of the start bit; ACH field in PSTAT register indicates the active channel (the channel on which the start bit was detected).
- End Of Transaction:
  The shift mechanism is in End Of Reception or End Of Transmission state. This indicates that the last bit of the transfer sequence was detected (and the data can therefore be read from PSDAT register) or that the data transmission was completed (for receive and transmit, respectively). EOT bit in PSTAT register indicates transfer completion. If a parity error was detected in the received data, PERR bit in PSTAT register is set. If a stop bit was detected low instead of high, RFERR bit in PSTAT register is set.

Input Signal Debounce

The WPC8763L performs a debounce operation on the clock input signal before determining its logical value. IDB field in PSCON register determines the number of clock cycles for which the input signal must be stable to allow a change in its value to be detected.

Interrupt Generation

The PSINT1 is an interrupt signal generated by the shift mechanism to allow an interrupt-driven firmware interface.

The ICU should be programed to detect high-level interrupts on the PSINT1 interrupt; see Section 4.3 on page 74 for details on the ICU. SOTIE and EOTIE bits in PSIEN register mask the interrupt signaling for SOT and EOT bits, respectively, in PSTAT register.

Receive Mode

Receive Inactive

When the shift mechanism is enabled and bit XMT=0 in PSCON register, the shift mechanism enters Receive mode in the Receive Inactive state. When one (or more) channel is enabled, Receive Idle state is entered by setting the channel enable bit (CLK3-1 for channels 3-1, respectively). In this state, the shift-mechanism sets the clock and data lines of the enabled channels high (1) and waits for a start bit.

Receive Idle

In the Receive Idle state, the PS/2 interface waits for input from any one of the enabled channels. The first of the enabled channels to send a start bit is selected for handling by the shift mechanism. The other two channels are disabled by forcing 0 on their clock lines.

Start Bit Detection

The start bit is identified by a falling edge on the clock signal while the data signal is low (0).

If the start bit is identified simultaneously in more than one channel, only one channel (the one with the lowest number) is selected for receive and the transfer on the other channel is aborted. For example, channel 1 has priority over channels 2and 3, channel 2 has priority over channel 3. The data transfer in the non-selected channels is aborted before 10 data bits have been sent (by forcing the clock signal to 0), and the transmitting PS/2 device will resend its data when its interface is re-enabled by the firmware. This mechanism ensures that no incoming data is lost.

When the hardware sets (1) SOT bit and designates the selected channel in ACH field, this indicates receipt of the start bit in PSTAT register. In addition, if SOTIE bit is set in PSIEN register, an interrupt signal to the ICU is set high. The firmware may use this interrupt to start a time-out timer for the data transfer.

Receive Active

After identifying the start bit, the shift mechanism enters the “Receive-Active” state. In this state the clock signal of the selected device (PSCLK1, PSCLK2, PSCLK3) sets the data bit rate. On each falling edge of the clock, new data is sampled on the data signal of the active channel (i.e., PSDAT1 PSDAT2, PSDAT3).

Following the start bit, eight bits of data are received (clocks 2 through 9); a parity bit follows (10th clock) and then a stop bit (11th clock). The stop bit is indicated by a falling edge of the clock with the data signal high (1). If the 11th clock is identified with data low, the receive frame error bit (RFERR in PSTAT register) is set, but the clock is treated as the stop bit.

After the parity is received, the shift mechanism checks the incoming data for parity errors. If there are eight data bits with a value of 1 and the parity bit is even, PERR bit in PSCON register is set, indicating a parity error.

4.0 Embedded Controller Modules (Continued)

End Of Reception

When the stop bit is detected, the shift mechanism enters the End Of Reception state. In this state, the shift mechanism:

- Disables all the clock signals by forcing them low.
- Sets the End Of Transaction status bit (EOT = 1 in PSTAT register).
- If EOTIE bit in PSTAT register is set, the interrupt signal to the ICU is asserted (1).

The shift mechanism stays in this state until it is reset.

Figure 14 shows the receive byte sequence, as defined by the PS/2 standard.

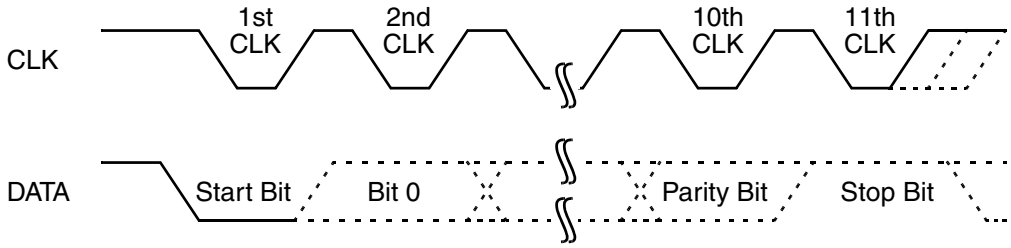


Figure 14. PS/2 Receive Data Byte Timing

Transmit Mode

Transmit Inactive

When the shift mechanism is enabled and XMT bit in PSCON register is set (1):

- The shift mechanism enters Transmit mode in Transmit Inactive state with all clock signals low and data signals high (PSOSIG = 47h).
- The firmware writes the data to be transmitted to the PS/2 data register (PSDAT).
- The data line of the channel to be transmitted is forced low by the firmware clearing its data bit (WDAT3-1 for channels 3-1, respectively).

Transmit Idle

The Transmit Idle state can be entered by setting the channel enable bit (CLK3-1 for channel 3-1, respectively). This enables the channel to be used for transmission. In this state, the shift-mechanism sets the clock of the enabled channel high (1), holds the data line of that channel low and waits for a start bit. When a PS/2 device senses the clock signal high with the data signal low, it identifies a transmit request from the WPC8763L.

The two channels that is not in use are disabled by forcing 0 on their clock lines.

Start Bit Detection

The start bit is identified by a falling edge on the clock signal while the data signal is low (0).

When a start bit is detected, data transmission begins by outputting bit 0 (LSB) of the transmitted data and setting data bits WDAT3-1 in PSOSIG register. This allows bit 0 of the transmitted data to be output on the PS/2 data signal (PSDAT1, PSDAT2, PSDAT3, according to the active channel).

In addition, the hardware sets the SOT bit (to 1) and stores the active channel number in ACH field, indicating transmission of the start bit in PSTAT register. Note that if SOTIE bit in PSIEN register is set, an interrupt signal to the ICU is set high. The firmware can use this interrupt to start a time-out timer for the data transfer.

Transmit Active

After identifying the start bit, the shift mechanism enters the Transmit Active state. The clock signal of the selected device (PSCLK1, PSCLK2, PSCLK3) sets the data bit rate.

After each of the next seven falling edges of the clock line, one more data bit (bits 1 through 7) is driven on the data line of the active channel (PSDAT1, PSDAT2, PSDAT3).

On the ninth falling edge of the clock, the parity bit is output. The parity bit is high (1) if the number of bits with a value of 1 in the transmitted data is even (i.e., odd parity).

The tenth falling edge causes a 1 to be output as a stop bit. The data signal remains high to allow the PS/2 device to send the line control bit.

The auxiliary device then completes the transfer by sending the line-control bit. The line-control bit is identified by the data signal at low level after the 11th falling edge of the clock.

4.0 Embedded Controller Modules (Continued)

End Of Transmission

The End Of Transmission state is entered when the line-control bit is detected. In response, the shift mechanism holds all clock signals low, and if the internal pull-up is enabled, all data signals are pulled high by the internal pull-up.

The End Of Transaction flag (EOT in PSTAT register) is set to indicate that the transmit operation was completed; in addition, if EOTIE bit in PSIEN register is set, the interrupt signal to the ICU is set high.

The shift mechanism stays at this state until it is reset.

Figure 15 shows the transmit byte sequence, as defined by the PS/2 standard.

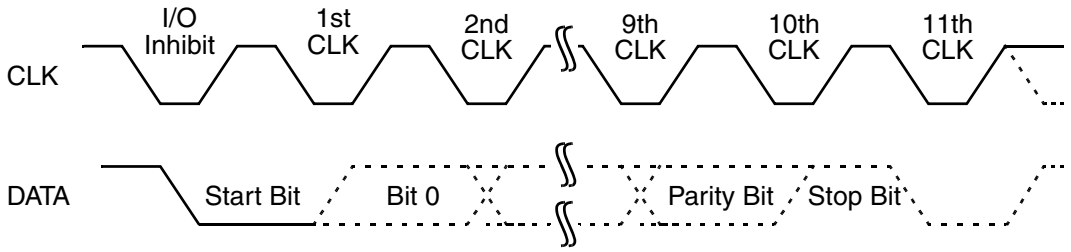


Figure 15. PS/2 Transmit Data Byte Timing

Transfer Abort

At each stage of a receive or transmit operation, the transaction can be aborted by clearing all three channel enable bits (CLK3-1) in PS/2 Output Signal register (PSOSIG) to 0. This resets the shifter state machine and puts it in the Enabled Inactive state. If the shift mechanism is in Transmit Inactive or Transmit Idle state, WDAT3-1 bits should also be set.

4.7.5 PS/2 Interface Registers

These registers are reset by Core Domain reset. For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

PS/2 Register Map

Location	Mnemonic	Register Name	Size	Type
FF F300h	PSDAT	PS/2 Data Register	Byte	R/W
FF F302h	PSTAT	PS/2 Status Register	Byte	RO
FF F304h	PSCON	PS/2 Control Register	Byte	R/W
FF F306h	PSOSIG	PS/2 Output Signal Register	Byte	R/W
FF F308h	PSISIG	PS/2 Input Signal Register	Byte	RO
FF F30Ah	PSIEN	PS/2 Interrupt Enable Register	Byte	R/W

PS/2 Data Register (PSDAT)

In Receive mode, PSDAT holds the data received in the last message from the PS/2 device. In Transmit mode, the data to be shifted out is written to this register. When the PS/2 interface is reset, the contents of this register become invalid.

On reset, the PS/2 interface is set to Receive mode. In this mode, PSDAT should only be read when EOT bit in PSTAT register is set to 1.

Setting the transmit enable bit in PSCON register to 1 (XMT = 1 in PSCON register) puts the PS/2 interface in Transmit mode. PSDAT should only be written when in Transmit mode **and** when all three channel enable bits CLK3-1 in PSOSIG register are cleared (0).

Location: FF F300h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Data**							

Bit	Description
7-0	**Data.** Contains the data received in the last message (or the data to be transmitted in the next transmission). Bit 0 is the first bit to be shifted (LSB).

4.0 Embedded Controller Modules (Continued)

PS/2 Status Register (PSTAT)

PSTAT contains the status information on the data transfer on the PS/2 ports. All non-reserved bits are cleared (0) on reset when CLK1, CLK2 and CLK3 in PSOSIG are cleared **and** when EN bit in PSCON register is cleared. Reading PSTAT does not clear any of its bits.

Location: FF F302h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**RFERR**	**ACH**			**PERR**	**EOT**	**SOT**
Reset	x	0	0	0	0	0	0	0

Bit	Description
7	**Reserved.**
6	**RFERR (Receive Frame Error).** When set to 1, indicates that the stop bit in a received frame was detected low instead of high.
5-3	**ACH (Active Channel).** Defines which PS/2 channel is currently active (i.e., a start bit was detected). If more than one channel become active simultaneously, only the one with the highest priority (lowest number) is flagged. **Bits 5 4 3 Description** 0 0 0: None of the channels is active (default). 0 0 1: Channel 1. 0 1 0: Channel 2. 1 0 0: Channel 3. Others: Reserved.
2	**PERR (Parity Error).** When set to 1, indicates that a parity error was detected in the last data reception. PERR must be ignored during data transmission.
1	**EOT (End Of Transaction).** When set to 1, indicates that a PS/2 data transfer was completed, i.e., a stop bit was detected at Receive mode or a line control bit was detected at Transmit mode.
0	**SOT (Start Of Transaction).** When set to 1, indicates that a start bit was detected. The ACH field (bits 5-3 of this register) indicates on which channel it was detected.

PS/2 Control Register (PSCON)

PSCON controls the operation of the PS/2 interface by enabling it and controlling the data transfer direction. On reset, it is set to 00h.

Location: FF F304h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WPUEN**	**IDB**			**HDRV**		**XMT**	**EN**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**WPUEN (Weak Pull-Up Enable).** 0: Disables the pull-up. In this state, for proper PS/2 operation, the system must ensure that PS/2 interface signals are not left floating (default). 1: Enables the internal pull-up of the output buffer. The pull-up remains active as long as the buffer does not drive the signal to low level.

4.0 Embedded Controller Modules (Continued)

Bit	Description
6-4	**IDB (Input Debounce).** Defines the number of WPC8763L clock cycles during which the clock input is expected to be stable before the shift mechanism identifies its new value. This protects the shift mechanism from false edge detections. The number of WPC8763L clock cycles for which the input should be stable before an edge is detected is: **Bits 6 5 4 Description** 0 0 0: One cycle (default). 0 0 1: Two cycles. 0 1 0: Four cycles. 0 1 1: Eight cycles. 1 0 0: 16 cycles. 1 0 1: 32 cycles.
3-2	**HDRV (High Drive).** Defines the behavior of the quasi-bidirectional buffers on transition from low to high. It defines the period of time for which the output is pulled high with a low-impedance drive (when the WPC8763L changes the output level from low to high). This period is a function of the WPC8763L clock as follows: **Bits 3 2 Description** 0 0: Disabled (default). 0 1: Low-impedance drive for one clock cycle. 1 0: Low-impedance drive for two clock cycles. 1 1: Low-impedance drive for three clock cycles.
1	**XMT (Transmit Enable).** 0: Receive mode. 1: Causes the PS/2 interface to enter Transmit mode.
0	**EN (Shift Mechanism Enable).** 0: The hardware shift mechanism is disabled; the software both controls and monitors the PS/2 signals using PSOSIG and PSISIG registers (default). 1: The hardware shift mechanism is enabled. The enabled channels are controlled by PSOSIG register; Transmit/Receive mode is controlled by XMT bit.

PS/2 Output Signal Register (PSOSIG)

PSOSIG sets the value of the PS/2 port signals. When the shift mechanism is enabled, the clock control bits in this register define the active channel(s). On reset, it is set to 47h.

Location: FF F306h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved (must be '01')**		**CLK3**	**CLK2**	**CLK1**	**WDAT3**	**WDAT2**	**WDAT1**
Reset	0	1	0	0	0	1	1	1

Bit	Description
7-6	**Reserved.** Must be '01'.
5	**CLK3 (Enable Channel 3)** 0: Forces the PSCLK3 pin low (0) and disables channel 0 of the shift mechanism. 1: Depends on whether or not the shift mechanism is enabled. • When the shift mechanism is enabled (EN bit in PSCON register is set to 1), channel 3 of the PS/2 ports is enabled. • When the shift mechanism is disabled (EN bit in PSCON register is set to 0), the clock line output buffer data is 1 (i.e., the signal is pulled high by the pull-up, if enabled, and may be pulled low by an external device).
4	**CLK2 (Enable Channel 2).** Same as CLK3, but for channel 2.
3	**CLK1 (Enable Channel 1).** Same as CLK3, but for channel 1.

Bit	Description
2	**WDAT3 (Write Data Signal Channel 3).** Controls the data output to channel 3 data signal (PSDAT3). Use of this bit depends on whether or not the shift mechanism is enabled. • When the shift mechanism is disabled (EN bit in PSCON register is set to 0), the data in WDAT3 is output to PSDAT3 signal. – If WDAT3 is cleared (0), the output buffer data is 0 (i.e., PSDAT3 is forced low). – If WDAT3 is set (1), the output buffer data is 1 (i.e., PSDAT3 is pulled high by the internal pull-up and may be pulled low by an external device). • When the shift mechanism is enabled (EN=1), WDAT3 should be set to 1, except when the shift mechanism is in Transmit mode. In this case, when in transmit-inactive and it is intended to transmit data to channel 3, the firmware should clear WDAT3 bit to force the transmit signaling (low) to the PS/2 device. Note: WDAT3 is set by the hardware after the WPC8763L detects a start bit (i.e., on entering Transmit Active state). If a transmission is aborted before Transmit Active state, WDAT3 should be set (1) prior to disabling the channel.
1	**WDAT2 (Write Data Signal Channel 2).** Controls the data output to the channel 2 data signal (PSDAT2). For more information, see the description of WDAT3.
0	**WDAT1 (Write Data Signal Channel 1).** Controls the data output to channel 1 data signal (PSDAT1). For more information, see the description of WDAT3.

Note: When CLK1, CLK2, CLK3 are 0, the shift mechanism is reset. In this case, the PSTAT register and the shift state machine are reset to their initial state.

PS/2 Input Signal Register (PSISIG)

PSISIG provides the current value of the PS/2 port signals.

Location: FF F308h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**RCLK3**	**RCLK2**	**RCLK1**	**RDAT3**	**RDAT2**	**RDAT1**

Bit	Description
7-6	**Reserved.**
5	**RCLK3 (Read Clock Signal Channel 3).** When read, returns the current value of the channel 3 clock signal (PSCLK3).
4	**RCLK2 (Read Clock Signal Channel 2).** When read, returns the current value of the channel 2 clock signal (PSCLK2).
3	**RCLK1 (Read Clock Signal Channel 1).** When read, returns the current value of the channel 1 clock signal (PSCLK1).
2	**RDAT3 (Read Data Signal Channel 3).** The current value of the channel 3 data signal (PSDAT3).
1	**RDAT2 (Read Data Signal Channel 2).** The current value of the channel 2 data signal (PSDAT2).
0	**RDAT1 (Read Data Signal Channel 1).** The current value of the channel 1 data signal (PSDAT1).

4.0 Embedded Controller Modules (Continued)

PS/2 Interrupt Enable Register (PSIEN)

PSIEN enables/disables interrupts generated by the PS/2 module. To prevent spurious interrupts, the bits in PSIEN register should be cleared to 0 only when interrupts are disabled (i.e., in the core, I or E bits in PSR register are 0) or when the corresponding interrupts in the ICU are masked. Bits in PSIEN register may be set to 1 at any time. On reset, non-reserved bits of PSIEN are cleared.

Location: FF F30Ah

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved (must be 00h)**						**EOTIE**	**SOTIE**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-2	**Reserved.**
1	**EOTIE (End Of Transaction Interrupt Enable).** Used to enable interrupt generation on End Of Transaction detection. 0: EOT bit in PSTAT register does not affect the interrupt signal (default). 1: The interrupt signal (PSINT1) to the ICU is active (1) when EOT bit in PSTAT register is set. Note: Once set, EOT is not cleared until the shift mechanism is reset. Therefore, EOTIE should be cleared on the first occurrence of an EOT interrupt. EOTIE should be set (1) when the PS/2 module is programed to handle the next transfer.
0	**SOTIE (Start Of Transaction Interrupt Enable).** Used to enable interrupt generation on Start Of Transaction detection. 0: SOT bit in PSTAT register does not affect the interrupt signal (default). 1: The interrupt signal (PSINT1) to the ICU is active (1) when SOT bit in PSTAT register is set. Note: Once set, SOT is not cleared until the shift mechanism is reset. Therefore, SOTIE should be cleared on the first occurrence of an SOT interrupt. SOTIE should be set (1) when the PS/2 module is programed to handle the next transfer.

4.0 Embedded Controller Modules (Continued)

4.8 MULTI-FUNCTION 16-BIT TIMER (MFT16)

The WPC8763L includes two Multi-Function Timer (MFT) modules. The registers of each module are prefixed with Tn, and the signals are suffixed with an n (where n is the module number). Each MFT16 module satisfies a wide range of application requirements. The module includes:

- Clock Source Unit that contains a prescaler and one clock source selector for each counter.
- Main Timer/Counter and Control unit that contains two counters, two reload registers for PWM, Input Capture or Counter modes and two compare registers.
- Mode select and control unit that defines the function of the I/O pins and interrupts.

Figure 16 shows the contents of a MFT16 module functional units.

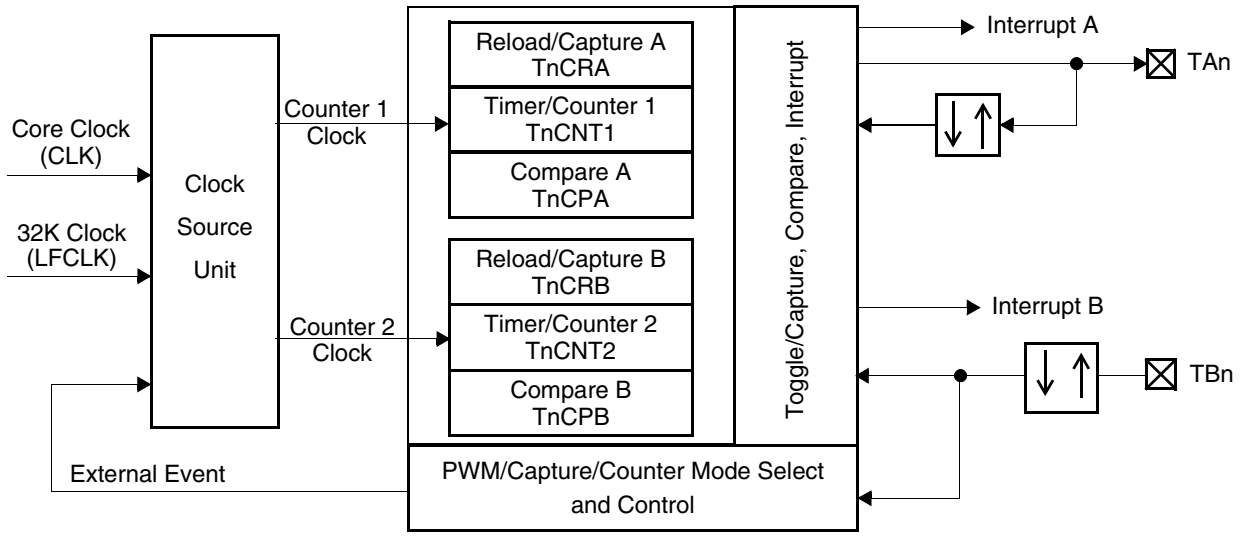


Figure 16. MFT16 Functional Diagram

TA2 output is connected to both a WPC8763L pin and the ADC timer trigger, internal input. TB2 input is not connected to a WPC8763L pin.

4.8.1 Features

- Two 16-bit programmable timers/counters.
- Two 16-bit reload/capture registers. Used as reload or capture registers, depending on the mode of operation.
- Two 16-bit timer or capture/compare registers, which hold a value to be compared with the counter or capture register. An interrupt can be generated on compare match.
- A 5-bit programmable clock prescaler.
- Clock source selectors (one for each counter). These enable each counter to operate in:
  - Pulse accumulate mode
  - External event mode
  - Prescaled core clock mode
  - Slow-speed clock (LFCLK) input mode
- Two I/O pins (TAn and TBn), with programmable edge detection; these operate as:
  - Capture inputs
  - Capture and preset inputs
  - External event (clock) inputs
  - PWM outputs
- Two interrupts, one for each counter, that can be generated/ triggered by:
  - Timer underflow
  - Timer reload
  - Input capture
  - Compare match
- Six pending bits, which can be polled by software, are associated with the two interrupts.

4.0 Embedded Controller Modules (Continued)

4.8.2 Clock Source Unit

The clock source unit, as shown in Figure 17, contains two clock selectors for each counter and a 5-bit clock prescaler.

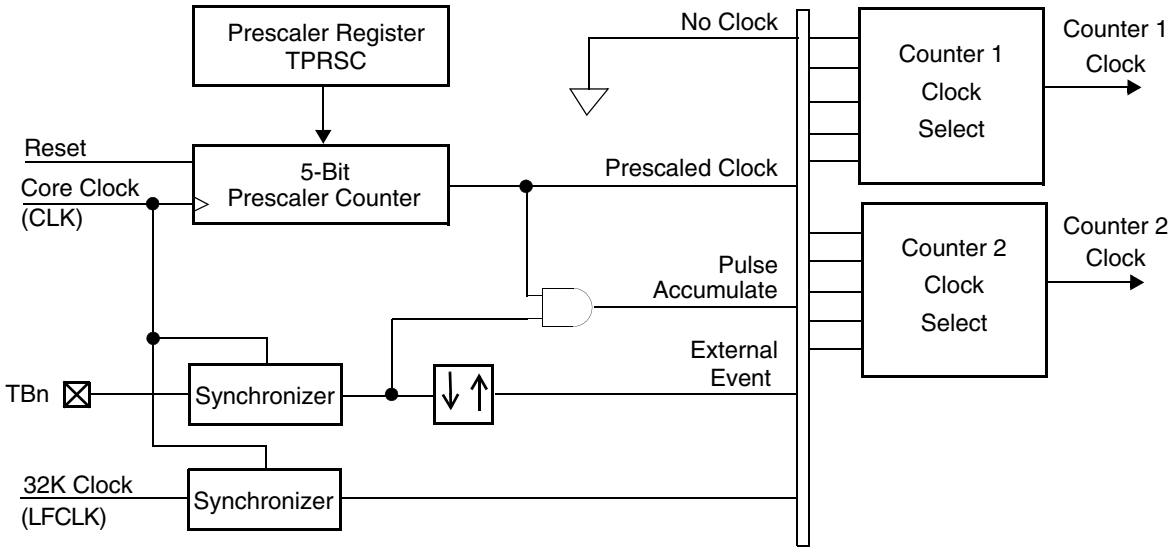


Figure 17. Clock Prescaler and Selector

Prescaler

The 5-bit clock prescaler consists of a prescaler register and a 5-bit counter, allowing the timer to run with a prescaled clock. The core clock is divided by the value contained in TnPRSC+1. The minimum counter clock frequency is thus the core clock divided by 32; the maximum counter clock frequency is equal to the core clock. The prescaler register, TnPRSC, can be read or written by software at any time. The prescaler counter is a 5-bit down counter that cannot be read or written by software. The 5-bit counter and the prescaler register TnPRSC are cleared on reset.

External Event Clock

The TBn I/O pin can be selected as an external event input clock source for either of the two 16-bit counters. The polarity of the input signal is programmable to trigger a count if either a rising or a falling edge is detected on TBn. The minimum pulse width of the external signal is one core clock cycle; therefore, the maximum frequency with which the counter can run in this mode is limited to half the core clock frequency.

Note: An external event clock is not available for TnCNT1 in Dual Input Capture mode (mode 2) and for TnCNT2 in Input Capture and Timer mode (mode 4) because these modes require TBn as an input. external event clock is not available for either counters in Dual Independent Capture mode (mode 5).

Pulse Accumulate Mode

In Pulse Accumulate mode, the counter can also be clocked while an external signal on TBn is either high or low. In this configuration, the output of the prescaler is gated with an external signal applied on TBn input. This mode can be used to obtain a cumulative count of prescaler output clock pulses, as shown in Figure 18.

Note: Pulse Accumulate mode is not available for TnCNT1 in Dual Input Capture mode (mode 2) and for TnCNT2 in Input Capture and Timer mode (mode 4) because these modes require TBn as an input. Pulse accumulate mode is not available in Dual Independent Capture mode (mode 5) for either counters.

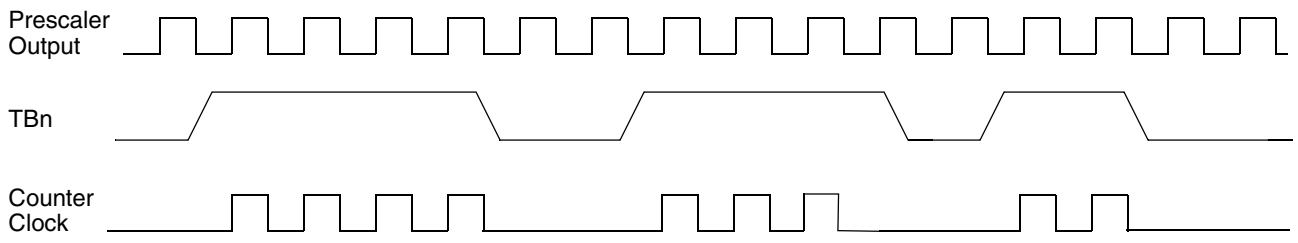


Figure 18. Pulse Accumulate Mode

4.0 Embedded Controller Modules (Continued)

Slow-Speed Clock (LFCLK)

A slow-speed clock of 32.768 KHz (LFCLK) can be used as a clock source for the two 16-bit counters. The MFT16 synchronizes the slow-speed clock with the core clock.

Some power save modes stop the core clock completely. When this occurs, the timer stops counting LFCLK until the core clock resumes.

Counter Clock Source Select

The clock source unit contains two clock source selectors that allow the clock source to be selected independently for each of the two 16-bit counters from one of the following sources:

- No clock, in which case the counter is stopped and capture events are ignored.
- Prescaled core clock.
- External event count based on TBn.
- Pulse Accumulate mode based on TBn.
- Slow Speed Clock (LFCLK) i.e., 32.768 KHz.

4.8.3 Timer/Counter and Control Unit

The Timer/Counter and Control unit consists of two 16-bit counters, TnCNT1 and TnCNT2, in addition to two 16-bit reload/capture registers, TnCRA and TnCRB and two compare registers, TnCPA and TnCPB. The timers are down counters capable of triggering events on underflow detection (count roll-over from 0000h to FFFFh). In addition, the unit contains the mode control logic, which allows the timer to operate in any of five operation modes described below.

Different interrupts can be triggered on certain conditions, and the functionality of the I/O pins changes depending on the mode of operation. Therefore, the interrupt control and the I/O control are an integral part of the timer/counter unit.

Operation Modes

The MFT16 can be configured to operate in any one of five modes, as summarized in Table 13 and described in this section.

Table 13. Operation Modes

Mode	Description	Timer/Counter 1 (TnCNT1)	Reload/Capture A (TnCRA)	Reload/Capture B (TnCRB)	Timer/Counter 2 (TnCNT2)
1	PWM and system timer or external event counter	Counter for PWM	Auto Reload A = PWM time 1	Auto Reload B = PWM time 21gd	System Timer or external event counter
2	Dual input capture and system timer	Capture A and B time base	Capture counter 1 value on TAn event	Capture counter 1 value on TBn event	System Timer
3	Dual independent timer	Time base for first timer	Reload register for TnCNT1	Reload register for TnCNT2	Time base for second timer
4	Input capture and timer	Time base for first timer	Reload register for TnCNT1	Capture counter 2 value on TBn event	Capture B time base
5	Dual independent input capture	Capture A time base	Capture counter 1 value on TAn event	Capture counter 2 value on TBn event	Capture B time base

4.0 Embedded Controller Modules (Continued)

Mode 1, PWM and Counter

PWM can be used to generate pulses of known width and duty cycle on the TAn pin. The timer is clocked by the selected clock. An underflow causes the timer register to be reloaded, alternately, from the TnCRA and TnCRB registers and, if enabled, causes TAn output to toggle. Thus the values stored in TnCRA and TnCRB registers control the high time and low time of the signal produced on TAn. In PWM mode, TnCNT2 can be used as either a simple system timer or an external event counter. The counter can be loaded by software with a specific value; the counter then generates an interrupt after the preprogramed number of external events is received on TBn input.

Figure 19 shows a block diagram of the timer operating in mode 1. In PWM mode, TnCNT1 functions as the time base for the PWM timer. Counter 1 counts down at the clock rate selected via the counter 1 clock selector. When an underflow occurs, the timer register is reloaded alternately from the TnCRA and TnCRB registers, and counting proceeds downward from the loaded value. On reset, and every time this mode is entered, the first reload in this mode is from the TnCRA register. Once enabled, the counter starts counting down from the value currently in TnCNT1. At the first underflow, the timer is loaded from TnCRA; on the second underflow, it is loaded from TnCRB; on the third underflow, it is loaded from TnCRA, and so on. Note that every time the counter is stopped through the selection of “No-Clock” in the counter 1 clock selector (TnCKC), it obtains its first reload value after it has been restarted from TnCRA register.

The timer can be configured to toggle TAn output signal on underflow. This results in the generation of a clock signal on TAn, with the width and duty cycle controlled by the values stored in TnCRA and TnCRB registers. This PWM clock is core-independent because once the timer is set up, no further interaction is required by software (i.e., the core) to generate a continuous PWM signal. Software can select the initial value of the PWM output signal as either high or low; see “Timer I/O Functions” on page 110 for additional details. The timer can be configured to generate separate interrupts on reload from TnCRA and TnCRB. The interrupts can be enabled or disabled under software control. The TAnPND or TnBPND flags, respectively, which are set by the hardware on occurrence of a timer reload, indicate which interrupt occurred; see Section 4.8.5 on page 109 for detailed information.

In this mode of operation, TnCNT2 can be used as a “simple” system timer, an external event counter or a pulse accumulate counter. Counter TnCNT2 counts down with the clock selected via the counter 2 clock selector, and TnCNT2 can be configured to generate an interrupt on underflow if the interrupt is enabled by TnDIEN bit; see Section 4.8.5 on page 109 for detailed information.

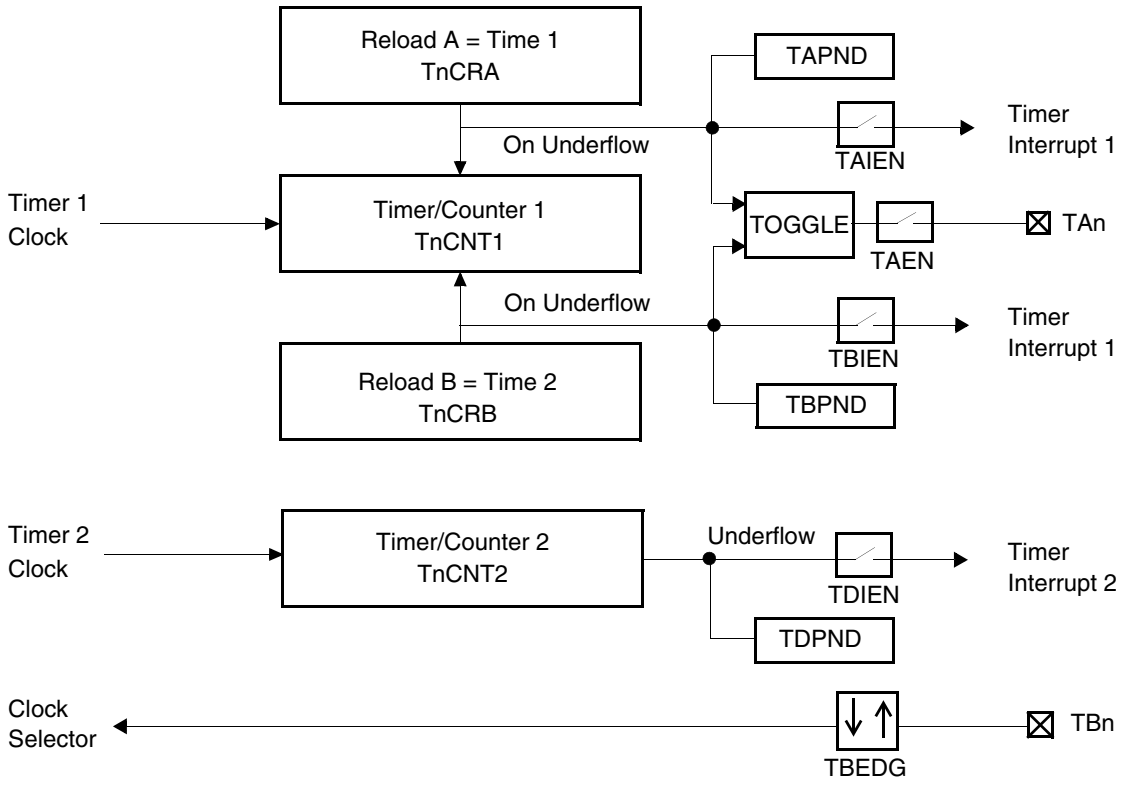


Figure 19. Mode 1, PWM and Counter

4.0 Embedded Controller Modules (Continued)

Mode 2, Dual Input Capture

Dual Input Capture mode can be used to measure either the frequency of an external clock that is slower than the selected clock source frequency or the elapsed time between external events. A transition received on the TAn or TBn pin causes a transfer of TnCNT1 contents to TnCRA or TnCRB register, respectively. In this mode, TnCNT2 can be utilized as a system timer that is preloaded by software and generates an interrupt on underflow.

Figure 20 shows a block diagram of the timer operating in mode 2. In this mode of operation, the time base of the capture timer is formed by counter 1, which counts down with the clock selected via the counter 1 clock selector. In Dual Input Capture mode, TAn and TBn pins function as capture inputs. A transition received on TAn pin causes a transfer of the timer contents to TnCRA register. Similarly, a transition received on the TBn pin causes a transfer of the timer contents to TnCRB register. TAn and TBn inputs can be configured to perform a counter preset to FFFFh on reception of a valid capture event. In this case, the current value of the counter is transferred to the corresponding capture register; following this, the counter is preset to FFFFh. Using this approach enables control over the on-time, off-time or cycle time of an external signal, thus reducing core overhead.

The pulse width of the input signal on TAn and TBn must be equal to or greater than one core clock cycle; see Section 8.5.8 on page 295 for additional details. The values captured in TnCRA register at different times reflect the elapsed time between transitions on the TAn pin. The same is true for TnCRB register and the TBn pin. Each input pin can be configured to sense either rising or falling edge transitions.

The timer can be configured to generate interrupts on reception of a transition on either TAn or TBn. The interrupts can be enabled or disabled separately for TAn or TBn by TnAIEN and TnBIEN bits, respectively. An underflow of TnCNT1 can also generate an interrupt if the interrupt was enabled by TnCIEN bit. All three interrupts have individual pending flags; see Section 4.8.5 on page 109 for detailed information.

TnCNT2 can be used as a "simple" system timer in this mode of operation. The TnCNT2 counter counts down with the clock selected via the counter 2 clock selector, and TnCNT2 can be configured to generate an interrupt on underflow if the interrupt was enabled by TnDIEN bit; see Section 4.8.5 on page 109 for detailed information.

Note that TnCNT1 cannot operate in the "Pulse Accumulate" or "External Event Counter" clock source modes of operation because TBn input is used as a capture input. Selecting either Pulse Accumulate mode or External Event Counter mode for TnCNT1 causes it to stop. However, all available clock source modes may be selected for TnCNT2. Thus it is possible to use TnCNT2 to determine the number of capture events on TBn or the elapsed time between capture events on TBn.

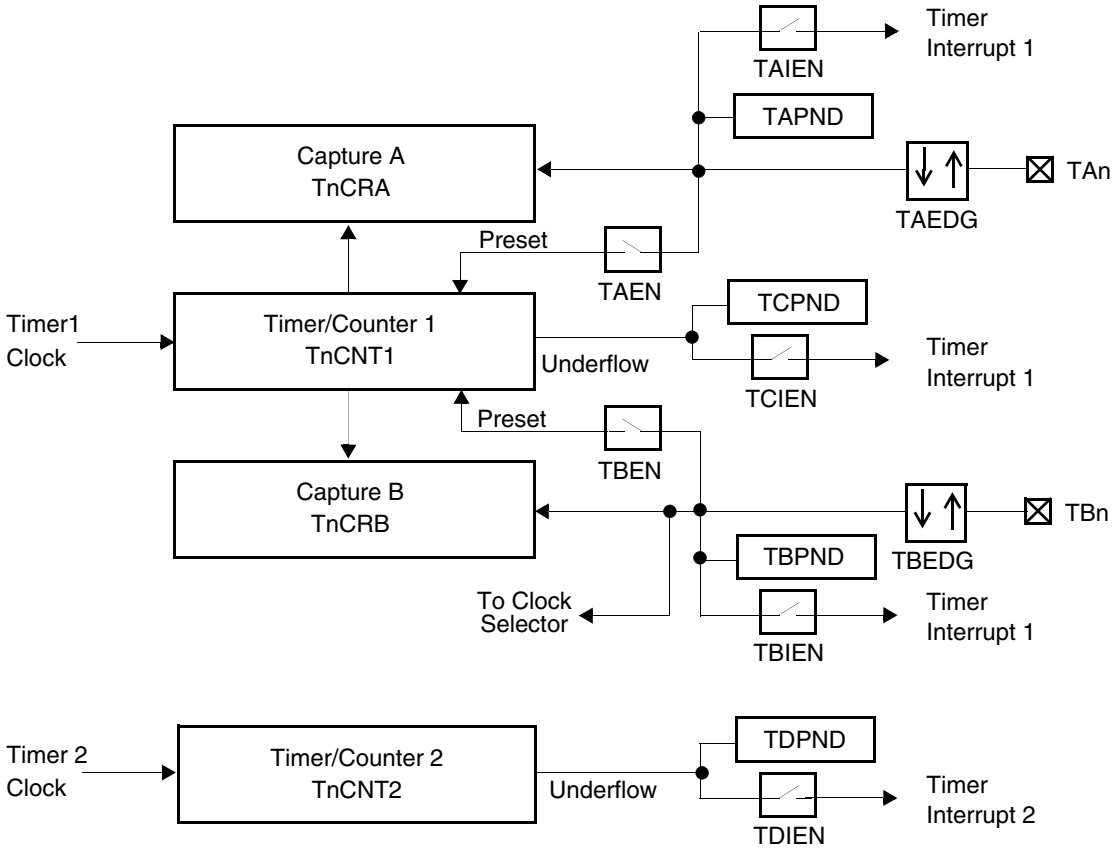


Figure 20. Mode 2, Dual Input Capture

4.0 Embedded Controller Modules (Continued)

Mode 3, Dual Independent Timer

Dual Independent Timer mode can be used for a wide variety of system tasks such as the periodic core interrupt generation, based either on the prescaled clock or external events on TBn. The timer can also toggle TAn pin on underflow, allowing the simple generation of a core-independent 50% duty cycle signal on TAn. In mode 3, TnCNT1 counts down and reloads from TnCRA on underflow while TnCNT2 is reloaded from TnCRB on underflow. Both counters can also be operated using the prescaled core clock. Figure 21 shows a block diagram of the timer in mode 3.

TnCNT1 counts down at the rate of the selected clock; see "Counter Clock Source Select" on page 103 for additional details. On underflow, TnCNT1 is reloaded from TnCRA register and counting proceeds. If enabled, the TAn pin toggles on underflow of TnCNT1. Software can select the initial value of the TAn output signal as either high or low; see "Timer I/O Functions" on page 110 for additional details. In addition, the TnAPND interrupt pending flag is set, and, if TnAIEN bit is set to 1, a timer interrupt 1 is generated; see Section 4.8.5 on page 109 for detailed information. Since TAn toggles on every underflow, a 50% duty cycle PWM signal can be generated on TAn without requiring interaction by the core.

TnCNT2 counts down at the rate of the selected clock; see "Counter Clock Source Select" on page 103 additional details. On every underflow of TnCNT2, the value contained in TnCRB register is loaded into TnCNT2, and counting proceeds downwards from that value. In addition, the TnDPND interrupt pending flag is set, and a timer interrupt 2 is generated if TnDIEN bit is set to 1; see Section 4.8.5 on page 109 for detailed information.

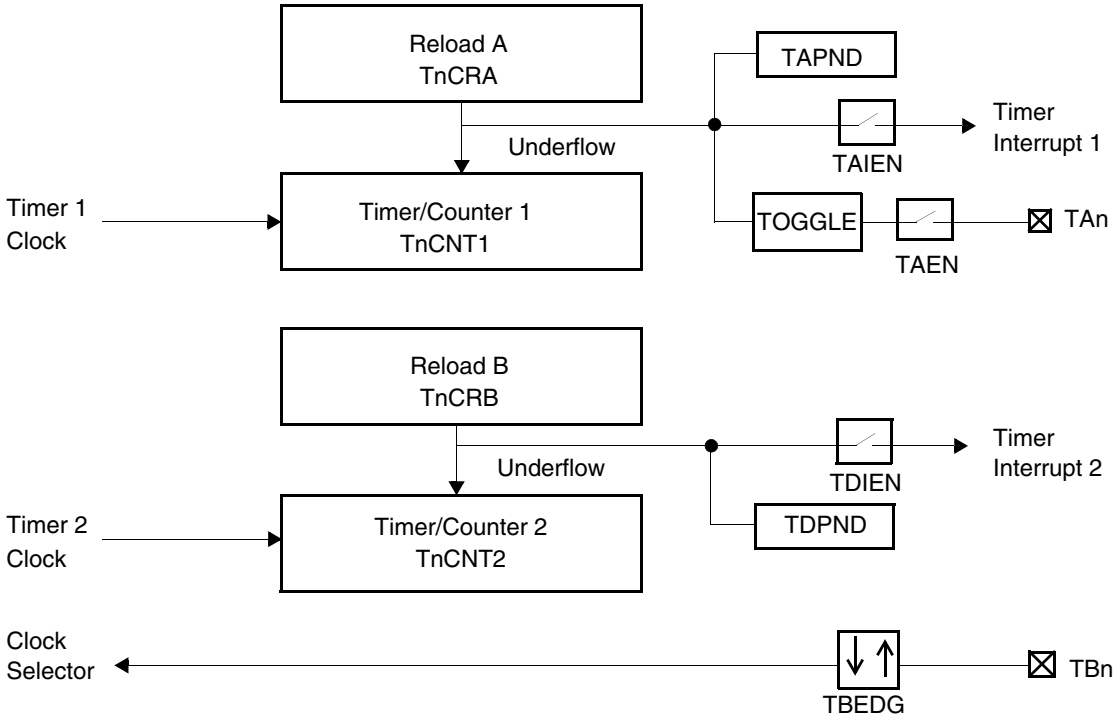


Figure 21. Mode 3, Dual Independent Timer

4.0 Embedded Controller Modules (Continued)

Mode 4, Input Capture and Timer

This operating mode offers a combination of a single timer (with automatic reload) and a single capture timer. In this mode, TnCNT1 operates as a timer that is reloaded from TnCRA on underflow. It is possible to toggle TA on every underflow of TnCNT1 and thus generate a 50% duty cycle signal on TAn. TnCNT2 forms the time base of the capture timer. The value on TnCNT2 is transferred to TnCRB when a valid event on TBn is detected.

This mode is a combination of modes 3 and 2. It allows TnCNT2 to operate as a single-input capture timer concurrently with TnCNT1. (TnCNT1 can be used as a system timer, as described in mode 3.) Figure 22 shows a block diagram of the timer in mode 4.

TnCNT1 starts counting down once a clock has been enabled. On underflow, TnCNT1 is reloaded from TnCRA register, and counting proceeds downwards from that value. If enabled, the TAn pin toggles on every underflow of TnCNT1. Software can select the initial value of the TAn output signal as either high or low; see Section 4.8.6 on page 110 for additional details. In addition, the TnAPND interrupt pending flag is set, and a timer interrupt 1 is generated if TnAIEN bit is set to 1; see Section 4.8.5 on page 109 for detailed information. Since TAn toggles on every underflow, a 50% duty cycle signal can be generated on TAn without requiring any interaction of software (i.e., without requiring any interaction with the core).

TnCNT2 starts counting down once a clock has been enabled. When a transition is received on TBn, the value contained in TnCNT2 is transferred to TnCRB, and the interrupt pending flag (TnBPND) is set. A timer interrupt 1 is generated if it is enabled. A preset of the counter to FFFFh on detection of a transition on TBn can be enabled. In this case, the current value of TnCNT2 is transferred to TnCRB, followed by a preset of the counter to FFFFh. TnCNT2 starts counting downwards from FFFFh until the next transition is received on TBn, which causes the procedure of capture and preset to be repeated. Underflow of TnCNT2 sets the TnDPND interrupt pending flag and can also generate a timer interrupt 2 if the interrupt was enabled; see Section 4.8.5 on page 109 for detailed information. The input signal on TBn must have a pulse width equal to or greater than one core clock cycle; see Section 8.5.8 on page 295 for additional details. TBn can be configured to sense either rising or negative edge transitions.

Notes:

- TnCNT2 cannot operate in the Pulse Accumulate or External Event Counter modes because TBn input is used as a capture input. Selecting either Pulse Accumulate mode or External Event Counter mode for TnCNT2 causes TnCNT2 to stop.
- All available clock source modes may be selected for TnCNT1. Thus it is possible to use TnCNT1 to determine the number of capture events on TBn or the elapsed time between capture events on TBn.

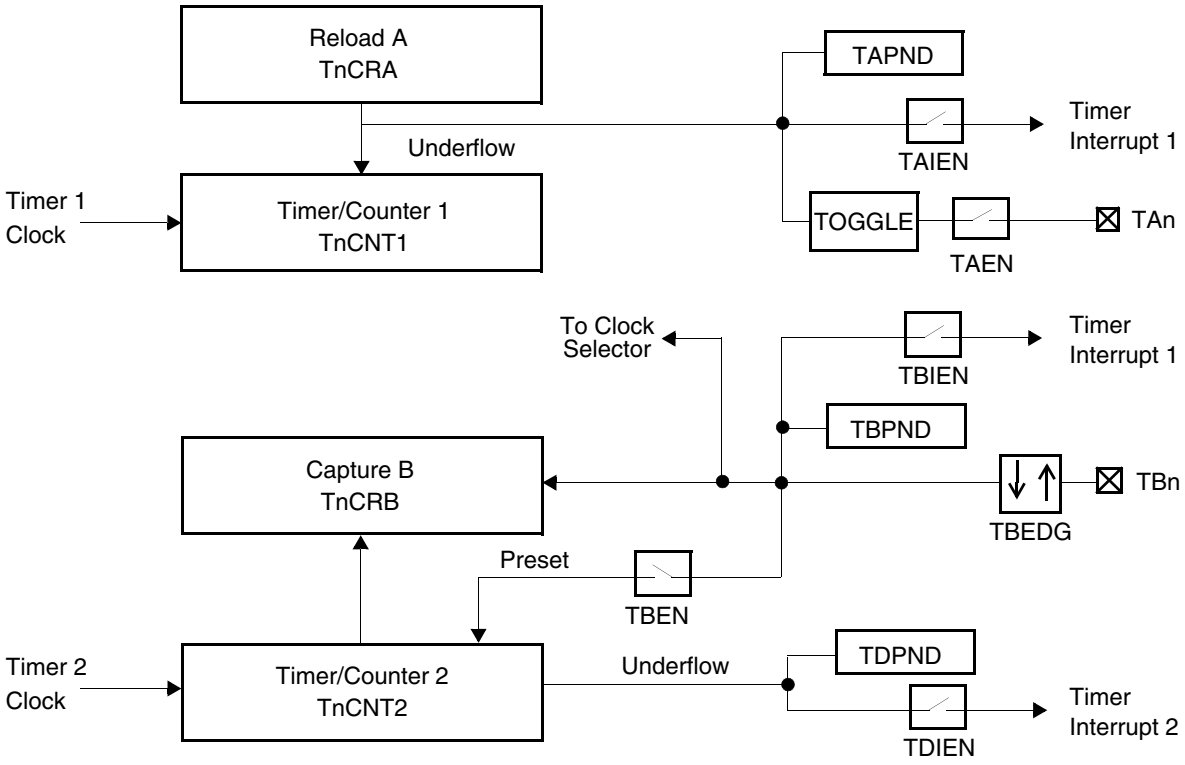


Figure 22. Mode 4, Input Capture and Timer

4.0 Embedded Controller Modules (Continued)

Mode 5, Dual Independent Input Capture

Dual Independent Input Capture mode can be used to measure either the frequency of two external clocks that are slower than the selected clock source frequency or the elapsed time between external events. A transition received on the TAn pin causes a transfer of TnCNT1 contents to TnCRA. A transition received on the TBn pin causes a transfer of TnCNT2 contents to TnCRB.

Figure 23 shows a block diagram of the timer operating in mode 5. In mode 5, the time base of each capture timer is formed by counters 1 and 2, which count down with the clock selected via the clock selectors. In Dual Independent Input Capture mode, TAn and TBn pins function as capture inputs. A transition received on TAn pin causes a transfer of timer 1 contents to TnCRA register. Similarly, a transition received on the TBn pin causes a transfer of timer 2 contents to TnCRB register. TAn and TBn inputs can be configured to perform counter preset to FFFFh on reception of a capture event. Each input presets the associated timer. In this case, the current value of the counter is transferred to the corresponding capture register; following this, the counter is preset to FFFFh. Using this approach enables control over the on-time, off-time or cycle time of an external signal, thus reducing core overhead.

The pulse width of the input signal on TAn and TBn must be equal to or greater than one core clock cycle; see Section 8.5.8 on page 295 for additional details. The values captured in TnCRA register at different times reflect the elapsed time between transitions on the TAn pin. The same applies for TnCRB register and the TBn pin. Each input pin can be configured to sense either rising or falling edge transitions.

The timer can be configured to generate interrupts on reception of a transition on either TAn or TBn. The interrupts can be enabled or disabled separately for TAn or TBn by TAnIEN and TnBIEN bits. An underflow of TnCNT1 or TnCNT2 can also generate an interrupt if the interrupt was enabled by TnCIEN or TnDIEN bit, respectively. All four interrupts have individual pending flags; see Section 4.8.5 on page 109 for detailed information.

Note that TnCNT1 and TnCNT2 cannot operate in the Pulse Accumulate or External Event Counter modes of operation because TBn input is used as a capture input. Selecting either Pulse Accumulate mode or External Event Counter mode for either counter causes it to stop.

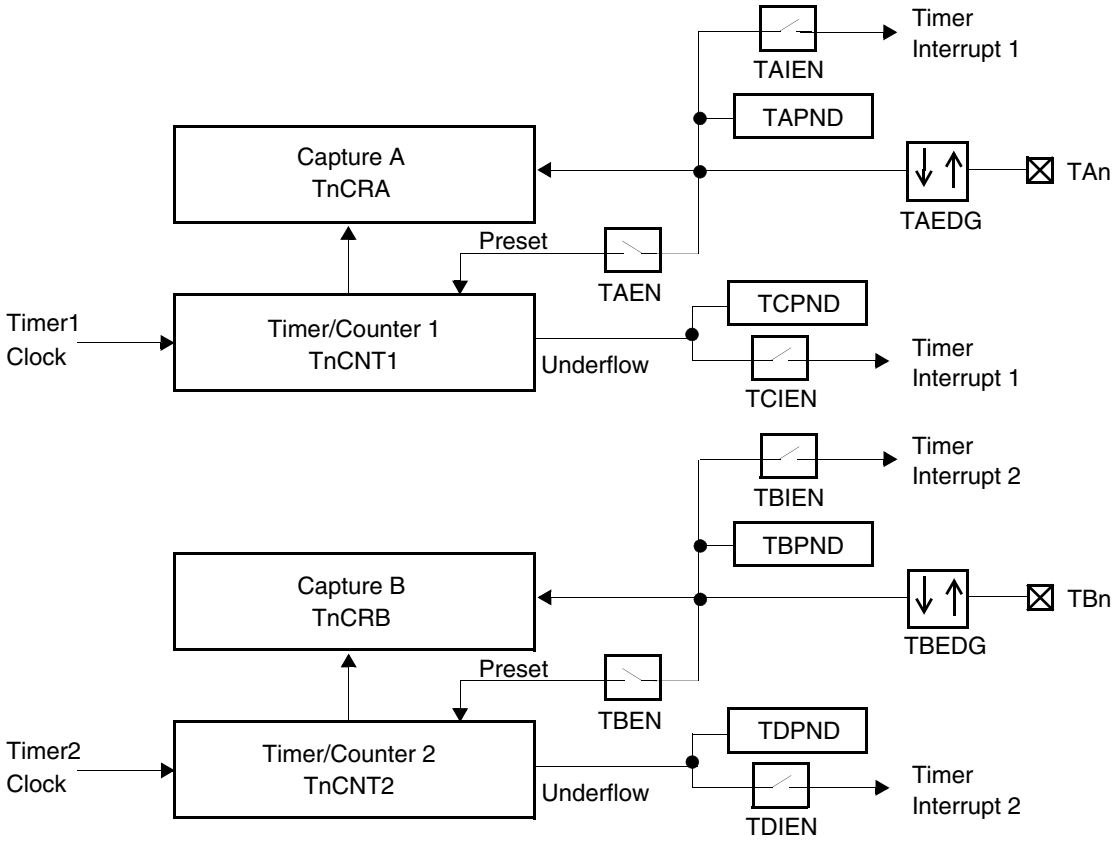


Figure 23. Mode 5, Dual Independent Input Capture

4.0 Embedded Controller Modules (Continued)

4.8.4 Compare Function

The MFT16 supports two compare functions. Compare function A can be assigned to Counter 1 or to Capture Register A. Compare function B can be assigned to Counter 2 or to Capture Register B. Each compare function compares a 16-bit value in the compare register TnCPA or TnCPB, with the value in the assigned counter or capture register. The compare module can be configured to generate an interrupt on any combination of three terms, 'equality', 'higher than' and 'lower than'. The comparison is done between the value in the assigned counter or capture register, and the value in the TnCPA/B register. The 'higher than' term for comparator A is true when TnCNT1 or TnCRA (depending on the configuration) is higher than TnCPA. The 'higher than' term for comparator B is true when TnCNT2 or TnCRB (depending on the configuration) is higher than TnCPB. When selecting the counter for comparison, the compare is performed every count. If the capture register is selected for compare, the comparison is performed only when a capture event is detected and after the capture register is updated.

Each compare function can generate an interrupt when one of the enabled terms is met during comparison, and if an interrupt is enabled for that compare function. Each compare function has its own pending flag, TnEPND or TnFPND.

The compare function is available in all MFT16 modes of operation and is independent of the state and mode of the timers. Note that no compare is performed if a compare function is assigned to a capture register and the capture event is disabled.

4.8.5 Timer Interrupts

The MFT16 contains a total of six interrupt sources that are mapped to two different system interrupts. Each source has a pending flag associated with it and can be enabled or disabled under software control to generate an interrupt. The pending flags are TnXPND, where n is the module and X is a letter from A to F. An interrupt enable flag, TnXIEN, is associated with each interrupt pending flag. Interrupt sources A, B, C or E can generate a timer interrupt 1; interrupt source D or F can generate a timer interrupt 2. Note that not all interrupt sources are available in all modes. Table 14 shows which events can trigger an interrupt in each mode of operation:

Table 14. MFT16 Interrupts

System Interrupt	Interrupt Pending Flag	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
		PWM and Counter	Dual Input Capture	Dual Independent Timer	Input Capture and Timer	Dual Independent Input Capture
Timer Int. 1	TnAPND	TnCNT1 reload from TnCRA	Input capture on TAn transition	TnCNT1 reload from TnCRA	TnCNT1 reload from TnCRA	Input capture on TAn transition
	TnBPND	TnCNT1 reload from TnCRB	Input Capture on TBn transition	N/A	Input Capture on TBn transition	Input capture on TBn transition
	TnCPND	N/A	TnCNT1 under-flow	N/A	N/A	TnCNT1 under-flow
	TnEPND	Compare Match for TnCNT1 or TnCRA	Compare Match for TnCNT1 or TnCRA	Compare Match for TnCNT1 or TnCRA	Compare Match for TnCNT1 or TnCRA	Compare Match for TnCNT1 or TnCRA
Timer Int. 2	TnDPND	TnCNT2 under-flow	TnCNT2 under-flow	TnCNT2 reload from TnCRB	TnCNT2 under-flow	TnCNT2 under-flow
	TnFPND	Compare Match for TnCNT2 or TnCRB	Compare Match for TnCNT2 or TnCRB	Compare Match for TnCNT2 or TnCRB	Compare Match for TnCNT2 or TnCRB	Compare Match for TnCNT2 or TnCRB

4.0 Embedded Controller Modules (Continued)

4.8.6 Timer I/O Functions

There are two I/O pins associated with each of the MFT16 modules: TAn and TBn, where n denotes the module on a given device. The functionality of TA and TB depends on the mode of operation and the value of TAEN and TBEN bits. Table 15 shows the function of TA and TB versus the selected mode of operation. Note that if TA functions as a PWM output, the initial and present value of TA is defined by TAOUT. For example, to start with TA high, TAOUT must be set (1) prior to enabling the timer clock.

Table 15. MFT16 I/O Functions

I/O	TnAEN TnBEN	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
		PWM and Counter	Dual Input Capture	Dual Independent Timer	Input Capture and Timer	Dual Independent Input Capture
TAn	TnAEN=0 TnBEN=X	Output = TnAOUT	Capture TnCNT1 into TnCRA	Output = TnAOUT	Output = TnAOUT	Capture TnCNT1 into TnCRA
	TnAEN=1 TnBEN=X	Toggle Output on underflow of TnCNT1	Capture TnCNT1 into TnCRA and preset TnCNT1	Toggle Output on underflow of TnCNT1	Toggle Output on underflow of TnCNT1	Capture TnCNT1 into TnCRA and preset TnCNT1
TBn	TnAEN=X TnBEN=0	Ext. Event or Pulse Accumulate Input	Capture TnCNT1 into TnCRB	Ext. Event or Pulse Accumulate Input	Capture TnCNT2 into TnCRB	Capture TnCNT2 into TnCRB
	TnAEN=X TnBEN=1	Ext. Event or Pulse Accumulate Input	Capture TnCNT1 into TnCRB and preset TnCNT1	Ext. Event or Pulse Accumulate Input	Capture TnCNT2 into TnCRB and preset TnCNT2	Capture TnCNT2 into TnCRB and preset TnCNT2

4.8.7 Operation in Development System

The MFT16 supports freezing the counters during breakpoints while operating in development systems. If FREEZE bit is asserted, all timer counter clocks are stopped and the current values of timer/counter registers TnCNT1 and TnCNT2 are frozen. Once FREEZE becomes inactive, counting resumes from the previous value; see Section 4.20.3 on page 187 for more details.

4.0 Embedded Controller Modules (Continued)

4.8.8 MFT16 Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

MFT16 Register Map

All MFT16 registers are powered by V_{CC}. All MFT16 registers are set to their default values by Core Domain reset unless otherwise specified in the register description below.

Location	Mnemonic	Register Name	Size	Type
MFT16-n Base + 00h	TnCNT1	Timer/Counter 1	Word	R/W
MFT16-n Base + 02h	TnCRA	Reload/Capture A	Word	R/W
MFT16-n Base + 04h	TnCRB	Reload/Capture B	Word	R/W
MFT16-n Base + 06h	TnCNT2	Timer/Counter 2	Word	R/W
MFT16-n Base + 08h	TnPRSC	Clock Prescaler	Byte	R/W
MFT16-n Base + 0Ah	TnCKC	Clock Unit Control	Byte	R/W
MFT16-n Base + 0Ch	TnMCTRL	Timer Mode Control	Byte	R/W
MFT16-n Base + 0Eh	TnICTRL	Timer Interrupt Control Register	Byte	R/1WS
MFT16-n Base + 10h	TnICLR	Timer Interrupt Clear	Byte	WO
MFT16-n Base + 12h	TnIEN	Timer Interrupt Enable	Byte	R/W
MFT16-n Base + 14h	TnCPA	Compare A	Word	R/W
MFT16-n Base + 16h	TnCPB	Compare B	Word	R/W
MFT16-n Base + 18h	TnCPCFG	Compare Configuration Register	Byte	R/W

Timer/Counter Register 1 (TnCNT1)

TnCNT1 is not affected by reset and thus contains random data on power-up.

Locations: MFT16-1 - FF F340h
MFT16-2 - FF F380h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TCNT1**															
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Reload/Capture Register A (TnCRA)

TnCRA is not affected by reset and thus contains random data on power-up.

Locations: MFT16-1 - FF F342h
MFT16-2 - FF F382h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TCRA**															
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

4.0 Embedded Controller Modules (Continued)

Reload/Capture Register B (TnCRB)

TnCRB is not affected by reset and thus contains random data on power-up.

Locations: MFT16-1 - FF F344h
MFT16-2 - FF F384h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TCRB**															
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Timer/Counter Register 2 (TnCNT2)

TnCNT2 is not affected by reset and thus contains random data on power-up.

Locations: MFT16-1 - FF F346h
MFT16-2 - FF F386h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TCNT2**															
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Clock Prescaler Register (TnPRSC)

TnPRSC defines the timer clock prescaler ratio. It is set to 00h on reset.

Locations: MFT16-1 - FF F348h
MFT16-2 - FF F388h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**CLKPS**				
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-5	**Reserved.**
4-0	**CLKPS (Clock Prescaler).** The timer clock is generated by dividing the core clock by CLKPS+1. Therefore, the maximum timer clock frequency is equal to the core clock (CLKPS=00000), and the minimum timer clock is the core clock divided by 32 (CLKPS=11111).

Clock Unit Control Register (TnCKC)

TnCKC defines the clock source selection for each timer counter. It is set to 00h on reset, thus disabling TnCNT1 and TnCNT2 clocks.

Locations: MFT16-1 - FF F34Ah
MFT16-2 - FF F38Ah

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**C2CSEL**			**C1CSEL**		
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-6	**Reserved.**

Bit	Description
5-3	**C2CSEL (Counter 2 Clock Select).** Defines the clock mode for TnCNT2. **Bits 5 4 3 Description** 0 0 0: No Clock (Counter 2 stopped and capture events for TnCNT2 are ignored) (default). 0 0 1: Prescaled core clock. 0 1 0: External event on TBn. 0 1 1: Pulse Accumulate. 1 0 0: Slow-speed clock (LFCLK). Other: Reserved.
2-0	**C1CSEL (Counter 1 Clock Select).** Defines the clock mode for TnCNT1: **Bits 2 1 0 Description** 0 0 0: No Clock (Counter 1 stopped and capture events for TnCNT1 are ignored) (default). 0 0 1: Prescaled core clock. 0 1 0: External event on TBn. 0 1 1: Pulse Accumulate. 1 0 0: Slow-speed clock (LFCLK). Other: Reserved.

Timer Mode Control Register (TnMCTRL)

TnMCTRL defines the operation mode of timer/counter and TAn and TBn I/O pins. It is set to 00h on reset.

Locations: MFT16-1 - FF F34Ch

MFT16-2 - FF F38Ch

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**TAOUT**	**TBEN**	**TAEN**	**TBEDG**	**TAEDG**	**MDSEL**		
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**TAOUT (TAn Output Data).** When TAn is used as a PWM output, in addition to the hardware control (which sets or resets the TAOUT bit on a toggle event), TAOUT allows software control over the level of TAn output. When read, TAOUT reflects the current value driven at the TAn output. TAOUT can also be used to set the initial value of TAn output in PWM mode. Note that if the hardware attempts to toggle this bit at the same time as the software is writing to it, the software write takes precedence over the hardware update. This bit has no effect on TAn pin, when TAn pin is used as input. 0: TAn is driven low (default). 1: TAn is driven high.
6	**TBEN (TBn Enable).** When set to 1, enables TBn to function as a preset input if operating in Dual Input Capture mode (mode 2), Input Capture and Timer mode (mode 4) or Dual Independent Input Capture mode (mode 5). The bit has no effect while operating in modes 1 or 3; see Table 15 on page 110 for additional information.
5	**TAEN (TAn Enable).** When set to 1, enables TAn to function either as a preset input or as a PWM output, depending on the mode of operation; see Table 15 on page 110 for additional information.
4	**TBEDG (TBn Edge Polarity).** 0: A high-to-low transition on TBn causes the action defined by the mode of operation, e.g., input capture or external event count (default). 1: A low-to-high transition on TBn results in the defined action. Note: In Pulse Accumulate mode, when this bit is set to 1, the count is enabled if TBn is high. When this bit is set to 0, the counter is enabled if TBn is low.

4.0 Embedded Controller Modules (Continued)

Bit	Description
3	**TAEDG (TAn Edge Polarity).** 0: A high-to-low transition on TAn causes the action defined by the mode of operation, e.g., input capture (default). 1: A low-to-high transition on TAn results in the defined action.
2-0	**MDSEL (Mode Select).** Defines the MFT16 mode of operation. **Bits 2 1 0 — Description** 0 0 0: Mode 1 (default). 0 0 1: Mode 2. 0 1 0: Mode 3. 0 1 1: Mode 4. 1 0 0: Mode 5. Other: Reserved.

Timer Interrupt Control Register (TnICTRL)

TnICTRL contains the interrupt pending bits for the timer interrupt sources. It is set to 00h on reset.

Locations: MFT16-1 - FF F34Eh

MFT16-2 - FF F38Eh

Type: R/W1S

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**TFPND**	**TEPND**	**TDPND**	**TCPND**	**TBPND**	**TAPND**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-6	**Reserved.**
5	**TFPND (Timer Interrupt Source F Pending).** When asserted, indicates that an interrupt condition (see Table 14 on page 109) occurred. Can be set by either hardware or software. This bit cannot be cleared (set to 0) directly. It can be cleared via the Timer Interrupt Clear register. A write of 0 to TFPND is ignored. The bit is cleared on reset.
4	**TEPND (Timer Interrupt Source E Pending).** Same as TFPND but for a different condition; see Table 14.
3	**TDPND (Timer Interrupt Source D Pending).** Same as TFPND but for a different condition; see Table 14.
2	**TCPND (Timer Interrupt Source C Pending).** Same as TFPND but for a different condition; see Table 14.
1	**TBPND (Timer Interrupt Source B Pending).** Same as TFPND but for a different condition; see Table 14.
0	**TAPND (Timer Interrupt Source A Pending).** Same as TFPND but for a different condition; see Table 14.

4.0 Embedded Controller Modules (Continued)

Timer Interrupt Clear Register (TnICLR)

TnICLR enables clearing the interrupt pending bits of TnICTRL register.

Locations: MFT16-1 - FF F350h
MFT16-2 - FF F390h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**TFCLR**	**TECLR**	**TDCLR**	**TCCLR**	**TBCLR**	**TACLR**

Bit	Description
7-6	**Reserved.**
5	**TFCLR (Timer Pending F Clear).** Writing a 1 clears the TFPND flag in TnICTRL register. 0: Has no effect on TFPND. The previous value of TFPND is maintained. 1: Causes the TFPND flag to be cleared (0).
4	**TECLR (Timer Pending E Clear).** Writing a 1 clears the TEPND flag in TnICTRL register. 0: Has no effect on TEPND. The previous value of TEPND is maintained. 1: Causes the TEPND flag to be cleared (0).
3	**TDCLR (Timer Pending D Clear).** Writing a 1 clears the TDPND flag in TnICTRL register. 0: Has no effect on TDPND. The previous value of TDPND is maintained. 1: Causes the TDPND flag to be cleared (0).
2	**TCCLR (Timer Pending C Clear).** Writing a 1 clears the TCPND flag in TnICTRL register. 0: Has no effect on TCPND. The previous value of TCPND is maintained. 1: Causes the TCPND flag to be cleared (0).
1	**TBCLR (Timer Pending B Clear).** Writing a 1 clears the TBPND flag in TnICTRL register. 0: Has no effect on TBPND. The previous value of TBPND is maintained. 1: Causes the TBPND flag to be cleared (0).
0	**TACLR (Timer Pending A Clear).** Writing a 1 clears the TAPND flag in TnICTRL register. 0: Has no effect on TAPND. The previous value of TAPND is maintained. 1: Causes the TAPND flag to be cleared (0).

Timer Interrupt Enable Register (TnIEN)

TnIEN contains the interrupt enable for the timer interrupt sources. It is cleared on reset.

Locations: MFT16-1 - FF F352h
MFT16-2 - FF F392h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**TFIEN**	**TEIEN**	**TDIEN**	**TCIEN**	**TBIEN**	**TAIEN**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-6	**Reserved.**
5	**TFIEN (Timer Interrupt F Enable).** 0: No system interrupt occurs, but the associated pending flag TFPND is set (default). 1: Enables a system interrupt based on the occurrence of a condition, as listed in Table 14 on page 109.
4	**TEIEN (Timer Interrupt E Enable).** 0: No system interrupt occurs, but the associated pending flag TEPND is set (default). 1: Enables a system interrupt based on the occurrence of a condition, as listed in Table 14

Bit	Description
3	**TDIEN (Timer Interrupt D Enable).** 0: No system interrupt occurs, but the associated pending flag TDPND is set (default). 1: Enables a system interrupt based on the occurrence of a condition, as listed in Table 14.
2	**TCIEN (Timer Interrupt C Enable).** 0: No system interrupt occurs, but the associated pending flag TCPND is set (default). 1: Enables a system interrupt based on the occurrence of a condition, as listed in Table 14.
1	**TBIEN (Timer Interrupt B Enable).** 0: No system interrupt occurs, but the associated pending flag TBPND is set (default). 1: Enables a system interrupt based on the occurrence of a condition, as listed in Table 14.
0	**TAIEN (Timer Interrupt A Enable)**. 0: No system interrupt occurs, but the associated pending flag TAPND is set (default). 1: Enables a system interrupt based on the occurrence of a condition, as listed in Table 14.

Compare Register A (TnCPA)

The value of TnCPA is compared with the value of TnCNT1 or TnCRA registers, depending on the CPASEL bit of the TnCPCFG register.

Locations: MFT16-1 - FF F354h

MFT16-2 - FF F394h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TCPA**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Compare Register B (TnCPB)

The value of TnCPB is compared with the value of TnCNT2 or TnCRB registers, depending on the CPBSEL bit of the TnCPCFG register.

Locations: MFT16-1 - FF F356h

MFT16-2 - FF F396h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TCPB**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4.0 Embedded Controller Modules (Continued)

Compare Configuration Register (TnCPCFG)

TnCPCFG contains configurations of the compare functions and interrupt generation. It is set to 00h on reset.

Locations: MFT16-1 - FF F358h
MFT16-2 - FF F398h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**HIBEN**	**EQBEN**	**LOBEN**	**CPBSEL**	**HIAEN**	**EQAEN**	**LOAEN**	**CPASEL**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**HIBEN (High Indication Enable for Comparator B)**. When set, enables the setting of TFPND flag in TnICTRL register when the TnCNT2 or TnCRB register value (depending on CPBSEL) is higher than the TnCPB value. 0: 'Higher than' comparison result has no effect on TFPND flag (default). 1: 'Higher than' comparison result sets TFPND flag.
6	**EQBEN (Equal Indication Enable for Comparator B)**. When set, enables the setting of TFPND flag in TnICTRL register when the TnCNT2 or TnCRB register value (depending on CPBSEL) is equal to the TnCPB value. 0: 'Equal' comparison result has no effect on TFPND flag (default). 1: 'Equal' comparison result sets TFPND flag.
5	**LOBEN (Low Indication Enable for Comparator B)**. When set, enables the setting of TFPND flag in TnICTRL register when the TnCNT2 or TnCRB register value (depending on CPBSEL) is lower than the TnCPB value. 0: 'Lower than' comparison result has no effect on TFPND flag (default). 1: 'Lower than' comparison result sets TFPND flag.
4	**CPBSEL (Compare Function B Select).** Compare function B configuration bit. 0: TnCPB register is compared with the value of counter 2 (TnCNT2 register). Compare is performed on every count (default). 1: TnCPB register is compared with the value of Capture register B (TnCRB register). Compare is performed only on capture events after the capture register is updated.
3	**HIAEN (High Indication Enable for Comparator A)**. When set, enables the setting of TEPND flag in TnICTRL register when the TnCNT1 or TnCRA register value (depending on CPASEL) is higher than the TnCPA value. 0: 'Higher than' comparison result has no effect on TEPND flag (default). 1: 'Higher than' comparison result sets TEPND flag.
2	**EQAEN (Equal Indication Enable for Comparator A)**. When set, enables the setting of TEPND flag in TnICTRL register when the TnCNT1 or TnCRA register value (depending on CPASEL) is equal to the TnCPA value. 0: 'Equal' comparison result has no effect on TEPND flag (default). 1: 'Equal' comparison result sets TEPND flag.
1	**LOAEN (Low Indication Enable for Comparator A)**. When set, enables the setting of TEPND flag in TnICTRL register when the TnCNT1 or TnCRA register value (depending on CPASEL) is lower than the TnCPA value. 0: 'Lower than' comparison result has no effect on TEPND flag (default). 1: 'Lower than' comparison result sets TEPND flag.
0	**CPASEL (Compare Function A Select).** Compare function A configuration bit. 0: TnCPA register is compared with the value of counter 1 (TnCNT1 register). Compare is performed on every count (default). 1: TnCPA register is compared with the value of Capture register A (TnCRA register). Compare is performed only on capture events after the capture register is updated.

4.0 Embedded Controller Modules (Continued)

4.9 PULSE WIDTH MODULATOR (PWM)

This chapter describes a PWM module. A device may include “n” such modules.

The PWM module generates “m” 16-bit PWM outputs; each may have a different duty cycle. A common 16-bit clock prescaler and a 16-bit down counter determine the cycle time, the minimal possible pulse width and the duty cycle steps.

The PWM module can be configured to provide up to 16-bit PWM resolution. The combined prescaler value (PRSCn) and down counter preset value (CTRn) determine the cycle time. The resolution is set by the value of CTRn, which determines the number of PWM steps per cycle.

The notation for the PWM output signals is “n_PWMm”; notation for the common registers is “*NAME*n”; notation for the duty cycle channel registers is “*NAME*mn”.

The WPC8763L includes two PWM modules, PWM A and PWM B (“n” is either A or B). The PWM A module has two outputs (“m” is from 0 to 1); the PWM B module has one output (“m” is 0). Note that the current state of A_PWM1 can be read via GPIO21, and the current state of B_PWM0 can be read via GPIO13.

4.9.1 Features

- “m” PWM outputs
- Common16-bit programmable prescaler
- Common16-bit programmable down counter
- 16-bit duty cycle control per output
- Programmable polarity per output
- Low power mode

4.9.2 Functional Description

The PWM generates m, 16-bit PWM outputs, n_PWM0 to n_PWMm. Duty Cycle registers 0 to m (DCRmn) control the duty cycle of n_PWM0 to n_PWMm output signals, respectively. The Cycle Time register (CTRn) controls the cycle time and duty cycle steps of all outputs.

The PWM uses the core clock as a reference for its activities. The prescaler divider divides the core clock by a programmable ratio of 1:1 through 1: 65536, as defined by the Clock Prescaler register (PRSCn).

The PWM output cycle period is CTR+1 cycles of the prescaler output clock. The 16-bit down counter counts, using the output clock from the prescaler divider, starting from the value held in CTRn register down to 0. When it reaches 0, the down counter restarts from the CTR value.

The DCRmn register defines the number of clock cycles in which n_PWMm signal is high during the down-counter cycle.

If Inverse PWMm bit (INVPm) in PWM Control register (PWMCTLn) is active (1), the n_PWMm output signal is inverted, regardless of the DCRmn or CTRn values of the down counter.

Figure 24 is a functional block diagram of the PWM module.

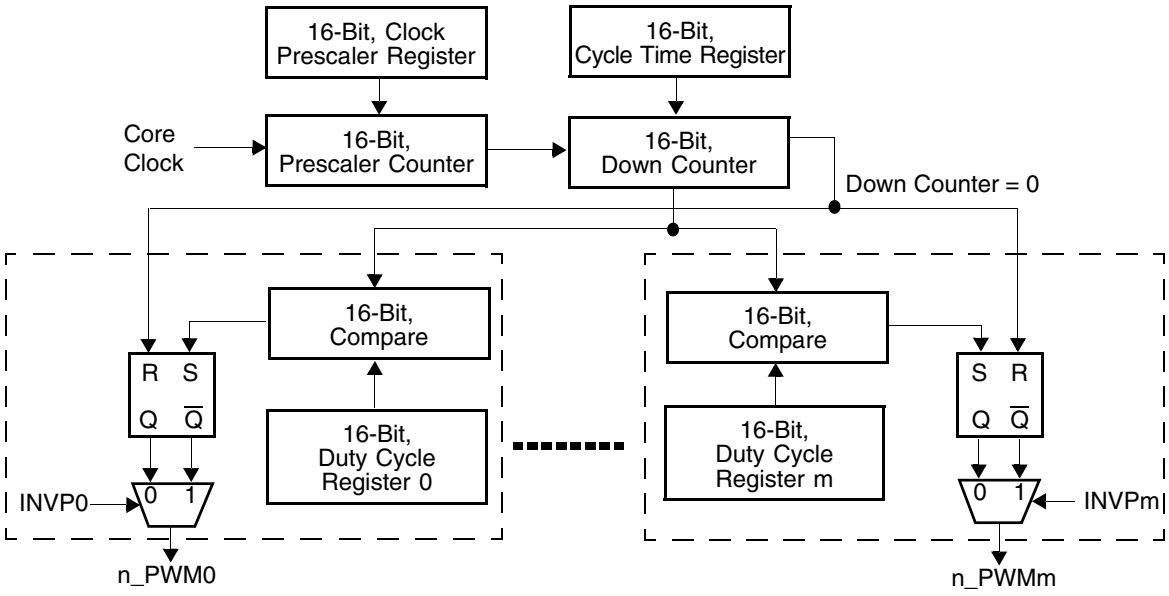


Figure 24. PWM Block Diagram

4.0 Embedded Controller Modules (Continued)

4.9.3 Cycle Time and Duty Cycle Calculation

The PWM module supports duty cycles in the range of 0% to 100%.

The n_PWMm output signal cycle time is: **(PRSCn + 1) x (CTRn + 1) x T_{CLK}** where:

- T_{CLK} is the core clock cycle time (i.e., the PWM input clock).
- The cycle time may range from 2 x T_{CLK} to 65536 x 65536 x T_{CLK}.

The n_PWMm output signal duty cycle (in %, when INVPm is 0) is: **((DCRmn + 1) / (CTRn + 1)) x 100**

Special cases:

- If the DCRmn value is greater than the CTRn value, the n_PWMm signal is always low.
- If DCRmn value is equal to the CTRn value, the n_PWMm signal is always high.

INVPm bit may be used to invert the n_PWMm signal.

4.9.4 Power Modes

The PWM is in low power mode when Power Mode bit (PWR) in PWM Control Register (PWMCTLn) is 0. In this mode, the PWM input clock is disabled (stopped), but the registers are accessible and maintained. The n_PWMm signal is 0 when INVPm bit is 0; it is 1 when INVPm bit is 1.

The PWM is in normal power mode when PWR bit in PWMCTLn register is 1. In this mode, the PWM module is enabled, its registers are accessible and its clock is functional.

The PRSCn and CTRn registers should be updated during Low Power mode. Otherwise, an unpredictable transient may be generated on n_PWMm output signals.

4.9.5 PWM Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

PWM Register Map

All PWM registers are powered by V_{CC} and are reset by Core Domain reset.

Location	Mnemonic	Register Name	Size	Type
n_PWM Base + 00h	PRSCn	Clock Prescaler	Word	R/W
n_PWM Base + 02h	CTRn	Cycle Time	Word	R/W
n_PWM Base + 04h	PWMCTLn	PWM Control	Byte	R/W
n_PWM Base + 06h	DCR0n	Duty Cycle Channel 0	Word	R/W
n_PWM Base + 08h	DCR1n	Duty Cycle Channel 1	Word	R/W

Clock Prescaler Register (PRSCn)

PRSCn controls the cycle time and the minimal pulse width. It is set to 0000h on reset.

Locations: A_PWM - FF F440h
B_PWM - FF F480h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**PRSC15-0**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-0	**PRSC (Prescaler Divider Value).** The divider of the Input Clock is the number defined by PRSC15-0 + 1. For example, a value of 0000h results in a divide by 1; a value of FFFFh results in a divide by 65536. The contents of this register should be changed only when the PWM module is in low power mode. Otherwise, there may be unpredictable results.

4.0 Embedded Controller Modules (Continued)

Cycle Time Register (CTRn)

CTRn controls the cycle time and duty cycle steps. It is set to FFFFh on reset.

Locations: A_PWM - FF F442h

B_PWM - FF F482h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**CTR15-0**															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Description
15-0	**CTR (Cycle Time Value).** The 16-bit down counter divides the prescaler output clock by CTR + 1. For example, a value of 0000h results in a divide by 1; a value of FFFFh results in a divide by 65536. The contents of this register should be changed only when the PWM module is in low power mode. Otherwise, there may be unpredictable results.

PWM Control Register (PWMCTLn)

PWMCTLn controls the PWM power mode and the polarity of n_PWM0 to n_PWMm outputs. It is set to 00h on reset.

Locations: A_PWM - FF F444h

B_PWM - FF F484h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**PWR**	**Reserved**					**INVP1-0**	
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**PWR (Power Mode).** Controls the operating mode of the PWM. 0: Low Power mode. PWM input clock is disabled (stopped). PWM registers are accessible and maintained. n_PWMm signal is 0 when INVPm bit is 0; n_PWMm signal is 1 when INVPm bit is 1 (default). 1: Normal Power mode. PWM module is enabled. Its registers are accessible and its clock is functional.
6-2	**Reserved.**
1-0	**INVP1-0 (Inverse PWM Outputs).** Each bit controls the corresponding polarity of n_PWM0 to n_PWM1; e.g., INVP0 controls n_PWM0. Note that INVP1 is not supported in B_PWM module. 0: The n_PWMm output signal is not inverted (default). 1: The n_PWMm output signal is inverted.

4.0 Embedded Controller Modules (Continued)

Duty Cycle Registers 0 to 1 (DCRmn)

DCRmn (m = 0 to 1) control the duty cycle of n_PWMm output signal. They are set to 0000h on reset.

Locations: A_PWM Channel 0 - FF F446h
Channel 1 - FF F448h
B_PWM Channel 0 - FF F486h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**DCR15-0**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-0	**DCR (Duty Cycle Value).** DCR field defines the number of clocks in which the n_PWMm signal is high out of the n_PWMm total cycle time. When INVPm bit of PWMCTLn register is set to 1 this behavior is inverted. The n_PWMm output signal duty cycle (in %, when INVPm is 0) is: **`((DCRmn + 1) / (CTRn + 1)) x 100`** For Special cases, see Section 4.9.3 on page 119.

4.0 Embedded Controller Modules (Continued)

4.10 CORE UNIVERSAL ASYNCHRONOUS RECEIVER-TRANSMITTER (CR_UART)

The CR_UART is a full-duplex asynchronous receiver-transmitter that supports a wide range of software programmable baud rates and data formats. It handles automatic parity generation and provides several error detection schemes.

4.10.1 Features

- Full-duplex, double-buffered receiver-transmitter
- Asynchronous operation
- Programmable baud rate
- Numerous framing formats
  - — seven, eight or nine data bits
  - — one or two stop bits
  - — odd, even, mark, space or no parity
- Hardware support of parity-bit generation during transmission and parity-bit check during reception
- Interrupt on transmit buffer empty, receive buffer full or receive error conditions (each with separate enable bits)
- Software-controlled break transmission and detection
- Internal diagnostic capability
- Automatic error detection
  - — Parity Error
  - — Framing Error
  - — Data Overrun Error
- 9-bit Attention mode

4.10.2 Functional Overview

Figure 25 shows the CR_UART block diagram. The CR_UART is composed of the following functional units:

- Transmitter
- Receiver
- Baud rate generator
- Control and error detection

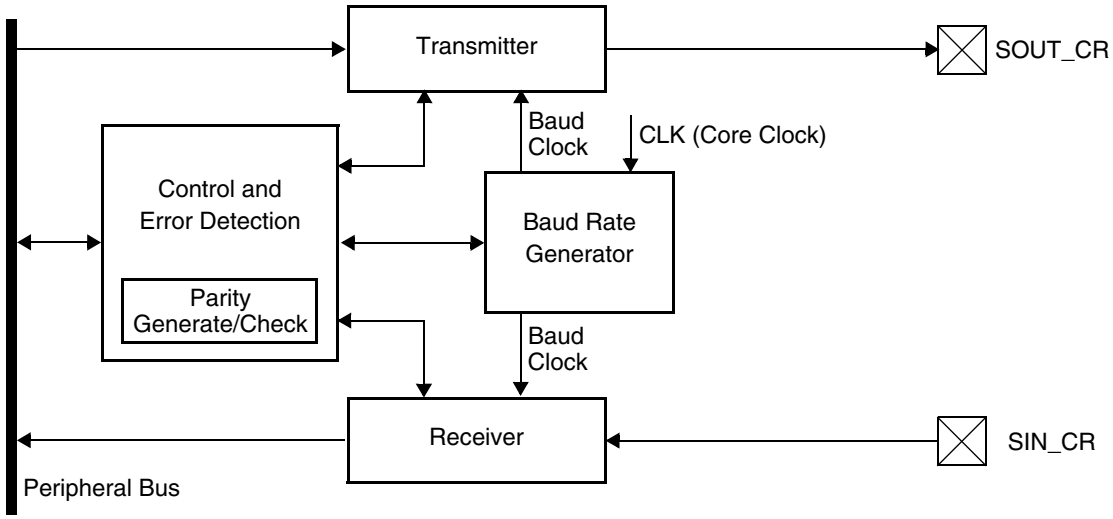


Figure 25. CR_UART Block Diagram

Transmitter

The Transmitter consists of an 8-bit Transmit Shift register and an 8-bit Transmit buffer. Data is loaded in parallel from the buffer into the shift register and then transmitted serially on the SOUT_CR pin.

Receiver

The Receiver consists of an 8-bit Receive Shift register and an 8-bit Receive buffer. Data is received serially into the shift register from the SIN_CR pin. After eight bits have been received, the contents of the shift register are transferred in parallel to the Receive buffer.

Baud Rate Generator

The Baud Rate Generator generates the baud clock. It consists of two registers and a two-stage counter. The registers are used to specify a prescaler value and baud rate divisor. The first stage of the counter divides the CLK clock in 0.5 increments based on the value of the prescaler. The second stage of the counter divides the output of the first stage in integer increments based on the value of the baud rate divisor.

Control and Error Detection

This function contains the control registers, control logic, error detection circuitry and parity generation/check. It supports:

- Selection of the data format, mode of operation and parity type
- Generation and detection of parity bit
- Reporting of parity errors
- Detection and reporting of data overrun and frame errors
- Interrupts on transmit buffer empty, receive buffer full, and receive error conditions
- Generation and detection of line breaks

4.10.3 Functional Description

Operation of the CR_UART includes Normal mode and two special modes (Attention and Diagnostic).

Normal Mode

CR_UART Normal mode enables the WPC8763L to communicate with other devices using two communication signals: transmit (SOUT_CR) and receive (SIN_CR).

In Normal mode, Transmit Shift register (TSFT) and Transmit Buffer register (UTBUF) double buffer the data for transmission. To transmit a character, a data byte is loaded into UTBUF register (a read/write register). Then the data is transferred to TSFT register. While TSFT register is shifting out the current character (LSB first) via the SOUT_CR pin, UTBUF register is loaded by software with the next byte to be transmitted. When TSFT completes transmitting the last stop bit of the current frame, the contents of UTBUF are transferred to TSFT register and the Transmit Buffer Empty flag (TBE) is set. The TBE flag is automatically reset by the CR_UART when the software loads a new character into UTBUF register. During transmission, the XMIP bit is set high by the CR_UART. This bit is reset only after the CR_UART has sent the last stop bit of the current character **and** UTBUF register is empty. TSFT register is not user accessible.

In Normal mode, the baud rate generator is always used as the CR_UART clock, and the input frequency to the CR_UART is 16 times the baud rate (i.e., there are 16 clock cycles per bit time).

Receive Shift register (RSFT) and Receive Buffer register (URBUF) double buffer the data being received. The CR_UART receiver continually monitors the signal on the SIN_CR pin for a low level to detect the beginning of a start bit. On sensing this low level, the CR_UART waits for seven input clock cycles and samples SIN_CR three times. If all three samples indicate a low level, the receiver considers this as a valid start bit, and the remaining bits in the character frame are each sampled three times around the mid-bit position. For any bit following the start bit, the logic value is found by majority voting, i.e., the two samples with the same value define the value of the data bit. Figure 26 shows the process of start-bit detection and bit sampling. Serial data input at the SIN_CR pin is shifted into RSFT register. After the complete character is received, the contents of RSFT register are copied into URBUF register, and the Receive Buffer Full flag (RBF) is set. The RBF flag is automatically reset when software reads the character from URBUF register. The RSFT register is not user accessible.

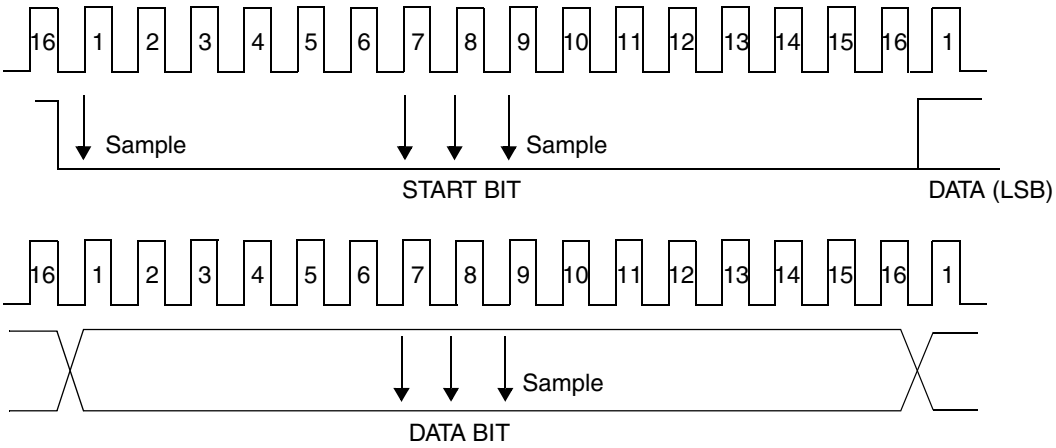


Figure 26. CR_UART Normal Mode

4.0 Embedded Controller Modules (Continued)

Attention Mode

Attention mode is available for networking the WPC8763L with other processors. This mode requires 9-bit data format with no parity bit. The number of start/stop bits are user selectable. In this mode, two types of 9-bit characters are sent on the network: address characters consisting of eight address bits and a '1' in the ninth bit position and data characters consisting of eight data bits and a '0' in the ninth bit position.

While in Attention mode, the CR_UART receiver monitors the communication flow but ignores all characters until an address character is received. When the address character is received, the contents of Receive Shift register are copied to the receive buffer. The RBF flag is set and an interrupt (if enabled) is generated. The ATN bit is automatically reset to zero, and the CR_UART begins receiving all subsequent characters. The software must examine the contents of URBUF register and respond by accepting the subsequent characters (by leaving the ATN bit reset) or waiting for the next address character (by setting ATN bit again).

Except for what has been described above, operation of the CR_UART transmitter is not affected by the selection of this mode. The value of the ninth bit to be transmitted is programed by setting the STPXB9 bit accordingly. The value of the ninth bit received is read from RB9 bit.

Diagnostic Mode

This mode is used for diagnostic tests of the CR_UART. In this mode, the SOUT_CR and SIN_CR pins are internally connected, and data that is shifted out of Transmit Shift register is immediately transferred to Receive Shift register. This mode supports only 9-bit data format with no parity bit. The number of start and stop bits is user selectable.

Frame Format Selection

The format shown in Figure 27 consists of one start bit, seven data bits (excluding parity) and one or two stop bits. If parity-bit generation is enabled (by setting the PEN bit), a parity bit is generated and transmitted following the seven data bits.

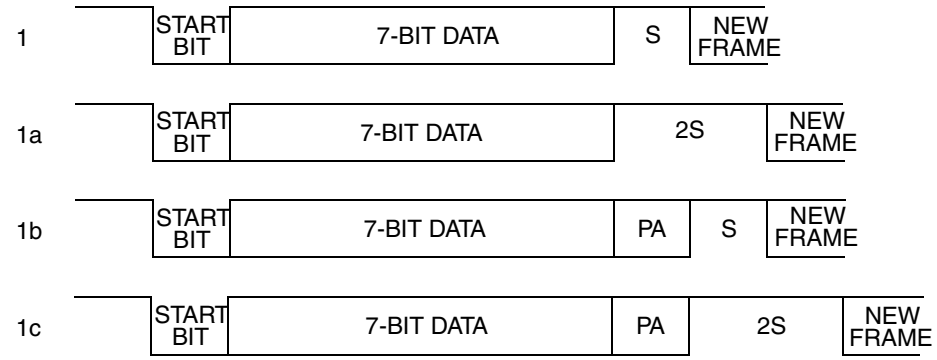


Figure 27. Seven Data Bit Frame Options

The format shown in Figure 28 consists of one start bit, eight data bits (excluding parity) and one or two stop bits. If parity bit generation is enabled by setting the PEN bit, a parity bit is generated and transmitted following the eight data bits.

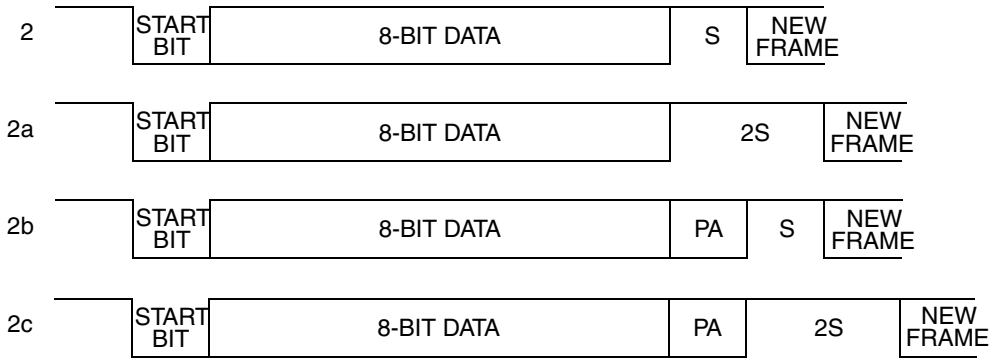


Figure 28. Eight Data Bit Frame Options

4.0 Embedded Controller Modules (Continued)

The format shown in Figure 29 consists of one start bit, nine data bits and one or two stop bits. This format also supports CR_UART Attention mode. When operating in this format, all eight bits of UTBUF and URBUF are used for data. The ninth data bit is transmitted and received using two bits in the control registers, STPXB9 and RB9. Parity is not generated or checked in this mode.

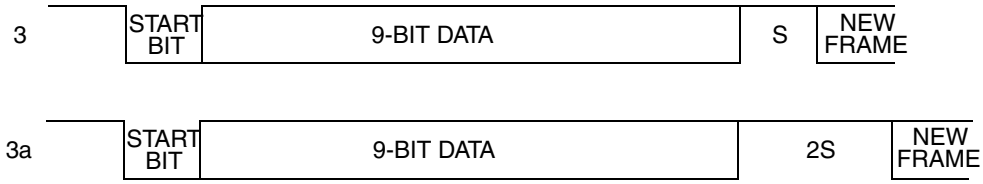


Figure 29. Nine Data Bit Frame Options

Baud Rate Generator

The Baud Rate Generator provides the basic baud clock from the core clock (CLK). The core clock passes through a two-stage divider chain consisting of a 5-bit baud rate prescaler (UPSC) and an 11-bit baud rate divisor (UDIV10-0).

The correspondences between the 5-bit prescaler select (UPSC) and prescaler divide factors are shown in Table 16.

Table 16. Prescaler Factors

UPSC	Prescaler Factor	UPSC	Prescaler Factor
00000	NO CLOCK	10000	8.5
00001	1	10001	9
00010	1.5	10010	9.5
00011	2	10011	10
00100	2.5	10100	10.5
00101	3	10101	11
00110	3.5	10110	11.5
00111	4	10111	12
01000	4.5	11000	12.5
01001	5	11001	13
01010	5.5	11010	13.5
01011	6	11011	14
01100	6.5	11100	14.5
01101	7	11101	15
01110	7.5	11110	15.5
01111	8	11111	16

A prescaler factor of zero corresponds to “NO CLOCK”. The “NO CLOCK” condition is the CR_UART Power-Down mode. In this mode, the CR_UART clock is turned off in order to reduce power consumption. The user should set the prescaler factor to zero (“NO CLOCK”) before selecting a new baud rate. Changing the baud rate while the USART is in operation might cause incorrect data to be received or transmitted.

The baud rate is calculated by:

```
BR = CLK/(16 x DIV x P)
```

where:

- BR is the baud rate.
- CLK is the core clock.
- DIV = UDIV10-0 + 1 (see “Baud Rate Divisor Register (UBAUD)” on page 131).
- P is the “Prescaler Factor” selected by UPSC (see Table 16 on page 125).

The baud rate clock is 16 times the baud rate.

4.0 Embedded Controller Modules (Continued)

Interrupts

The CR_UART is capable of generating interrupts on one of the following conditions:

- Receive Buffer Full
- Receive Error
- Transmit Buffer Empty

Figure 30 shows a diagram of the interrupt sources and the associated enable bits.

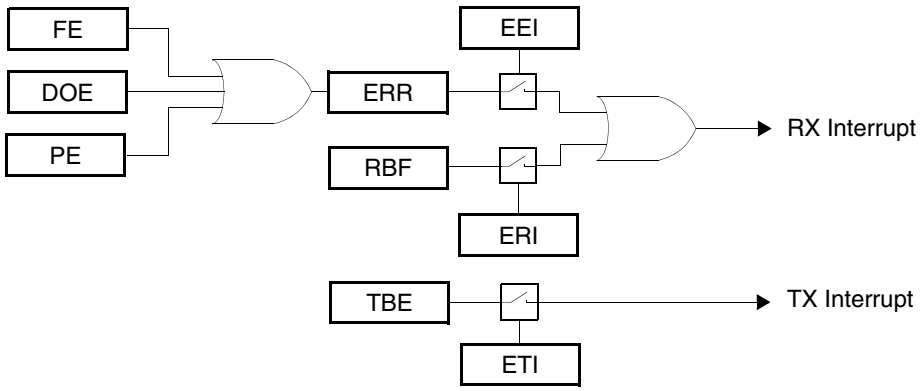


Figure 30. CR_UART Interrupt Routing

The interrupts can be individually enabled or disabled using Enable Transmit Interrupt (ETI), Enable Receive Interrupt (ERI) and Enable Receive Error Interrupt (EEI) bits in UICTRL register.

A transmit interrupt is generated when both the TBE and ETI bits are set. To remove this interrupt request, the software must either disable the interrupt by clearing the ETI bit or write to UTBUF register (thus clearing the TBE bit).

A receive interrupt is generated on two conditions:

- If both the RBF and ERI bits are set. To remove this interrupt request, the software must either disable the interrupt (by clearing the ERI bit) or read from URBUF register (thus clearing the RBF bit).
- If both the ERR and the EEI bits are set. To remove this interrupt request, the software must either disable it (by clearing the EEI bit) or read USTAT register, which causes ERR flag to be cleared.

Break Generation and Detection

A line break is generated when BRK bit is set in MDSL register. The SOUT_CR line remains low until the user resets the BRK bit.

A line break is detected if SIN_CR remains low for a time equivalent to 10 bit times or longer, after a missing stop bit has been detected.

Parity Generation and Detection

Parity is only generated or checked with 7- and 8-bit data formats. It is not generated or checked in Diagnostic Loopback mode, Attention mode or in Normal mode with 9-bit data format. Parity generation and checking is enabled and disabled via PEN bit in UFRS register. PSEL field in UFRS register is used to select odd, even, mark or space parity.

Freeze Mode

The CR_UART module supports breakpoint operation by preserving some of the status bits of the USTAT and UICTRL registers. While the UARTFEN bit in DBGFRZEN1 register (on page 188) is set to 1, the PE, FE, DOE and BKD bits are not cleared by a read from USTAT register, and RBF bit is not cleared by a read from URBUF register.

4.0 Embedded Controller Modules (Continued)

4.10.4 CR_UART Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20. All CR_UART registers are powered by V_{CC} and are reset by Core Domain reset unless otherwise specified in the register description below.

CR_UART Register Map

Location	Mnemonic	Register Name	Size	Type
FF F680h	UTBUF	Transmit Data Buffer	Byte	R/W
FF F682h	URBUF	Receive Data Buffer	Byte	RO
FF F684h	UICTRL	Interrupt Control	Byte	Varies per bit
FF F686h	USTAT	Status	Byte	RO
FF F688h	UFRS	Frame Select	Byte	R/W
FF F68Ah	UMDSL	Mode Select	Byte	R/W
FF F68Ch	UBAUD	Baud Rate Divisor	Byte	R/W
FF F68Eh	UPSR	Baud Rate Prescaler	Byte	R/W

Transmit Data Buffer Register (UTBUF)

Location: FF F680h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**UTBUF**							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**UTBUF.** Holds the data byte to be transmitted.

Receive Data Buffer Register (URBUF)

Location: FF F682h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**URBUF**							
Reset	X	X	X	X	X	X	X	X

Bit	Description
7-0	**URBUF.** Holds the received data byte.

Interrupt Control Register (UICTRL)

Contains the interrupt enable bits and the interrupt status flags.The register is set to 01h on reset.

Location: FF F684h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**EEI**	**ERI**	**ETI**	**Reserved**			**RBF**	**TBE**
Reset	0	0	0	0	0	0	0	1

Bit	Type	Description
7	R/W	**EEI (Enable Receive Error Interrupt).** An RX interrupt is generated when the ERR flag is set, indicating that a receive error occurred. 0: Disable receive error interrupt (default). 1: Enable receive error interrupt.
6	R/W	**ERI (Enable Receive Interrupt).** An RX interrupt is generated when the RBF flag is set. 0: Disable receive interrupt (default). 1: Enable receive interrupt.
5	R/W	**ETI (Enable Transmit Interrupt).** A TX interrupt is generated when the TBE flag is set. 0: Disable transmit interrupt (default). 1: Enable transmit interrupt.
4-2		**Reserved.**
1	RO	**RBF (Receive Buffer Full).** The bit is set by the hardware when the CR_UART has received a complete data frame and has transferred the data from RSFT register to URBUF register. The bit is automatically cleared when RBUF register is read. 0: Receive buffer not full; new data has not been transferred to URBUF since the last time it was read (default). 1: Receive buffer full; URBUF contains new data.
0	RO	**TBE (Transmit Buffer Empty).** The bit is set by the hardware when the CR_UART transfers data from UTBUF register to TSFT register for transmission. It is automatically cleared on the next write to UTBUF register. The bit is set on reset. 0: Transmit buffer not empty. 1: Transmit buffer empty (default).

Status Register (USTAT)

Contains the receive and transmit status bits. The register is cleared to 00h on reset.

Location: FF F686h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**XMIP**	**RB9**	**BKD**	**ERR**	**DOE**	**FE**	**PE**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7		**Reserved.**
6	RO	**XMIP (Transmit).** Indicates that the CR_UART is transmitting data. It is reset by hardware at the end of the last frame bit. 0: CR_UART is not transmitting (default). 1: CR_UART is transmitting.
5	RO	**RB9 (Received Bit 9).** Contains the ninth data bit of the last frame received when operating with the 9-bit data format. 0: '0' received in ninth bit position (default). 1: '1' received in ninth bit position.

Bit	Type	Description
4	RO	**BKD (Break Detect).** If set, indicates that a line break condition has occurred. A break condition is detected if SIN_CR remains low for a least ten bit times after a missing stop bit has been detected at the end of a frame. The bit is cleared under the following conditions: – On a read of USTAT register, if the break condition on SIN_CR is no longer present. – If the read of USTAT register did not cause the bit to be cleared because the break condition on SIN_CR was still in effect, the hardware clears the bit as soon as the break condition no longer exists (i.e., SIN_CR returns to a high level). 0: No break condition on SIN_CR (default). 1: A break condition detected on SIN_CR.
3	RO	**ERR (Receive Error).** Is set when DOE, FE or PE is set. It is cleared if DOE, FE and PE are all zero. 0: No DOE, FE or PE error has occurred, since the last time USTAT register was read (default). 1: A DOE, FE or PE error has occurred, since the last time USTAT register was read.
2	RO	**DOE (Data Overrun Error).** If no framing error was detected (i.e., FE bit is 0), DOE bit is set when a new character is received and transferred to RBUF before the software has read the previous character. The bit is cleared by the hardware when USTAT register is read. 0: No data overrun error detected (default). 1: Data overrun error detected since the last time USTAT was read.
1	RO	**FE (Framing Error).** Is set when the CR_UART does not receive a valid stop bit at the end of a frame. The bit is cleared by the hardware when USTAT register is read. 0: No framing error detected (default). 1: Framing error detected on a received byte since the last time USTAT was read.
0	RO	**PE (Parity Error).** If no framing error was detected (i.e., FE bit is 0), PE bit is set when a parity error is detected within a received character. The bit is cleared by the hardware when USTAT register is read. 0: No parity error detected (default). 1: Parity error detected in a received byte since the last time USTAT was read.

Frame Select Register (UFRS)

Controls the selection of the frame format, including number of data bits, number of stop bits and parity. The register is cleared to 00h on reset.

Location: FF F688h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**PEN**	**PSEL**		**XB9**	**STP**	**CHAR**	
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7		**Reserved.**
6	R/W	**PEN (Parity Enable).** Enables or disables the generation of a parity bit generation and parity check. Note that there is no parity bit when operating in the nine data bits-per-frame mode. In this case, PEN bit has no effect. 0: Parity disabled (default). 1: Parity enabled.

Bit	Type	Description
5-4	R/W	**PSEL (Parity Select).** Controls the mode of parity bit generation and checking. Note that there is no parity bit when operating in the nine data bits-per-frame mode. In this case, PSEL field has no effect. **Bits 5 4 Description** 0 0: Odd parity (default) 0 1: Even parity 1 0: Mark (1) 1 1: Space (0)
3	R/W	**XB9 (Transmitted Bit 9).** Contains the value of the 9th data bit for transmission only. The bit has no effect while operating with seven or eight data bits per frame. 0: Transmit '0' as 9th data bit (default). 1: Transmit '1' as 9th data bit.
2	R/W	**STP (Stop Bits).** Programs the number of stop bits to be transmitted. 0: One stop bit transmitted (default). 1: Two stop bits transmitted.
1-0	R/W	**CHAR (Character Length).** Selects the number of data bits per frame. Note that the parity bit is not included in the number of data bits. **Bits 1 0 Description** 0 0: Frame contains eight data bits (default) 0 1: Frame contains seven data bits 1 0: Frame contains nine data bits 1 1: Loopback mode selected; frame contains nine data bits.

Mode Select Register (UMDSL)

Controls the Attention mode and line break generation. The register is cleared to 00h on reset.

Location: FF F68Ah

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**BRK**	**ATN**	**Reserved**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-3		**Reserved.**
2	R/W	**BRK (Break Transmit).** Setting the bit to 1 causes SOUT_CR to go low. SOUT_CR remains low until BRK bit is set to 0. 0: Normal transmission (default). 1: Transmit Line Break (SOUT_CR at low level).
1	R/W	**ATN (Attention Mode).** Selects the Attention mode of operation. Cleared by hardware after reception of an address frame that is a 9-bit character with a '1' in the ninth bit position. 0: Disable Attention mode (default). 1: Enable Attention mode.
0		**Reserved.**

4.0 Embedded Controller Modules (Continued)

Baud Rate Divisor Register (UBAUD)

Contains the lower eight bits of the baud rate divisor. The UPSR register contains the upper three bits. The register is cleared to 00h on reset.

Location: FF F68Ch

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**UDIV7-0**							
Reset	0	0	0	0	0	0	0	0

Baud Rate Prescaler Register (UPSR)

Contains the prescaler value and the upper three bits of the baud rate divisor. The register is cleared to 00h on reset.

Location: FF F68Eh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**UPSC**					**UDIV10-8**		
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-3	**UPSC (Prescaler Select).** Holds the 5-bit prescaler value. For the conversion to the prescaler factor, see "Baud Rate Generator" on page 125.
2-0	**UDIV10-8 (Baud Rate Divisor).** Holds the upper three bits of the baud rate divisor. UDIV10-0 is used to generate the baud rate clock, as shown in "Baud Rate Generator" on page 125.

4.10.5 Usage Hints

Calculating the Baud Rate

The equation for calculating the baud rate is:

$$BR = CLK/(16 \times DIV \times P)$$

where:

- BR is the baud rate.
- CLK is the core clock.
- DIV = UDIV10-0 + 1 (see "Baud Rate Divisor Register (UBAUD)" on page 131).
- P is the "Prescaler Factor" selected by UPSC (see Table 16 on page 125).

Assuming a core clock of 5 MHz and a desired baud rate of 9600, the (DIV x P) term, according to the equation above, is:

$$DIV \times P = (5x10^6)/(16x9600) = 32.552$$

The (DIV x P) term is then divided by each prescaler factor in Table 16 on page 125 to obtain a value closest to an integer. The factor for this example is P = 6.5; therefore,

$$DIV = 32.552/6.5 = 5.008 \ (DIV = 5)$$

The baud rate register is programed with a baud rate divisor of 4 (DIV = UDIV10-0 + 1). This produces a baud clock of:

$$BR = (5x10^6)/(16x5x6.5) = 9615.385$$

$$\% \ error = (9615.385-9600)/9600 = 0.16$$

4.0 Embedded Controller Modules (Continued)

Note that the percent error is much lower than would be possible without the non-integer prescaler factor. Refer to the table below for more examples.

Core Clock	Required Baud Rate	DIV	P	Actual Baud Rate	Percent Error
4 MHz	9600	2	13	9615.385	0.16
5 MHz	9600	5	6.5	9615.385	0.16
10 MHz	19200	5	6.5	19230.769	0.16
20 MHz	19200	5	13	19230.769	0.16
25 MHz	9600	13	12.5	9615.385	0.16
25 MHz	19200	6	13.5	19290.123	0.47

4.0 Embedded Controller Modules (Continued)

4.11 TIMER AND WATCHDOG (TWD)

The Timer and Watchdog module (TWD) generates the clocks and interrupts used for timing periodic functions in the system. It also provides watchdog protection over software execution.

The TWD provides flexibility in system configuration by enabling the configuration of various clock ratios. After setting the TWD configuration, the software can lock it to provide a higher level of protection against subsequent erroneous software action. Once a section of the TWD is locked, only reset or the unlock sequence releases it. Figure 31 shows the TWD block diagram.

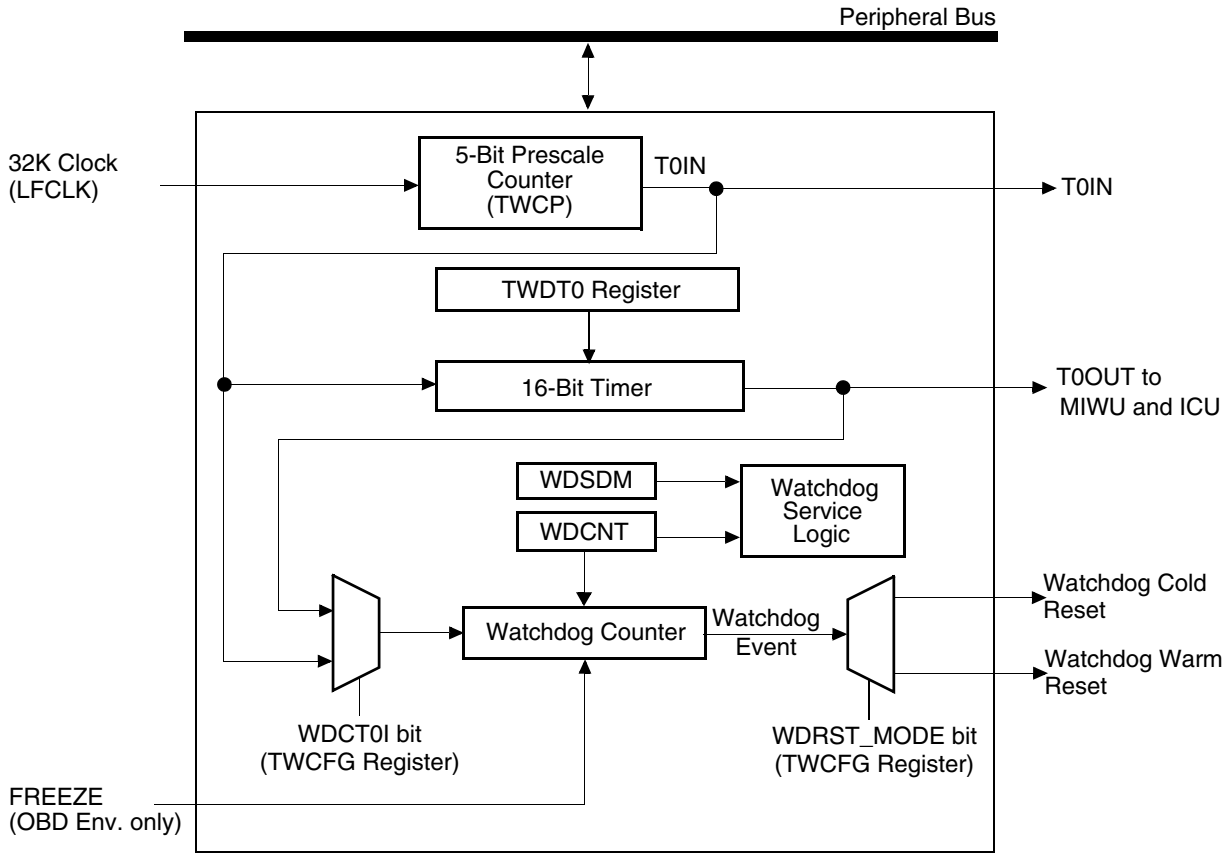


Figure 31. Timer and Watchdog Block Diagram

4.11.1 Features

- 32.768 KHz input clock
- Programmable prescale counter
- 16-bit programmable periodic interrupt timer
- 8-bit watchdog counter
- Watchdog reset signal generation in response to failure detection, such as:
  - — Watchdog service performed too early
  - — Watchdog service performed too late
  - — Wrong data used in a service by data match
- Two selectable watchdog reset options (Cold and Warm)
- Watchdog input clock selector
- Watchdog Freeze input
- Configuration lock option for fully protected watchdog
- Watchdog stop and unlock options using a special command sequence

4.0 Embedded Controller Modules (Continued)

4.11.2 Functional Description

Input Clock

All TWD counting activities are based on the 32K clock (LFCLK). The watchdog can count using a division of the LFCLK clock (either T0OUT or T0IN).

Prescale

A prescale counter divides the LFCLK input clock (32.768 KHz) by a factor of 2^{MDIV}. MDIV in TWCP register is in the range of 0 through 5 (i.e., divide ratio of 1:1 through 1:32). The prescaled output is used as an input clock for a 16-bit timer (TWDT0) and is referred to as T0IN.

TWD Timer 0

The TWD Timer 0 is a 16-bit, programmable, automatically retriggered down counter. It counts on the rising edge of T0IN. It starts from the value loaded to TWDT0 register down to 0 and then restarts counting from TWDT0 at the next T0IN cycle.

When the counter reaches 0, T0OUT is set (1) for one T0IN cycle. This makes the Timer 0 cycle:

```
(TWDT0 + 1) x T0IN-cycle.
```

T0OUT is input to MIWU and ICU and can be used as the time base for activities such as system tick.

When TWDT0 is loaded with a new value, the counter uses it the next time it restarts counting (i.e., after reaching 0). If RST in Timer Control register (T0CSR) is written 1, the timer is restarted on the next rising edge of T0IN.

Notes:

- RST bit in T0CSR register is cleared after completing this load.
- When MDIV in TWCP register is 0, the timer counter might skip one count when loaded with a new value.

Watchdog Operation

The watchdog is an 8-bit down counter, operating on the rising edge of its currently selected clock source. On reset, it is stopped (i.e., it does not count and no watchdog event signal is generated).

Start. The watchdog is started by a write to the Watchdog Count register (WDCNT), regardless of WDSDME bit in TWCFG register. Then, the watchdog begins counting down from the value written in WDCNT. In addition, if the “service on data match” is enabled (WDSDME bit is 1), writing 5Ch to WDSDM register also starts the watchdog counter from the value currently in WDCNT. Once the watchdog starts counting down, only a Core Domain reset, Watchdog Warm reset or stop sequence can stop it.

Touch. If the watchdog is running, a restart (“touch”) operation is performed according to the mode selected by WDSDME in TWCFG register:

- WDSDME = 0: By a write to the Watchdog Count register (WDCNT); however, if LWDCNT in TWCFG register is 1, the data written is ignored.
- WDSDME = 1: By writing 5Ch to the Watchdog Service Data Match (WDSDM) register. In this case, writing to WDCNT register (if LWDCNT is 0) only updates the value to be loaded into the counter at the next touch operation.

Event. A watchdog event signal is triggered if one of the following occurs:

- The counter reaches 0 (too late service).
- The watchdog is written to more than once per three watchdog clock cycles, for the currently selected clock (too early service).
- While the watchdog is running, data other than either the stop sequence (when WDSDME in TWCFG register is set to 1) or 5Ch is written to WDSDM register.

Watchdog Stop and Register Unlock

Stop. In addition to Core Domain reset and Watchdog Warm reset, the watchdog can be stopped by writing the “stop sequence” to the WDSDM register, regardless of the setting of WDSDME bit in TWCFG register. The stop sequence is: 87h, 61h, 63h; any other value, or an out-of-sequence value, resets the sequence and, if WDSDME bit in TWCFG register is set to 1, triggers a watchdog event. The current operation status of the watchdog counter is indicated by WD_RUN bit in T0CSR register.

A watchdog operation can be restarted, as explained in ”Watchdog Operation”, above.

4.0 Embedded Controller Modules (Continued)

Unlock. If the TWD registers are locked (i.e., by setting the LTWCFG, LTWCP, LTWDT0 or LWDCNT bits in TWCFG register to 1), they can be unlocked by writing the "unlock sequence" to the WDSDM register while the watchdog is **stopped**. The unlock sequence is also: 87h, 61h, 63h; any other value or a out-of-sequence value resets the sequence.

Watchdog Clock Source Selection

Select the clock source as follows:

- WDCT01 bit in TWCFG register is 0: T0OUT
- WDCT01 bit in TWCFG register is 1: T0IN

Changing the watchdog clock source may cause it to gain or lose one clock cycle.

Notes:

- When MDIV in TWCP register is 0, the watchdog counter may skip one count when loaded with a new value.
- After activating the watchdog, avoid entering Idle or Deep Idle mode in the first four LFCLK clock cycles.

Watchdog Reset Generation

The watchdog event can be routed to either a Watchdog Cold reset or a Watchdog Warm reset by WDRST_MODE bit in TWCFG register.

In addition, a Watchdog Cold reset (i.e., a watchdog event with WDRST_MODE bit set to 0) can be enabled to trigger a VCC_POR reset pulse by the WDC2POR bit in TWCFG register.

The occurrence of a watchdog event is indicated by WDRST_STS bit in T0CSR register.

TWD Control and Configuration

The TWD Configuration register (TWCFG) allows you to:

- Generate a system reset ($\overline{\text{VCC\_POR}}$) if a Watchdog Cold reset occurs.
- Select the type of reset to be generated: Watchdog Cold or Watchdog Warm.
- Set the watchdog clock source: T0IN or T0OUT.
- Enable watchdog "service on data match" (write to WDSDM register).
- Define which of TWCFG, TWCPR, TWDT0, T0CSR and WDCNT registers is/are locked.

Once LTWCFG, LTWCP, LTWDT0 or LWDCNT (in TWCFG register) is set, its respective resources are locked and can be cleared only by reset or by the unlock sequence. Setting any of these registers prevents runaway software from tampering with the respective watchdog function.

Operation in Idle and Deep Idle Modes

The TWD is active in both Idle and Deep Idle modes: the counters continue to function, and interrupts and error signals are issued.

Since writes to TWCP, TWDT0 and WDCNT may be delayed by up to three LFCLK clock cycles, the software must avoid entering Idle and Deep Idle modes during this period. WDTLD bit in T0CSR register indicates when it is safe to change power modes.

4.11.3 TWD Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20. All TWD registers are powered by V_{CC} and are reset by Core Domain reset unless otherwise specified in the register description below.

TWD Register Map

Location	Mnemonic	Register Name	Size	Type
FF F4C0h	TWCFG	Timer and Watchdog Configuration	Byte	Varies per bit or RUP
FF F4C2h	TWCP	Timer and Watchdog Clock Prescaler	Byte	R/W or RUP
FF F4C4h	TWDT0	TWD Timer 0	Word	R/W or RUP
FF F4C6h	T0CSR	TWDT0 Control and Status	Byte	Varies per bit
FF F4C8h	WDCNT	Watchdog Count	Byte	WO or Touch
FF F4CAh	WDSDM	Watchdog Service Data Match	Byte	WO

4.0 Embedded Controller Modules (Continued)

Timer and Watchdog Configuration Register (TWCFG)

TWCFG defines the watchdog reset mode, clock input and service method and enables TWD register locking. The non-reserved bits are initialized to 0 on reset.

Location: FF F4C0h

Type: Varies per bit or RUP

Bit	7	6	5	4	3	2	1	0
Name	**WDC2POR**	**WDRST_MODE**	**WDSDME**	**WDCT0I**	**LWDCNT**	**LTWDT0**	**LTWCP**	**LTWCFG**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W	**WDC2POR (Watchdog Routing to VCC_POR).** When set to 1, enables a Watchdog Cold reset to trigger a reset pulse at the $\overline{\text{VCC\_POR}}$ pin; see Section 3.2.7 on page 50. No reset pulse is triggered by a watchdog event if WDRST_MODE bit is set to 1 (Watchdog Warm reset). WDC2POR is set to its default value by V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset. 0: $\overline{\text{VCC\_POR}}$ pulse triggered only by V_{CC} power-up (default). 1: $\overline{\text{VCC\_POR}}$ pulse triggered **also** by a Watchdog Cold reset.
6	R/W	**WDRST_MODE (Watchdog Reset Mode Select).** Selects the type of reset generated by the occurrence of a watchdog event; see Section 3.2 on page 48. WDRST_MODE is set to its default value by V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset. 0: Generate Watchdog Cold reset (default). 1: Generate Watchdog Warm reset.
5	R/W	**WDSDME.** Selects the watchdog touch mechanism. 0: Watchdog touched by writing to WDCNT register (default). 1: Watchdog touched by writing 5Ch to WDSDM register (“service on data match”).
4	R/W	**WDCT0I.** 0: Selects T0OUT clock as the watchdog clock (default). 1: Selects T0IN clock as the watchdog clock.
3	R/W1S	**LWDCNT.** Once set, can be cleared either by reset or by the unlock sequence. When WDSDME is 0, touch operations (i.e., writing to WDCNT register) may be performed, regardless of the setting of LWDCNT. 0: Enables write to WDCNT register (default). 1: Any data written to WDCNT register is ignored; reading from it returns unpredictable values.
2	R/W1S	**LTWDT0.** Once set, can be cleared either by reset or by the unlock sequence. 0: Enables read/write from/to TWDT0 and T0CSR registers (default). 1: TWDT0 and T0CSR cannot be written to and TWDT0 cannot be read. Any data written to TWDT0 or T0CSR is ignored. An exception to this is WDRST_STS bit, which can be written even when T0CSR register is locked. Reading from TWDT0 returns unpredictable values.
1	R/W1S	**LTWCP.** Once set, can be cleared either by reset or by the unlock sequence. 0: Enables read/write from/to TWCP register (default). 1: Any data written to TWCP is ignored; reading from it returns unpredictable values.
0	R/W1S	**LTWCFG.** Once set, can be cleared either by reset or by the unlock sequence. 0: Enables read/write from/to TWCFG register (default). 1: Any data written to TWCFG is ignored; reading from it returns unpredictable values.

4.0 Embedded Controller Modules (Continued)

Timer and Watchdog Clock Prescaler Register (TWCP)

TWCP defines the prescale ratio of the input clock and generates the T0IN clock. The non-reserved bits are initialized to 0 on reset.

Location: FF F4C2h

Type: R/W or RUP

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**MDIV**		
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-3	**Reserved.**
2-0	**MDIV.** Defines the prescale ratio of the input clock. The prescale ratio is 2^{MDIV}. The value of MDIV must be in the range of 0-5, providing a prescale ratio of 1 to 32. MDIV valid values are: **Bits 2 10 Clock Ratio** 0 00: 1:1 (default) 0 01: 1:2 0 10: 1:4 0 11: 1:8 1 00: 1:16 1 01 1:32 Other Reserved

TWD Timer 0 Register (TWDT0)

TWDT0 defines the T0OUT interrupt rate. It is initialized to FFFFh on reset.

Location: FF F4C4h

Type: R/W or RUP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**PRESET**															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Description
15-0	**PRESET.** Defines the counter PRESET value. When the counter reaches 0, it restarts counting down from this value. The T0OUT frequency is the T0IN frequency divided by (PRESET+1). The allowed values of the PRESET field are 0001h through FFFFh.

4.0 Embedded Controller Modules (Continued)

TWDT0 Control and Status Register (T0CSR)

T0CSR controls the operation and provides the status of the T0 timer. The non-reserved bits are cleared (0) on reset.

Location: FF F4C6h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**WD_RUN**	**WDRST_STS**	**WDLTD**	**Reserved**	**TC**	**RST**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-6		**Reserved.**
5	RO	**WD_RUN (Watchdog Run Status).** Indicates the current operation status of the watchdog counter. 0: Watchdog counter is stopped (default). 1: Watchdog counter is running.
4	RW1C	**WDRST_STS (Watchdog Reset Status).** When set to 1, indicates the occurrence of a watchdog event (which in turn generates either a Watchdog Cold or Warm reset). Writing 1 clears this bit; writing 0 is ignored. WDRST_STS is set to its default value by V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset. This bit is not locked by LTWT0 bit in TWCFG register. 0: No watchdog event occurred from the time the bit was last cleared (default). 1: Either Watchdog Cold reset or Watchdog Warm reset was generated.
3	RO	**WDLTD (Watchdog Last Touch Delay).** Is set when a valid watchdog touch is performed. Cleared after the watchdog counter is updated. (After the watchdog counter is updated, it is safe to switch to Idle or Deep Idle mode.)
2		**Reserved.**
1	RO	**TC (Terminal Count).** Read-only bits that indicates that the counter has reached 0 (terminal count). Cleared each time the register is read. Data written to it is ignored.
0	R/W1S	**RST (Reset).** Synchronizes the TWD Timer 0 to a software command. When set (1), forces the Timer 0 to reload from TWDT0 register and to restart counting on the next input clock rising edge. Cleared by the input clock rising edge, which indicates that the counter has resumed its automatic retriggerable operation. Writing 0 to this bit is ignored.

Watchdog Count Register (WDCNT)

WDCNT holds the value to be loaded into the watchdog counter when the counter is touched. The counter then starts counting down from the PRESET value. A write to WDCNT functions as a “touch” operation even if WDCNT is locked, in which case, the watchdog counter is restarted using the value loaded in PRESET field before WDCNT was locked (i.e., the new PRESET value is ignored). A watchdog event signal is triggered if WDCNT is written more than once per three watchdog clock cycles. This register is initialized to 0Fh on reset.

Location: FF F4C8h

Type: WO or “Touch”

Bit	7	6	5	4	3	2	1	0
Name	**PRESET**							
Reset	0	0	0	0	1	1	1	1

Bit	Description
7-0	**PRESET.** Defines the counter preset value. A value of 00h is invalid.

Watchdog Service Data Match Register (WDSDM)

If WDSDME in TWCFG register is set to 1, the watchdog counting restarts from the value in WDCNT, when WDSDM is written with 5Ch. If any other data (except for the stop/unlock sequence) is written to this register, it triggers a watchdog event signal. A watchdog event signal is also triggered if 5Ch is written more than once per three watchdog clock cycles.

WDSDM also serves to stop the watchdog operation and to unlock the TWD registers by writing the stop/unlock sequence, as described in "Watchdog Stop and Register Unlock" on page 134. When WDSDME bit in TWCFG register is set to 0, a write to this register, except for the stop/unlock sequence, is ignored.

Location: FF F4CAh

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**RSDATA**							

Bit	Description
7-0	**RSDATA.**

4.11.4 Usage Hints

TWD protects watchdog operation from software tampering. To achieve the highest level of protection:

1. Program TWDT0 prescale and TMWT0 timers to the desired values.
2. Configure the watchdog clock to use T0IN or T0OUT using WDCT0I bit in TWCFG register.
3. Configure the type of reset (Cold or Warm) to be generated, using WDRST_MODE bit in TWCFG register.
4. Program WDCNT to the maximum period between watchdog touch operations. Note that from this point, the watchdog starts operating and must be touched periodically to prevent a watchdog event signal.
5. Configure the watchdog to use "service on data match", and lock all the TWD registers by setting bits 0-3 and bit 5 of TWCFG to 1.
6. Touch the watchdog by writing 5Ch to WDSDM at the appropriate rate (i.e., no more than once every three watchdog clock cycles, and no less than the period programed to WDCNT).

4.0 Embedded Controller Modules (Continued)

4.12 ANALOG TO DIGITAL CONVERTER (ADC)

The WPC8763L includes an 8-bit resolution Analog to Digital Converter (ADC). Up to four external voltage inputs can be measured. Measurement can be triggered by software, by a timer event, or automatically, in Autoscan mode.

Each input channel is assigned a separate result register, which is updated at the end of the conversion.

4.12.1 Features

- Voltage measurement
  - four external voltage inputs
  - 8-bit resolution
  - External voltage reference (V_{REF})
  - 0V to V_{REF} input voltage range
  - High-impedance, ground-referenced inputs
  - Input capacitance of 50 pF (typical)
  - Integral Non Linearity better than ±2.3 LSB
  - Offset better than ±2.5 LSB
  - Full Scale Error better than ±2.5 LSB
  - Monotonic
  - Accuracy better than ±3 LSB
- Separate result register per each input channel
- Sampling sequence and timing
  - 125 µs conversion time
  - Cyclic scan of inputs, for user selectable input channels
- Measurement Trigger
  - Software trigger
  - Timer pulse-based trigger
  - Repetitive Scan (Autoscan) mode for selected inputs
  - Single (One-Shot) or Repetitive conversion, for software trigger and scan
- Power consumption
  - Zero current when disabled
  - Low operating current

4.12.2 Functional Description

Inputs. The ADC has four inputs:

- Four External Voltage Inputs (AD0 to AD3).

Analog Multiplexer. A 4-to-1 analog multiplexer selects one of the inputs for measurement by connecting it to the A/D converter input.

Successive Approximation A/D Converter. The high-resolution A/D converter receives the selected input voltage and converts it. The result of the conversion is an 8-bit, unsigned integer digital value.

Reference Voltage. The A/D converter reference voltage is supplied from an external reference voltage, via the VREF pin.

Conversion Modes. The A/D converter selects an input source and initiates a conversion in one of three conversion modes (see Table 17 on page 141):

- **Software-Triggered.** In this mode, the core initiates the conversion by selecting the required input and setting START bit in ADCCNF register to 1. The multiplexer is set to the input selected by SADDR field in ASCADD register. When the conversion is completed, the result is written to the corresponding data register and the End Of Conversion Event flag is set. If the interrupt is enabled, the End Of Conversion Event flag generates an interrupt to the core.
  If Repetitive mode is selected (ADCRPTC bit in ADCCNF register is set to 1), a new conversion starts immediately. Otherwise, the software must start a new conversion by setting the START bit to 1.

4.0 Embedded Controller Modules (Continued)

- **Timer-Triggered.** This mode is selected by setting ADCTTE bit in ADCCNF register to 1. In this mode, the timer (MFT16-2) output TA2 initiates the A/D conversion. The multiplexer is set to the input selected by TTADDR field in ASCADD register. After the conversion ends, the conversion result is written to the corresponding data register and the End Of Timer-Triggered Conversion Event flag is set. If the interrupt is enabled, the End Of Timer-Triggered Conversion Event flag generates an interrupt to the core.

  A new conversion is performed on each pulse generated at TA2 output as long as ADCTTE bit is set to 1. Using the timer trigger to measure more than one input requires changing the contents of TTADDR field after every conversion.

- **Scan.** This mode is selected by setting ADCMD field in ADCCNF register to '01'. The core initiates the conversion by setting the START bit in ADCCNF register to 1. In this mode, selected inputs are scanned and converted. The input combination is selected by setting the corresponding bits in ADCCS register. Every set bit enables an input to be converted in the scan (any input combination may be selected). When the scan is complete, the End Of Conversion Cycle Event flag is set. If the interrupt is enabled, the End Of Conversion Cycle Event flag generates an interrupt to the core. An interrupt can also be generated for every completed conversion within the scan.

  If the Repetitive mode is selected (ADCRPTC bit in ADCCNF register is set to 1), a new scan starts immediately. Otherwise, the software must start a new scan by setting the START bit to 1.

Timer-Triggered mode can be performed "in parallel" with Software-Triggered mode. Conversions from the two modes are interleaved: a new triggered conversion from one mode waits to be performed until the completion of the currently performed conversion from the other mode. There is no priority between the modes regarding conversion execution.

Software-Triggered mode or Timer-Triggered mode cannot be selected together with Scan mode.

At the end of each A/D conversion, the result is stored in the corresponding data register.

Table 17. ADC Conversion Modes

Trigger Source	Conversion Mode	One-Shot/Repetitive	Converted Channel(s)
Software (START = 1)	Channel (ADCMD = 00)	One-shot (ADCRPTC = 0): START is cleared after conversion start	Single channel, selected by SADDR
		Repetitive (ADCRPTC = 1): START is cleared by software (write 1 to STOP)	
	Scan (ADCMD = 01)	One-shot (ADCRPTC = 0): START is cleared after conversion start	Multiple channels, selected by ADCCS
		Repetitive (ADCRPTC = 1): START is cleared by software (write 1 to STOP)	
Timer (ADCTTE = 1)	Channel (ADCMD = 00)	A conversion is performed on each timer pulse[1] until ADCTTE is cleared by software (write 0)	Single channel, selected by TTADDR

1. The conversion is triggered when the TA2 internal signal goes high.

Conversion Data Registers. Each A/D converter input (internal or external) has a corresponding data register that is updated after the input value is converted by the A/D converter.

Timing Control. This block reduces the frequency of the core clock (CLK) to the lower value required by the ADC.

4.0 Embedded Controller Modules (Continued)

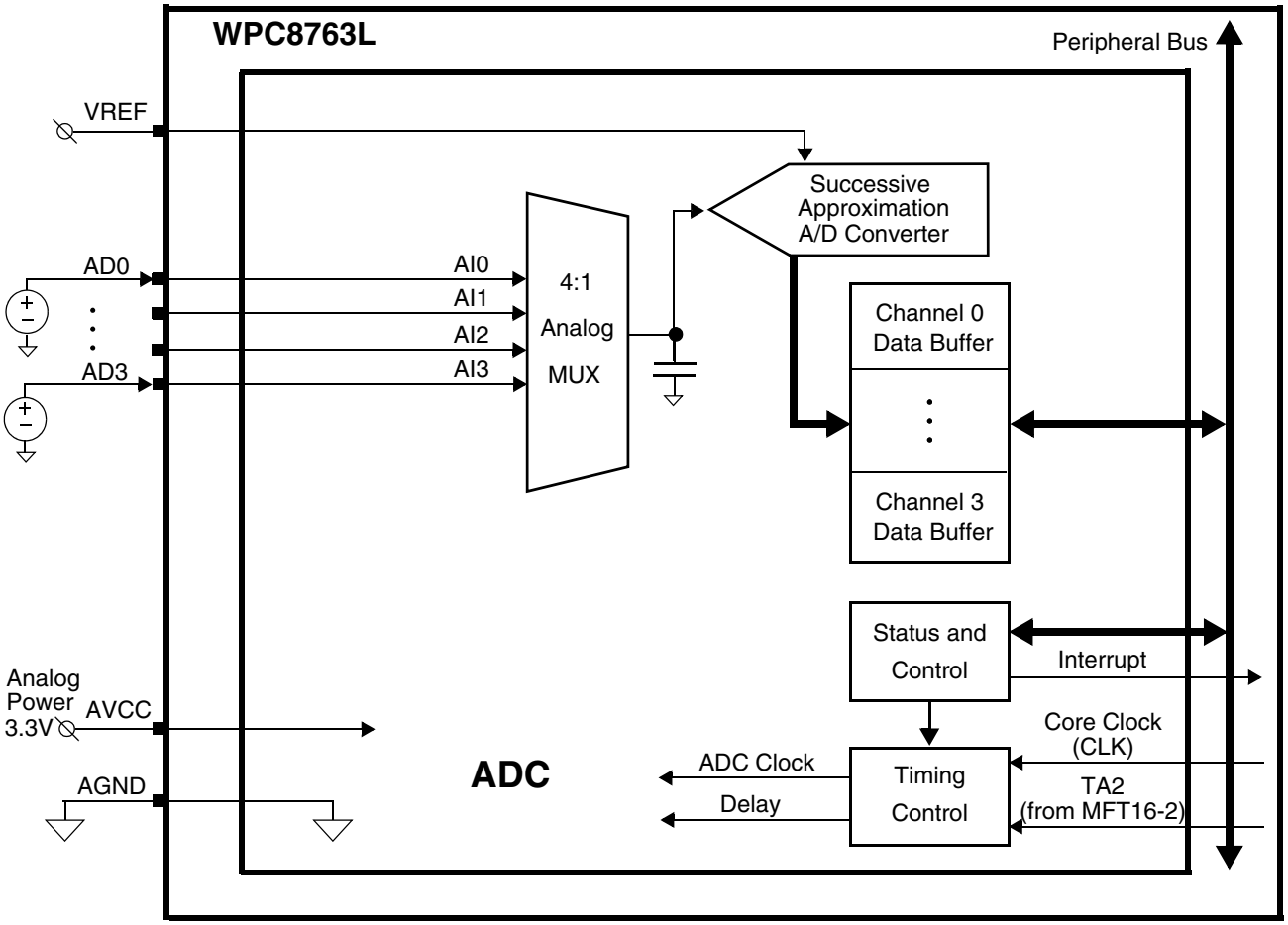


Figure 32. ADC Functional Diagram

4.12.3 Voltage Measurement

The ADC performs a linear conversion of the input voltage signal to an 8-bit, unsigned digital representation. The input signal should be applied relative to the AGND pin and should range from 0V to V_{FS} (Full Scale).

Use the equations in the following table to calculate the input voltage. The calculation is based on the reading from the Channel *x* Data Buffer (CHDAT field in CHN*x*DAT register).

Input Channel	Calculation[1]
AD0 to AD3 (external inputs)	Vi = CHDAT(9-2) * V_{FSE} / 256

1. See Section 8.4.1 on page 285 for the dynamic range relevant for each input.

4.12.4 ADC Operation

Reset

Section 3.2 on page 48 describes the WPC8763L reset types. The ADC is affected by Core Domain reset, as described below:

- All control, configuration and status registers are reset to their default values, as indicated in Section 4.12.5 on page 145.
- The clock division factor is set to its maximum value (for the slowest ADC operation speed).
- The ADC is disabled, with all interrupt sources masked and all event status bits are reset.

ADC Clock

The ADC clock is generated by dividing the core clock by a factor of between 1 and 64, as defined in SCLKDIV field in ATCTL register; see page 148. The ADC clock frequency must be F_{ACLK}; see Section 8.4.1 on page 285. SCLKDIV must be programed prior to enabling the ADC (i.e., while ADCEN of ADCCNF register is 0).

4.0 Embedded Controller Modules (Continued)

Enabling and Disabling the ADC

Enabling the ADC. The ADC module is enabled by setting ADCEN bit in ADCCNF register to 1. When the ADC module is enabled after reset, the first conversion starts after a delay of t_{END} (see Section 8.4.1 on page 285) from the moment the measurement is enabled (see START and ADCTTE bits in ADCCNF register on page 146).

Disabling the ADC. When either of the following conditions occurs, the ADC module is disabled:

- Core Domain reset is asserted.
- The software resets ADCEN bit in ADCCNF register.

When disabled, all ADC module activities are halted and the module's current consumption from AV_{CC} is lowered.

Initializing the ADC

The ADC module must be initialized before it is started by setting either the START or the ADCTTE bits. To do this, set these fields/registers, as follows:

1. Core Clock Division Factor - SCLKDIV field in ATCTL register.
2. ADC Operation Mode - ADCMD field in ADCCNF register.
3. ADC Repetitive Mode - ADCRPTC bit in ADCCNF register.
4. Setup the MFT16-2 timer to generate the TA2 pulse (if Timer-Triggered mode is used).
5. ADC module Enable - ADCEN bit in ADCCNF register.

Setting Up the MFT16-2 Timer. To set up the timer for the correct pulse generation (see Section 4.8 on page 101):

1. Set Reload/Capture A register (T2CRA) to a value "N" according to the formula:

$$T_{\text{Timer-Triggered ADC cycle}} = 2 * (N + 1) * t_{32KOSC}$$

Where t_{32KOSC} is the period of the LFCLK clock (32.768 KHz – see "Low-Frequency Clock Timing" on page 290).

2. Set the Timer Mode Control register (T2MCTRL) as follows:
   - Set TAOUT bit to 1 (this starts TA2 toggle sequence, beginning with high level).
   - Set TAEN bit to 1 (this enables TA2 output from timer to the ADC module).
   - Set MDSEL field to '010' (this sets MFT16-2 to Mode 3).
3. Set TAIEN bit in Timer Interrupt Enable (T2IEN) register to 0 (this disables interrupt from Timer 1 reload).
4. Set C1CSEL field in Clock Unit Control register (T2CKC) to '100' (this selects LFCLK as the clock input and starts the operation of Timer 1).

ADC Operation Stop

Software-Triggered, Channel and Scan Conversion Modes. In these modes, when STOP bit in ADCCNF register is set to 1 by software, START bit is set to 0 and ADC operation is stopped immediately (i.e., in-progress or pending conversions are aborted or discarded, respectively).

Note that, when STOP is set to 1, a Timer-Triggered measurement that is either in-progress or pending is aborted or discarded, respectively. However, the Timer-Triggered Conversion mode remains active.

Timer-Triggered Conversion Mode. In this mode, ADC operation is stopped by setting ADCTTE bit in ADCCNF register to 0. If a Timer-Triggered measurement is pending, it is discarded; if the measurement is in-progress, ADC operation stops after the measurement ends. To stop ADC operation immediately (i.e., to discard any in-progress measurement), set STOP bit in ADCCNF register to 1 after setting ADCTTE bit to 0.

Interrupt Structure

The ADC Interrupt is generated for any of the events listed in Table 18. The ADC interrupt is connected to the ICU.

Table 18. ADC Interrupt Structure

Event Flag	Register Mnemonic	Mask Bit	Register Mnemonic	Description
EOCEV	ADCSTS	INTECEN	ADCCNF	End Of Conversion Event
EOCCEV	ADCSTS	INTECCEN	ADCCNF	End Of Conversion Cycle Event
EOTCEV	ADCSTS	INTETCEN	ADCCNF	End Of Timer-Triggered Conversion Event
OVFEV	ADCSTS	INTOVFEN	ADCCNF	Data Overflow Event

4.0 Embedded Controller Modules (Continued)

When an event flag and its related enable bit are set, the ADC Interrupt request is asserted. The software must reset the event flag (or reset its mask bit) to deassert the ADC Interrupt request.

The ADC Interrupt is routed to the ICU as an ADCI signal; see Section 4.3 on page 74.

ADC Operating Principles

The ADC module can operate in two modes:

Single Channel Operation Mode

This mode includes the Software-Triggered and Timer-Triggered conversion modes. Measurement starts when either START bit is set (for Software-Triggered mode) or, if ADCTTE bit is set, a pulse is generated by the MFT16-2 timer (for Timer-Triggered mode). This initiates voltage conversion of the selected channel by ASCADD register. When the A/D conversion process starts, the input is selected. The resulting -bit digital value is stored in the respective CHNnDAT register, and the NEW flag is set.

Multiple Channels Operation Mode

This mode includes the Scan conversion mode. Measurement starts when START bit is set. All the channels selected by the ADCCS register are converted one by one in ascending order. Once the conversion cycle starts, START bit is cleared if ADCRPTC bit is set to 0. When all the selected channels are converted, EOCEV bit in ADCSTS register is set.

The software can read the measurement result for each channel immediately after it is converted (EOCEV flag can be used). Alternatively, the results can be read at the end of the cycle when EOCCEV flag is set.

The ADC conversion cycle duration is calculated using the formula below (N is the number of enabled channels):

$$T_{ADC\ cycle} = N * (t_{CC})$$

Where:t_{CC} - Channel Conversion time; see Section 8.4.1 on page 285.

If the channel selection is not changed, the same inputs are measured during the next ADC cycle. This gives a sampling rate of a full $T_{ADC\ cycle}$ for a specific input.

Operation Sequences

The following fields or bits must be set-up while conversions are disabled:

- ADCRPTC bit in ADCCNF register.
- ADCMD field in ADCCNF register.
- TTADDR and/or SADDR fields in ASCADD register.
- CCn bits in ADCCS register.

After the ADC module is properly enabled and initialized, one of the following example sequences can be used:

EOCEV-Driven Sequence, for Single Channel Operation Mode

1. Write the number of the channel to be converted to ASCADD register.
2. Select One-Shot or Repetitive mode by setting ADCRPTC in ADCCNF.
3. Set START in ADCCNF to start a conversion.
4. Wait for EOCEV flag to be set.
5. Read the measured data from CHDAT in CHNnDAT register.
6. Clear the EOCEV flag by writing 1 to it.
7. If Repetitive mode was selected, the operation repeats from step 4.

EOTCEV-Driven Sequence for Single Channel Operation Mode

1. Write the number of the channel to be converted to ASCADD register.
2. Set ADCTTE in ADCCNF to enable the timer trigger.
3. Wait for EOTCEV flag to be set.
4. Read the measured data from CHDAT in CHNnDAT register.
5. Clear the EOTCEV flag by writing 1 to it.
6. Return to step 3 for another measurement.
7. Clear ADCTTE to stop the measurements.

EOCCEV-Driven Sequence for Multiple Channels Operation Mode

1. Select the channels to be converted by writing to ADCCS register.
2. Select One-Shot or Repetitive mode by setting ADCRPTC in ADCCNF.
3. Set START in ADCNF to start a Scan conversion.
4. Wait for EOCCEV flag to be set.
5. Read the measured data from CHDAT in CHNnDAT register for each selected channel.
6. Clear the EOCCEV flag by writing 1 to it.
7. If Repetitive mode was selected, the operation repeats from step 4.

ADC Operation Stop Sequence

1. Depending on the active conversion mode(s), stop ADC operation as follows:
   - By writing 1 to STOP bit in ADCCNF register.
   - By writing 0 to ADCTTE bit, then writing 1 to STOP bit, both in ADCCNF register.
2. Wait 1 μs for ADC operation to stop.
3. Clear the event flags (OVFEV, EOTCEV, EOCCEV, EOCEV) in ADCSTS register by writing 1 to them.
4. Reconfigure the ADC registers before restarting ADC operation.

Reading Measurement Results

A separate result register is allocated for each ADC module input for a total of four result registers. An End Of Conversion can be detected either by using an interrupt or by polling the status flags. Once an End Of Conversion is detected, the user may read the newly converted data from the specific result register. In Software-Triggered and Timer-Triggered modes, the input select field in ASCADD register holds the number of the last converted input.

Overflow Detection

An overflow occurs when a conversion result is written to a CHNnDAT register before the previous measurement of this channel was read. If an overflow occurs for a channel, the new measurement result overrides the old data in CHNnDAT register, and OVFEV flag in ADCSTS register is set. In this case, the result of the previous measurement is lost.

4.12.5 ADC Registers

All ADC registers are powered by V_{CC} and are set to their default values by Core Domain reset.

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Register Map

Location	Mnemonic	Register Name	Size	Type
FF F800h	ADCSTS	ADC Status	Word	Varies per bit
FF F802h	ADCCNF	ADC Configuration	Word	Varies per bit
FF F804h	ATCTL	ADC Timing Control	Word	R/W
FF F806h	ASCADD	ADC Single Channel Address	Word	R/W
FF F808h	ADCCS	ADC Scan Channels Select	Word	R/W
FF F840h	CHN0DAT	Channel 0 Data Buffer	Word	RO
FF F842h	CHN1DAT	Channel 1 Data Buffer	Word	RO
FF F844h	CHN2DAT	Channel 2 Data Buffer	Word	RO
FF F846h	CHN3DAT	Channel 3 Data Buffer	Word	RO

4.0 Embedded Controller Modules (Continued)

ADC Status Register (ADCSTS)

ADCSTS indicates the global status of the ADC module.

Location: FF F800h

Type: Varies per bit

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**							
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**				**OVFEV**	**EOTCEV**	**EOCCEV**	**EOCEV**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
15-4		**Reserved.**
3	R/W1C	**OVFEV (Data Overflow Event).** Indicates that the measurement data in one of the CHNnDAT registers was overwritten with new data before being read. When a data overflow occurs, NEW bit (in CHNnDAT) remains set and new data overrides the current data in CHNnDAT register. 0: No overflow (default). 1: Overflow.
2	R/W1C	**EOTCEV (End Of Timer-Triggered Conversion Event).** Indicates that a channel conversion, triggered by a timer pulse, was completed. 0: No Timer-Triggered conversion was completed since the bit was last cleared (default). 1: A Timer-Triggered conversion was completed.
1	R/W1C	**EOCCEV (End Of Conversion Cycle Event).** Indicates that a Scan conversion cycle was completed (all the selected channels were measured). For each of the enabled channels, the data for the channel is stored in the respective CHNnDAT register and, if the register was not read during the current cycle, NEW bit in the respective CHNnDAT register is set. 0: No Scan conversion cycle was completed since the bit was last cleared (default). 1: A Scan conversion cycle was completed.
0	R/W1C	**EOCEV (End Of Conversion Event).** Indicates that a channel conversion, triggered by the START bit, was completed. This relates both to a Channel mode conversion and to any conversion in Scan mode. 0: No Software-Triggered conversion was completed since the bit was last cleared (default). 1: A Software-Triggered conversion was completed.

ADC Configuration Register (ADCCNF)

ADCCNF controls the operation and global configuration of the ADC module.

Location: FF F802h

Type: Varies per bit

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**				**STOP**	**Reserved**	**INTOVFEN**	**INTETCEN**
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**INTECCEN**	**INTECEN**	**ADCTTE**	**START**	**ADCRPTC**	**ADCMD**		**ADCEN**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
15-12		**Reserved.**

4.0 Embedded Controller Modules (Continued)

Bit	Type	Description
11	WO	**STOP (ADC Stop Conversion).** When set to 1, stops a Software-Triggered or Scan conversion by setting START bit to 0. Writing 0 is ignored; reading from it returns 0. 0: ADC operation is controlled by START bit (default). 1: Stops a Software-Triggered or a Scan conversion by clearing the START bit.
10		**Reserved.**
9	R/W	**INTOVFEN (Interrupt from Data Overflow Event Enable).** Enables generation of an ADC interrupt on a Data Overflow event (OVFEV in ADCSTS register). 0: Disabled (default). 1: Enabled - ADC Interrupt from OVFEV.
8	R/W	**INTETCEN (Interrupt from End Of Timer-Triggered Conversion Event Enable).** Enables generation of an ADC interrupt on an End Of Timer-Triggered Conversion event (EOTCEV in ADCSTS register). 0: Disabled.(default). 1: Enabled - ADC Interrupt from EOTCEV.
7	R/W	**INTECCEN (Interrupt from End Of Cyclic-Conversion Event Enable).** Enables generation of an ADC interrupt on an End Of Cyclic-Conversion event (EOCCEV in ADCSTS register). 0: Disabled (default). 1: Enabled - ADC Interrupt from EOCCEV.
6	R/W	**INTECEN (Interrupt from End Of Conversion Event Enable)** Enables generation of an ADC interrupt on an End Of Conversion event (EOCEV in ADCSTS register). 0: Disabled (default). 1: Enabled - ADC Interrupt from EOCEV.
5	R/W	**ADCTTE (ADC Timer-Triggered Conversion Enable).** Enables conversions triggered by a pulse from the internal Timer (MFT16-2). 0: Timer-Triggered conversions disabled (default). 1: Timer-Triggered conversions enabled (possible only in Channel conversion mode).
4	R/W1S	**START (ADC Start Conversion).** When set to 1, starts a Software-Triggered or Scan conversion. If ADCRPTC bit is 0, START bit is cleared when either the Software-Triggered conversion starts or the Scan cycle starts. To stop repetitive conversions (when ADCRPTC bit is 1), START bit must be cleared by software writing 1 to STOP bit. Writing 0 is ignored. 0: Software-Triggered or Scan conversions are disabled (default). 1: Start a Software-Triggered or Scan conversion.
3	R/W	**ADCRPTC (ADC Repetitive Mode).** Enables repetitive operation of the ADC. 0: One-Shot: Software-Triggered and Scan conversions are performed one time (START bit is cleared when the single conversion or the Scan cycle starts), when started by the software (default). 1: Repetitive: Software-Triggered and Scan conversions are performed continuously (START bit remains set after the single conversion or the Scan cycle starts) after being started by the software.
2-1	R/W	**ADCMD (ADC Conversion Mode).** Selects the conversion mode of the ADC module. **Bits 2 1 Conversion Mode** 0 0: Channel conversion mode. A single conversion is triggered either by software or by the Timer (default). 0 1: Scan conversion mode. A scan cycle is triggered by software. 1 0: Reserved. 1 1: Reserved.
0	R/W	**ADCEN (ADC Module Enable).** Controls the operation of the ADC module, to minimize power consumption; see "Interrupt Structure" on page 143. 0: ADC disabled (default). 1: ADC enabled.

4.0 Embedded Controller Modules (Continued)

ADC Timing Control Register (ATCTL)

ATCTL controls the division of the system clock to the ADC clock.

Location: FF F804h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**					**Reserved (must be '011')**		
Reset	0	0	0	0	0	0	1	1

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**SCLKDIV**					
Reset	0	0	1	1	1	1	1	1

Bit	Description
15-11	**Reserved.**
10-8	**Reserved.** Must be '011'.
7-6	**Reserved.**
5-0	**SCLKDIV (Core Clock Division Factor).** Controls the division of the core clock to obtain the ADC clock. The core clock frequency is set separately; see Figure 43 on page 181. The resulting ADC clock frequency should be equal to F_{ACLK}; see Section 8.4.1 on page 285. Range: 0 to 63 (default is 63). The core clock is divided by: SCLKDIV+1.

ADC Single Channel Address (ASCADD)

ASCADD selects the channels to be used by the two single Channel conversion modes.

Location: FF F806h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**			**TTADDR**				
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**SADDR**				
Reset	0	0	0	0	0	0	0	0

Bit	Description
15-13	**Reserved.**
12-8	**TTADDR (Timer-Triggered Address).** Channel number to be converted by a Timer-Triggered conversion. Values of not-implemented channels are illegal and may result in unpredictable ADC behavior.
7-5	**Reserved.**
4-0	**SADDR (Software-Triggered Address).** Channel number to be converted by a Software-Triggered conversion. The channel number value is sampled when the conversion starts. Values of not-implemented channels are illegal and may result in unpredictable ADC behavior.

4.0 Embedded Controller Modules (Continued)

ADC Scan Channels Select (ADCCS)

ADCCS selects channels to be converted during a Scan cycle.

Location: FF F808h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**							
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**				**CC3**	**CC2**	**CC1**	**CC0**
Reset	0	0	0	0	0	0	0	0

Bit	Description
15-0	**CCn (Convert Channel n).** 0: ADC does **not** convert channel n in the next Scan cycle (default). 1: ADC converts channel n in the next Scan cycle.

Channel n Data Buffer (CHNnDAT)

CHNnDAT (buffer) holds the data measured by Channel n.

Location: FF F840h + n * 2; n is 0 to 3

Type: RO

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**NEW**	**Reserved**					**CHDAT**								**Reserved**	
Reset	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit	Description
15	**NEW (New Conversion Result).** New channel data is available for reading. Cleared when the register is read (default 0).
14-10	**Reserved.**
9-2	**CHDAT (Channel Data).** Channel n voltage data, measured by the A/D converter. To calculate the voltage, see Section 4.12.3 on page 142. CHNnDAT holds a new result when NEW bit is also set. The value at reset is undefined. Range: 0 to 255 (0 to V_{FSx}); 8-bit, unsigned value with 1 LSB = V_{FSx} / 256; V_{FSx} is V_{FSE}, V_{FSI} or V_{FSS}, according to the channel number.
1-0	**Reserved.**

4.0 Embedded Controller Modules (Continued)

4.12.6 Usage Hints

Power Supply and Layout Guidelines

For more information, see Section 3.1.3 on page 47.

Power Consumption

ADC power consumption from the analog supply (AV_{CC}) is practically zero if the ADC is disabled by setting ADCEN in ADCCNF register to 0.

When the ADC is enabled, the current consumption depends on the ADC clock frequency. To minimize consumption of current, disable the ADC when not in use; see details in "Enabling and Disabling the ADC" on page 143.

Back-Drive Protection

To maintain the high performance of the analog circuits, AD0-3 pins are not back-drive protected. Therefore, the voltage applied to these pins must be within the AGND to AV_{CC} + 0.5V range; otherwise, the device may be damaged. External circuits should not drive currents into these pins when the WPC8763L is not powered up.

Connecting an External Voltage Reference

If ADC functionality is not used (i.e., the ADC is disabled), connect the VREF pin to AGND.

Measuring Out of Range Voltages

The ADC is capable of measuring positive input voltages from 0V to V_{FSE} or V_{FSS}. Input voltages outside this range should either be divided or level-shifted, as required.

For positive input voltages higher than V_{FSE} or V_{FSS}, place a resistor divider in front of the WPC8763L input pin; see Figure 33a. The divider should be calculated so that its output is lower than the full-scale value (V_{FSE} or V_{FSS}) for the maximum input signal voltage.

For negative input voltages, place a resistive level-shifter in front of the WPC8763L input pin; see Figure 33b. The level-shifter should be calculated so that its output is higher than 0V for the minimum input signal voltage.

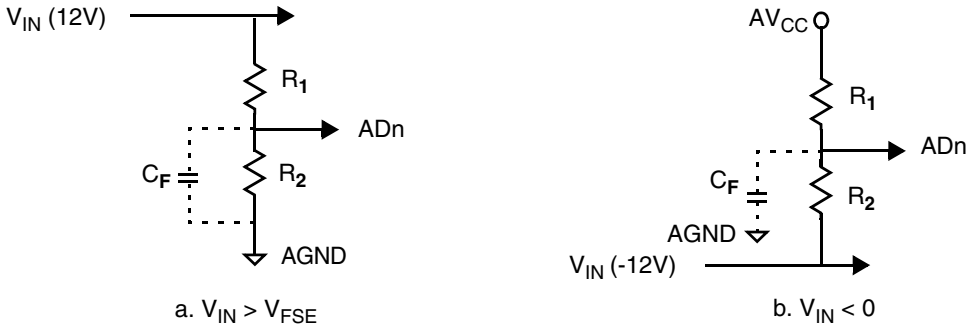


Figure 33. Measurement of Positive and Negative Voltages

Filtering the Noise on Input Signals

Noise may be coupled to the input signal for various reasons, including close proximity to digital circuits. The slow change rate of the input signals makes it possible to use an external Low Pass Filter (LPF). This can be implemented by placing a capacitor (C_F) between the divider output and AGND, as shown in Figure 33. The cutoff frequency of this LPF should be at least 22 times the maximum signal frequency required to be measured with 10-bit accuracy (a smaller capacitor may be used when a lower accuracy is acceptable). The following formula demonstrates the calculation of the components.

$$f_{(-3dB)} = 1 / (2 * \pi * R_{eq} * C_F)$$

where: $R_{eq} = (R_1 * R_2) / (R_1 + R_2)$

4.0 Embedded Controller Modules (Continued)

4.13 DIGITAL TO ANALOG CONVERTER (DAC)

The DAC converts digital input values to analog signals. The DAC supports two channels for handling up to two independent conversions in parallel.

4.13.1 Features

- 8-bit resolution
- Independent 2-channel D/A converter
- Fast settling time, 1 µs typical, on 50 pF capacitive load
- Output range from AGND to AV_{CC}
- Independent enable/disable for each channel
- Both converters can be automatically disabled in Idle or Deep Idle mode
- Low power consumption when enabled; minimal power consumption when disabled
- Outputs drive 0V when disabled

4.13.2 Functional Description

The DAC has two independent digital-to-analog converters. Each converter generates an output in the range of 0V to AV_{CC}, with 8-bit resolution. The converters drive the two output pins DA0-1, as shown in Figure 35. An output impedance of 3 KΩ allows a settling time of about 1 µs on a 50 pF load.

When a DAC channel is enabled, its output is defined by the value written to its DACDATn register. DACDAT0-1 control DA0-1, respectively. The maximum output voltage is (255 ÷ 256) * AV_{CC} and is obtained for a value of FFh. The minimum output, 0V, is obtained for a value of 00h.

The reference voltage of the converters is the AV_{CC} analog power supply voltage. This allows full swing of the outputs from 0V to nearly AV_{CC}.

After reset, all four channels are disabled, and the voltage on the DA0-1 outputs is 0V.

In Idle or Deep Idle mode, the DAC channels may be enabled or disabled (to reduce power consumption). Two control modes are provided:

- Automatic disable of all channels on entering Idle or Deep Idle mode
- Selective disable of channels by software before entering Idle or Deep Idle mode

4.13.3 D/A Conversion

Output Signal

The DAC performs a linear conversion of the input digital value (DACDATA) to a unipolar analog output signal, relative to the analog ground pin (AGND).

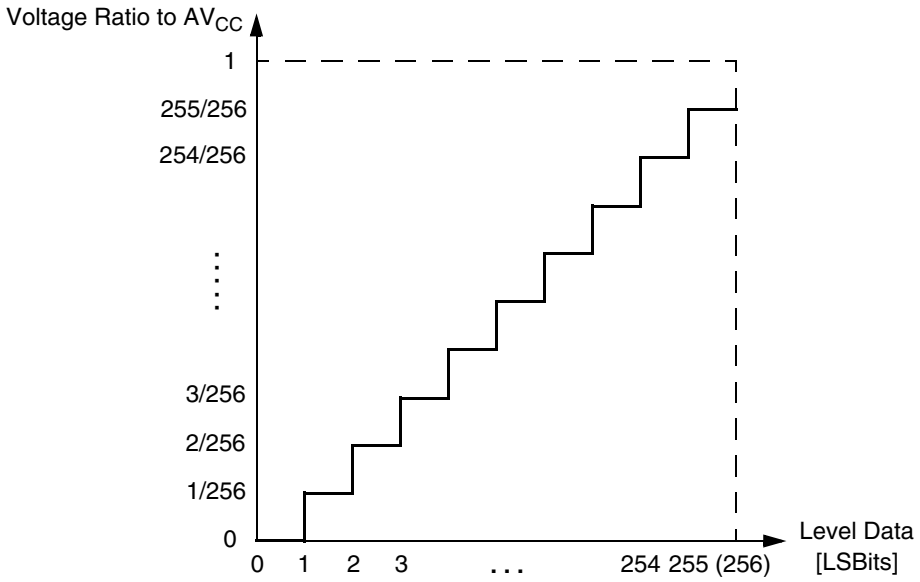


Figure 34. Channel Data to Output Voltage Ratio Conversion

4.0 Embedded Controller Modules (Continued)

When the value of DACDATA field is 00h, the respective output has an output signal of 0V (AGND). When the value of DACDATA field is FFh, the respective output has an output signal of (255/256) * AV_{CC}. For other values, as shown in Figure 34:

$$V_{OUT} = (DACDATA) * (AV_{CC} / 256)$$

Reference Voltage

The analog output voltages are converted relative to a reference voltage. The reference voltage of the converters is the analog power supply. To ensure good signal quality at the WPC8763L output, use a low-noise analog power supply.

Conversion Time

When a DAC channel is enabled, the conversion is started by writing to DACDATn registers. The output settling time is defined as the time the DAC requires to get to within 1/2 LSB of the final value; see "Output Settling Time" on page 154.

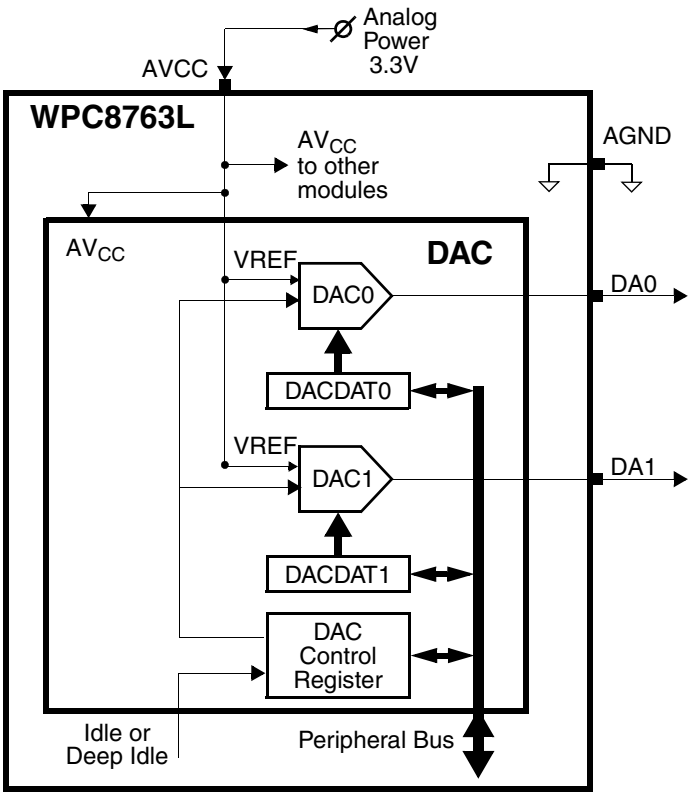


Figure 35. DAC Functional Diagram

4.13.4 Operation

Initializing the DAC

The WPC8763L wakes up after power-up with all the D/A channels disabled (DACEN0-1 bits in DACCTRL register are cleared to 0). In this state, all DAC activities are halted and the DAC current consumption is lowered to zero.

Enabling and Disabling the DAC

Enabling the DAC. Each channel of the DAC is enabled independently by setting its DACEN bit. After being enable, the DAC channel settles to the value stored in DACDATn register after the specified settling time.

Disabling the DAC. To reduce current consumption, the DAC channels can be disabled independently by clearing the corresponding DACENn (n=0 to 1) bit in DACCTRL register. In this case, the output pin drives 0V even if the respective DACDATn register does not contain 00h.

All DAC channels are automatically disabled when entering Idle or Deep Idle mode, if ENIDLE bit in DACCTRL register is cleared to 0. This happens regardless of the state of DACENn (n=0 to 1) bit in DACCTRL register. In this case, the DA0-1 outputs drive 0V.

If ENIDLE bit is set to 1, entering the Idle or Deep Idle mode does not affect DAC operation, and DA0-1 outputs drive the voltage level set by DACDATn (n=0 to 1) registers.

4.0 Embedded Controller Modules (Continued)

4.13.5 DAC Registers

The DAC interfaces with the core using one control and two data registers. These registers are reset by Core Domain reset. For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

DAC Register Map

Location	Mnemonic	Register Name	Size	Type
FF F880h	DACCTRL	DAC Control	Byte	R/W
FF F882h	DACDAT0	DAC Data Channel 0	Byte	R/W
FF F884h	DACDAT1	DAC Data Channel 1	Byte	R/W

DAC Control Register (DACCTRL)

DACCTRL controls the operation of the DAC module. It is cleared (00h) on reset.

Location: FF F880h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**ENIDLE**	**Reserved**		**DACEN1**	**DACEN0**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-5	**Reserved.**
4	**ENIDLE (Enable in Idle).** Controls the DAn (n = 0 to 1) outputs in Idle or Deep Idle mode. 0: Disabled - DAn outputs drive 0V (default). 1: Enabled - DAn outputs according to DACENn bits and DACDATn registers.
3-2	**Reserved.**
1	**DACEN1 (DAC Channel 1 Enable).** Same as DACEN0 bit description, using DA1 output and DACDAT1 register.
0	**DACEN0 (DAC Channel 0 Enable).** Enables the DAC channel. The DA0 output pin drives a voltage level according to the value written into the corresponding DACDAT0 register. When cleared, the DA0 output pin drives 0V. 0: Disabled (default). 1: Enabled.

DAC Data Channel 0-1 Registers (DACDAT0-1)

DACDAT0-3 hold the data to be loaded into Channels 0-1 of the DAC. They are cleared (00h) on reset, but are not affected when the respective channel is disabled.

Location: Channel 0 - FF F882h
Channel 1 - FF F884h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**DACDATA**							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**DACDATA (DAC Data).** 8-bit unsigned binary value used for the D/A operation.

4.0 Embedded Controller Modules (Continued)

4.13.6 Usage Hints

Power Consumption

When a channel is enabled and no load is connected, the DAC current consumption depends on the value set in DACDAT register. Minimum current is consumed when the data is 00h; maximum current is consumed when the data is 55h.

DAC Output Protection

To maintain high performance of the analog circuits, the DAn (n = 0 to 1) pins are not back-drive protected. Therefore, the voltage applied to these pins must be within the AGND to AV_{CC} range; otherwise, the device may be damaged.

External circuits should not drive currents into these pins when the WPC8763L power is off because this may cause the internal Power-Up reset circuit to fail.

DAC Load Protection

Before setting DACEN0-1 in DACCTRL register to 1, DACDATn registers (n=0 to 1) must be initialized either to 00h or according to the required output level.

Output Voltage Accuracy

Besides the intrinsic accuracy of the D/A channels; see Section 8.4.2 on page 286, the output voltage accuracy directly depends on the accuracy of the AV_{CC} power supply, which serves as reference voltage. To improve the accuracy of the output voltage, the actual AV_{CC} value, measured by the ADC module (see Section 4.12 on page 140), should be used when computing the value of DACDATA field in DACDATn registers; see Section 4.13.3 on page 151.

The external load on DA0-1 pins may also affect the final output voltage of the DAC. Since the output resistance of these pins is typically 3 KΩ, use high-impedance loads; if high-accuracy or high-output currents are required, use external analog drivers.

For worst case calculation, if the output resistance is 4 KΩ (maximum limit), the external load (R_L) must not be lower than 2 MΩ; see Figure 36. In this case, the error caused by the load is lower than 1/2 LSB.

To work with loads of 3 KΩ (1 mA at 3V) with an error lower than 1/2 LSB, the output resistance of the external driver should be lower than:

$$R_{OEXT} < 3\ K\Omega\ /\ (2*256) = 5.8\Omega$$

Output Settling Time

The DAC output settling time depends on the external load characteristics and the required accuracy. Figure 36 shows the equivalent circuit used for evaluating DAC behavior. Each DAC output has a typical output impedance of 3 KΩ. For example, if the total load is a 50 pF capacitor only, the output settles to 1/2 LSB within 1 μs. The total load capacitance comprises the analog output capacitance (C_{AO}) and the external load capacitance (C_L).

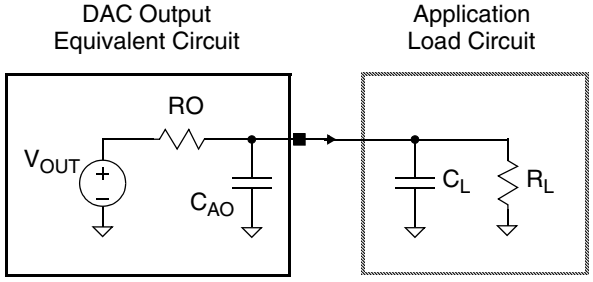


Figure 36. DAC Output Equivalent Circuit

Filtering Noise on Output Signals

Output signals may have unwanted noise caused by nearby digital circuits. When using slow changing signals in a noisy environment, a low-pass filter (LPF) may be added externally. The LPF, which can be implemented as a simple R-C circuit, may also be required in applications where the DAC outputs control sensitive circuits such as audio amplifiers. The cutoff frequency of the LPF should be above the required signal frequency.

4.0 Embedded Controller Modules (Continued)

4.14 SMBUS (SMB) INTERFACE

The WPC8763L includes two SMBus Interface (SMB) modules. The registers of each module are prefixed with SMBn, and the signal names are suffixed with 'n', where 'n' is module number 1 or 2.

Each SMBus interface is a two-wire serial interface that is compatible with the SMBus physical layer. The module can be configured as either a bus master or slave and can maintain bidirectional communication with multiple master and slave devices. As a slave device, the SMB module can issue a request to become the bus master.

The SMB interface provides full support for a two-wire SMBus synchronous serial interface. It permits easy interfacing to a wide range of low-cost memories and I/O devices, including EEPROMs, SRAMs, timers, A/D converters, D/A converters, clock chips and peripheral drivers.

4.14.1 Features

- SMBus compliant
- SMBus master
- SMBus slave, supporting:
  - — Two user-defined addresses
  - — Global (broadcast) address
  - — ARP address
- Supports polling- or interrupt-controlled operation
- Generates a wake-up signal on detection of a Start condition in Idle or Deep Idle mode
- Optional internal pull-up on SDAn and SCLn pins

4.14.2 Functional Description

The SMBus protocol uses a two-wire interface for bidirectional communication between the ICs connected to the bus. The two interface lines are the Serial Data Line (SDLn) and the Serial Clock Line (SCLn). These lines should be connected to a positive supply via a pull-up resistor and remain high even when the bus is idle.

The SMBus protocol supports multiple master and slave transmitters and receivers. Each IC has a unique address and can operate as a transmitter or a receiver. Some ICs are receivers only.

During data transactions, the master device initiates the transaction, generates the clock signal and terminates the transaction. For example, when the SMB initiates a data transaction with an attached SMBus-compliant peripheral, the SMB becomes the master. When the peripheral responds and transmits data to the SMB, their master/slave relationship (data transaction initiator and clock generator) is unchanged even though their transmitter/receiver functions are reversed.

Data Transactions

One data bit is transferred during each clock pulse. Data is sampled during the high state of the serial clock (SCLn). Consequently, throughout the clock's high period, the data should remain stable; see Figure 37. Any change on the SDAn line while SCLn is in high state during a transaction causes the current transaction to abort. New data should be sent during the low SCLn state. This protocol permits a single data line to transfer both command/control information and data, using the synchronous serial clock.

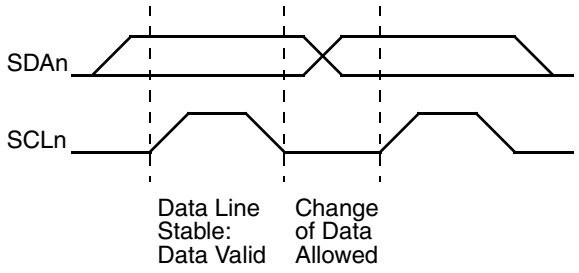


Figure 37. Bit Transfer

Each data transaction is composed of a Start condition, a number of byte transfers (set by the software) and a Stop condition to terminate the transaction. Each byte is transferred with the most significant bit first. An Acknowledge signal must follow each byte (8 bits). The following sections provide further details of this process.

At each clock cycle, the slave can stall the master while it handles the previous data or prepares new data. The slave does this, for each bit transferred or on a byte boundary, by holding SCLn low to extend the clock low period. Typically, slaves extend the first clock cycle of a transfer if a byte read has not yet been stored or if the next byte to be transmitted is not yet ready. Some microcontrollers with limited hardware support for the SMBus extend the access after each bit, thus giving the software time to handle this bit.

4.0 Embedded Controller Modules (Continued)

The SMBus master generates Start and Stop conditions (control codes). Following a Start condition, the bus is considered busy. It retains this status for a given amount of time following a Stop condition. A high-to-low transition of the data line (SDAn) while the clock (SCLn) is high indicates a Start condition. A low-to-high transition of the SDAn line while the SCLn is high indicates a Stop condition (Figure 38).

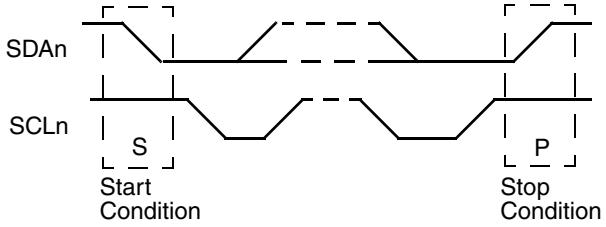


Figure 38. Start and Stop Conditions

In addition to the first Start condition, a Repeated Start condition can be generated in the middle of a transaction. This allows either another device to be accessed or a change in the direction of the data transfer.

Acknowledge Cycle

The Acknowledge cycle consists of two signals:

- Acknowledge Clock pulse - sent by the master with each byte transferred
- Acknowledge signal - sent by the receiving device; see Figure 39

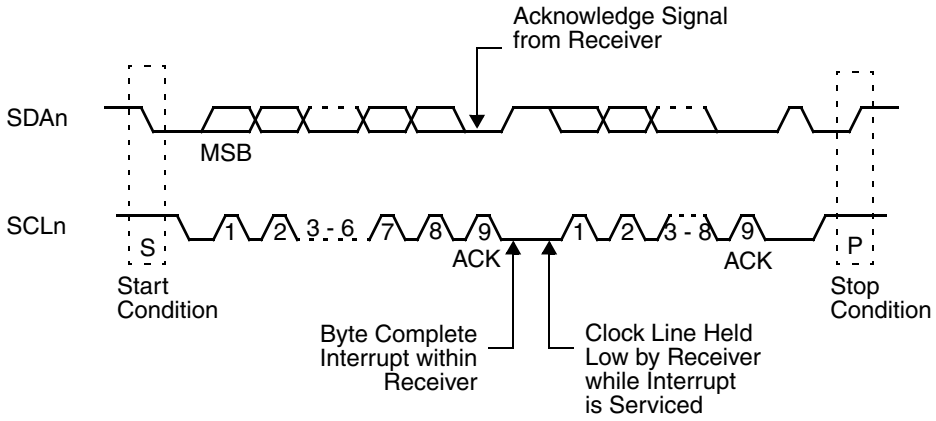


Figure 39. SMBus Data Transaction

The master generates the Acknowledge clock pulse on the ninth clock pulse of the byte transfer. The transmitter releases the SDAn line (permitting it to go high) to allow the receiver to send the Acknowledge signal.The receiver pulls down the SDAn line during the Acknowledge clock pulse, thus signalling that it has correctly received the last data byte and is ready to receive the next byte. Figure 40 shows the Acknowledge cycle.

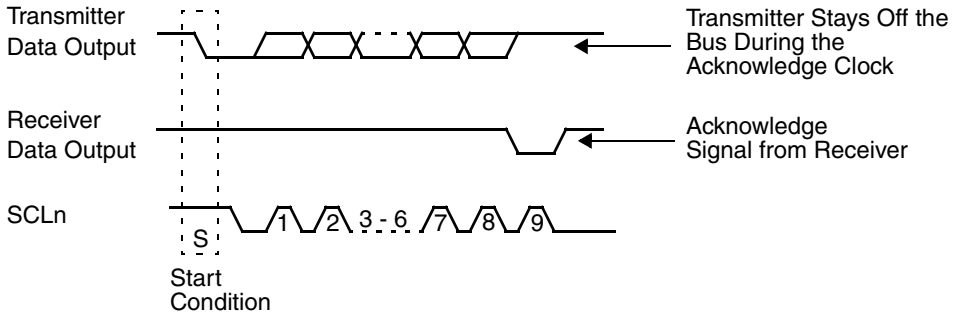


Figure 40. SMBus Acknowledge Cycle

"Acknowledge After Every Byte" Rule

The master generates an Acknowledge clock pulse after each byte transfer. The receiver sends an Acknowledge signal after every byte is received.

4.0 Embedded Controller Modules (Continued)

There are two exceptions to the “acknowledge after every byte” rule:

- When the master is the receiver, it must indicate to the transmitter an “end of data” by not acknowledging (negative acknowledge) the last byte clocked out of the slave. This negative acknowledge still includes the Acknowledge clock pulse (generated by the master), but the SDAn line is not pulled down.
- When the receiver is full or otherwise occupied or if a problem occurs, it sends a negative acknowledge to indicate that it cannot accept additional data bytes.

Addressing Transfer Formats

Each device on the bus has a unique address. Before any data is transmitted, the master transmits the address of the slave being addressed. The slave device should send an Acknowledge signal on the SDAn line once it recognizes its address.

The address consists of the first seven bits after a Start condition. The eighth bit contains the direction of the data transfer (R/$\overline{W}$). A low-to-high transition during a SCLn high period indicates a Stop condition and ends the transaction of SDAn (Figure 41).

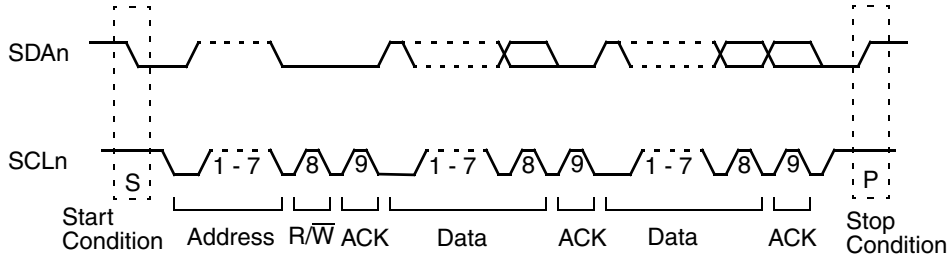


Figure 41. A Complete SMBus Data Transaction

When the address is sent, each device in the system compares this address with its own. If there is a match, the device considers itself addressed and sends an Acknowledge signal. Depending on the state of the R/$\overline{W}$ bit (1=read, 0=write), the device acts as a transmitter or a receiver.

The SMBus protocol allows a general call address to be sent to all slaves connected to the bus. The first byte sent specifies the general call address (00h); the second byte specifies the general call meaning (for example, “Write slave address by software only”). Slaves that require data acknowledge the call and become slave receivers; other slaves ignore the call.

Arbitration on the Bus

Multiple master devices on the bus require arbitration between their conflicting bus access demands. Control of the bus is initially determined according to address bits and clock cycle. If more than one master tries to address the same slave, data comparisons determine the outcome of this arbitration. In Master mode, the device immediately aborts a transaction if the value sampled on the SDAn line differs from the value driven by the device; however, if the master is receiving data, the lines may be driven low by the slave without causing an abort.

The SCLn signal is monitored for clock synchronization to allow the slave to stall the bus. The actual clock period is the longest of either the master or slave stall period. The clock high period is determined by the master with the shortest clock high period, on the bus.

When an abort occurs during address transmission, a master that identifies the conflict should give up the bus and switch to Slave mode. It should then continue to sample SDAn to see if it is being addressed by the winning master on the bus.

4.14.3 Master Mode

Requesting Bus Mastership

An SMBus transaction starts with a master device requesting bus mastership. It asserts a Start condition, followed by the address of the device it wants to access. If this transaction is completed successfully, the software may assume that the device has become the bus master.

For the device to become the bus master, the software should perform the following steps:

1. Configure INTEN in SMBnCTL1 register to the desired operation mode (Polling or Interrupt) and set START in the same register. This causes the SMB to issue a Start condition on the SMBus as soon as the SMBus is free (some conditions, such as when BB in SMBnCST register is set to 0, can delay the Start condition). It then stalls the bus by holding SCLn low.
2. If a bus conflict is detected (i.e., if another device pulls down the SCLn signal before the WPC8763L does), BER in SMBnST register is set.
3. If there is no bus conflict, MASTER and SDAST in SMBnST register are set.
4. If INTEN in SMBnCTL1 register is set, and either BER or SDAST in SMBnST register is set, an interrupt is sent to the core.

4.0 Embedded Controller Modules (Continued)

Sending the Address Byte

Once the WPC8763L is the active master of the SMBus (MASTER in SMBnST register is set), it can send the address on the bus. The address sent should **not** be any of the following:

- The WPC8763L's own address, as defined by ADDR in SMBnADDR1 register, if SAEN in SMBnADDR1 is set to 1.
- The WPC8763L's own address, as defined by ADDR in SMBnADDR2 register, if SAEN in SMBnADDR2 is set to 1.
- The global call address, if GCMATCH in SMBnCST register is set.
- The ARP address, if ARPMATCH in SMBnST register is set.

To send the address byte, use the following sequence:

1. For a receive transaction where the software requires only one byte of data, the software should set ACK in SMBnCTL1 register. If only an address needs to be sent (e.g., for quick read/write protocols) or if the device requires stall for some other reason, set STASTRE in SMBnCTL1 register to 1.
2. Write the address byte (7-bit target device address) and the direction bit to SMBnSDA register. This causes the module to generate a transaction. At the end of this transaction, the acknowledge bit received is copied to NEGACK in SMBnST register. During the transaction, the SDAn and SCLn lines are continuously checked for conflict with other devices. If a conflict is detected, the transaction is aborted, BER in SMBnST register is set and MASTER in SMBnST register is cleared.
3. If STASTRE in SMBnCTL1 register is set and the transaction was completed successfully (i.e., both BER and NEGACK in SMBnST register are cleared), STASTR in SMBnST register is set. In this case, the SMB stalls any further SMBus operations (i.e., holds SCLn low). If INTEN in SMBnCTL1 register is set, it also sends an interrupt to the core.
4. If the requested direction is transmit and the start transaction was completed successfully (i.e., neither NEGACK nor BER in SMBnST register is set and no other master has accessed the device), SDAST in SMBnST register is set to indicate that the module awaits attention.
5. If the requested direction is receive, the start transaction was completed successfully and STASTRE in SMBnCTL1 register is cleared, the module starts receiving the first byte automatically.
6. Check that both BER and NEGACK in SMBnST register are cleared. If INTEN in SMBnCTL1 register is set, an interrupt is generated when either BER or NEGACK is set.

Master Transmit

After becoming the bus master, the device can start transmitting data on the SMBus.

To transmit a byte, the software should:

1. Check that BER and NEGACK bits in SMBnST register are cleared and SDAST bit is set. In addition, if STASTRE bit in SMBnCTL1 register is set, clear STASTR bit in SMBnST register by writing 1 to it.
2. Write the data byte to be transmitted to SMBnSDA register.

When the slave responds with a negative acknowledge, NEGACK in SMBnST register is set and SDAST in SMBnST register remains cleared. In this case, if INTEN bit in SMBnCTL1 register is set, an interrupt is sent to the core.

Master Receive

After becoming the bus master, the device can start receiving data on the SMBus.

To receive a byte, the software should:

1. Check that SDAST bit in SMBnST register is set and BER bit is cleared. In addition, if STASTRE bit in SMBnCTL1 register is set, clear STASTR bit in SMBnST register by writing 1 to it.
2. If the next byte is the last byte that should be read, set ACK bit in SMBnCTL1 register to 1. This causes a negative acknowledge to be sent.
3. Read the data byte from SMBnSDA register.

Master Stop

To end a transaction, set STOP in SMBnCTL1 register before clearing the current stall flag (i.e., SDAST, NEGACK or STASTR in SMBnST register). This causes the module to send a Stop condition immediately and to clear STOP in SMBnCTL1 register. A Stop condition may be issued only when the WPC8763L is the active bus master (MASTER in SMBnST register is set to 1).

Master Bus Stall

The SMB module can stall the SMBus between transfers while waiting for the core's response. The SMBus is stalled by holding the SCLn signal low after the acknowledge cycle. Note that this is interpreted as the start of the following bus operation. The user must make sure that the next operation is prepared before the flag that causes the bus stall is cleared.

4.0 Embedded Controller Modules (Continued)

The flags that can cause a bus stall in Master mode are:

- Negative acknowledge after sending a byte (NEGACK in SMBnST register is set to 1).
- SDAST in SMBnST register is set to 1.
- STASTRE in SMBnCTL1 register is set to 1 after a successful start (STASTR in SMBnST is set to 1).

Repeated Start

A repeated start is performed when the WPC8763L is already the bus master (MASTER in SMBnST register is set). In this case, the SMBus is stalled and the SMB module awaits core handling due to a negative acknowledge (NEGACK in SMBnST register is set to 1), an empty buffer (SDAST in SMBnST is set to 1) and/or a stall after start (STASTR in SMBnST is set to 1).

For a repeated start:

1. Set START in SMBnCTL1 register to 1.
2. In Master Receive mode, read the last data item from SMBnSDA.
3. Follow the address send sequence, as described in "Sending the Address Byte" on page 158.
4. If the SMB is awaiting handling because STASTR in SMBnST is set to 1, clear it only after writing the requested address and direction to SMBnSDA.

Master Error Detection

The SMB detects an illegal Start or Stop condition (i.e., a Start or Stop condition within the data transfer or the acknowledge cycle) and a conflict on the data lines of the SMBus. If an illegal condition is detected, BER is set and Master mode is exited (MASTER in SMBnST register is cleared).

Bus Idle Error Recovery

When a request to become the active bus master (start) or a restart operation fails, BER in SMBnST register is set to indicate the error. In some cases, both the WPC8763L and the other device may identify the failure and leave the bus idle. In this case, the start sequence may not finish and the SMBus may remain deadlocked.

To recover from deadlock, use the following sequence:

1. Clear BER in SMBnST register and BB in SMBnCST register.
2. Wait for a time-out period to check that there is no other active master on the bus (i.e., BB in SMBnCST remains cleared).
3. Disable and then re-enable the SMB to put it in non-addressed Slave mode. (This completely resets the module.)

It is possible that some of the slaves did not identify the bus error. To recover from this error, use the following sequence:

1. Request bus mastership for the SMB module.
2. If SMB module becomes the bus master, generate a Start condition.
3. Send an address byte.
4. Generate a Stop condition (this synchronizes all the slaves).

4.14.4 Slave Mode

A slave device waits in Idle mode for a master to initiate a bus transaction. When an SMB module is enabled, it acts as a slave device (i.e., MASTER in SMBnST register is cleared).

Once a Start condition on the bus is detected, the WPC8763L checks whether the address sent by the current master matches any of the following possibilities:

- The ADDR value in SMBnADDR1 register, if SAEN in this register is set to 1.
- The ADDR value in SMBnADDR2 register, if SAEN in this register is set to 1.
- The global call address (00h), if GCMEM in SMBnCTL1 register is set to 1.
- The global ARP address (110 0001b), if ARPMEN in SMBnCTL3 register is set to 1.

The address match is checked even when MASTER in SMBnST register is set. If a bus conflict (on SDAn or SCLn) is detected, BER is set, MASTER is cleared and the WPC8763L continues to search the received message for a match.

If an address match (ADDR, ARP or global call) is detected:

1. The WPC8763L asserts its SDAn pin during the acknowledge cycle, regardless of the setting of ACK bit in SMBnCTL1 register.
2. MATCH in SMBnCST register, MATCHAF in SMBnST register (or GCMATCH if it is a global call address match, or ARPMATCH if it is an ARP address match) and NMATCH in SMBnST register are set. If XMIT in SMBnST register is set (i.e., Slave Transmit mode), SDAST in the same register is also set to indicate that the buffer is empty.
3. If INTEN in SMBnCTL1 register is set, an interrupt is generated if both INTEN and NMINTE in SMBnCTL1 register are set.
4. The software then reads XMIT in SMBnST register to identify the direction requested by the master device; it then clears NMATCH in the same register so that future byte transfers are identified as data bytes.

Slave Receive and Transmit

Slave Receive and Transmit are performed after a match is detected and the data transfer direction is identified. After a byte transfer, the SMB module extends the acknowledge clock until the software reads or writes SMBnSDA register. The receive and transmit sequences are identical to those used in the master routine.

Slave Bus Stall

When operating as a slave, the WPC8763L stalls the SMBus by extending the first clock cycle of the subsequent byte in the following cases:

- SDAST in SMBnST register is set.
- NMATCH in SMBnST register and NMINTE in SMBnCTL1 register are set.

Slave Error Detection

The SMB detects illegal Start and Stop conditions (occurring during the data transfer or during the acknowledge cycle) on the SMBus. If a transaction is terminated by a Stop condition as a result of a stalled clock (SCLn), the Stop condition is illegal. When an illegal Start or Stop condition is detected, BER is set and MATCH and GMATCH and ARPMATCH are cleared, setting the module as an unaddressed slave.

4.14.5 Power-Down

When the WPC8763L is in Idle or Deep Idle mode, the SMB module is not active but retains its registers. An exception is the SMBnCTL1 register, which is reset in Idle or Deep Idle mode. If wake-up is enabled (by the SMBxEEN bit in SMB_EEN register) on detection of a Start condition, a wake-up signal is issued to the MIWU; see Section 4.6.2 on page 88. The wake-up signal can be used to switch the WPC8763L to Active mode.

Following the Start condition that woke up the WPC8763L, the SMB module cannot check the address byte for a match. The SMB responds with a negative acknowledge (does not drive the SDA line during the ACK, SCL cycle, regardless of the setting of ACK bit in SMBnCTL1 register). The master device should resend both the Start condition and the address after the WPC8763L has had time to wake up.

Before entering Idle or Deep Idle mode, make sure that BUSY in SMBnCST register is inactive. This guarantees that the WPC8763L does not stop responding after it acknowledges an address that was sent.

4.14.6 SDA and SCL Pin Configuration

The SDAn and SCLn are open collector signals that the user can choose to enable or disable. SDAn and SCLn also have internal pull-up resistors that the user can enable. For more information about configuring these pins, see Section 2.3 on page 23 and DEVPU0 register in Section 2.6 on page 35.

4.14.7 SMB Clock Frequency Configuration

The SMB module enables the user to set the SMBus clock frequency. The SCLn clock period is set by SCLFRQ in SMBnCTL2 and SMBnCTL3 registers. The clock low period can be extended by stall periods initiated by the SMB module or by another SMBus device. In case of a conflict with another bus master, a shorter clock high period may be forced by the other bus master until the conflict is resolved.

4.14.8 SMB Registers

All SMB registers are powered by V_{CC}. They are set to their default values by Core Domain reset unless otherwise specified in the register description below.

SMB Register Map

Location	Mnemonic	Register Name	Size	Type
SMBn Base + 00h	SMBnSDA	SMB Serial Data	Byte	R/W
SMBn Base + 02h	SMBnST	SMB Status	Byte	Varies per bit
SMBn Base + 04h	SMBnCST	SMB Control Status	Byte	Varies per bit
SMBn Base + 06h	SMBnCTL1	SMB Control 1	Byte	R/W
SMBn Base + 08h	SMBnADDR1	SMB Own Address 1	Byte	R/W
SMBn Base + 0Ah	SMBnCTL2	SMB Control 2	Byte	R/W
SMBn Base + 0Ch	SMBnADDR2	SMB Own Address 2	Byte	R/W
SMBn Base + 0Eh	SMBnCTL3	SMB Control 3	Byte	R/W

4.0 Embedded Controller Modules (Continued)

SMB Serial Data Register (SMBnSDA)

SMBnSDA is a shift register used to transmit and receive data. The most significant bit is transmitted/received first and the least significant bit is transmitted/received last. Reading or writing to the SMBnSDA is allowed either when SDAST in SMBnST register is set or, in case of repeated starts, after setting the START bit. An attempt to access the register in other cases may produce unpredictable results.

Locations: SMB1 - FF F500h

SMB2 - FF F540h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Data**							

Bit	Description
7-0	**Data.**

SMB Status Register (SMBnST)

SMBnST maintains current SMB status. Some of its bits may be cleared by software. It is cleared (00h) on reset and when the module is disabled.

Locations: SMB1 - FF F502h

SMB2 - FF F542h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**SLVSTP**	**SDAST**	**BER**	**NEGACK**	**STASTR**	**NMATCH**	**MASTER**	**XMIT**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W1C	**SLVSTP (Slave Stop).** When set, indicates that a Stop condition was detected after a slave transfer (i.e., after a slave transfer in which MATCH, ARPMATCH or GCMATCH was set). Writing 1 to SLVSTP clears it. It is also cleared when the module is disabled. Writing 0 to SLVSTP is ignored.
6	RO	**SDAST (SDA Status).** When set, indicates that the SDA data register is waiting for data (Transmit mode - master or slave) or holds data that should be read (Receive mode - master or slave). This bit is cleared when reading from SMBnSDA register during a receive or when written to during a transmit. When START in the SMBnCTL1 is set, reading SMBnSDA register does not clear SDAST. This enables the SMB to send a repeated start in Master Receive mode.
5	R/W1C	**BER (Bus Error).** Set by the hardware either when an invalid Start/Stop condition is detected (i.e., during data transfer or acknowledge cycle) or when an arbitration problem is detected (such as a stalled clock transaction). Writing 1 to BER clears it. It is also cleared when the module is disabled. Writing 0 to BER is ignored.
4	R/W1C	**NEGACK (Negative Acknowledge).** Set by hardware when a transmission is not acknowledged on the ninth clock (in this case, SDAST is not set). After a Stop condition, writing 1 to NEGACK clears it. It is also cleared when the module is disabled. Writing 0 to NEGACK is ignored.
3	R/W1C	**STASTR (Stall After Start).** Set by the successful completion of the sending of an address (i.e., a Start condition sent without a bus error or negative acknowledge) if STASTRE in SMBnCTL1 register is set. This bit is ignored in Slave mode. When STASTR is set, it stalls the SMBus (by pulling down the SCL line) and suspends any further action on the bus (e.g., receiving the first byte in Master Receive mode). In addition, if INTEN in SMBnCTL1 register is set, it also causes the SMB module to send an interrupt to the core. Writing 1 to STASTR clears it. It is also cleared when the module is disabled. Writing 0 to STASTR has no effect.
2	R/W1C	**NMATCH (New Match).** Is set when the address byte following a Start condition or a repeated start causes an address match, ARP address match or a global call match. NMATCH is cleared by writing 1 to it. Writing 0 to NMATCH is ignored. If INTEN in SMBnCTL1 register is set, an interrupt is sent when this bit is set.

Bit	Type	Description
1	RO	**MASTER (Master Mode).** 0: Arbitration loss (BER is set) or Stop condition occurred (default). 1: SMB in Master mode (successful request for bus mastership).
0	RO	**XMIT (Transmit Mode).** 0: SMB not in master/slave Transmit mode (default). 1: SMB in master/slave Transmit mode.

SMB Control Status Register (SMBnCST)

SMBnCST maintains current SMB status and controls several SMB module functions. The non-reserved bits are cleared (00h) on reset and when the module is disabled. An exception is TSDA, which reflects the current value of the SDA pin.

Locations: SMB1 - FF F504h

SMB2 - FF F544h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**ARPMATCH**	**MATCHAF**	**TGSCL**	**TSDA**	**GCMATCH**	**MATCH**	**BB**	**BUSY**
Reset	0	0	0	X[1]	0	0	0	0

1. According to the current value of the SDA pin.

Bit	Type	Description
7	RO	**ARPMATCH (ARP Address Match).** In Slave mode, set (1) when ARPMEN in SMBnCTL3 register is set and the address byte (the first byte transferred after a Start condition) is 110 0001b. It is cleared by a Start, Repeated Start or Stop condition (including illegal Start or Stop condition).
6	RO	**MATCHAF (Match Address Field).** When MATCH bit is set, MATCHAF bit indicates which slave address was matched. MATCHAF is cleared for a match with ADDR in SMBnADDR1 register and is set for a match with ADDR in SMBnADDR2 register.
5	R/W	**TGSCL (Toggle SCL Line).** Enables toggling the SCL line during the process of error recovery. When the SDA line is low, writing 1 to this bit toggles the SCL line for one cycle. Writing 1 to TGSCL is ignored if any of the following conditions is true: • The SDA line is high. • The SMB module is in Slave mode and a transaction is performed on the bus. TGSCL bit is cleared when the SCL line toggle is completed.
4	RO	**TSDA (Test SDA Line).** Reads the current value of the SDA line. This bit can be used while recovering from an error condition in which the SDA line is constantly pulled low by a slave that went out of synch. Data written to this bit is ignored.
3	RO	**GCMATCH (Global Call Match).** In Slave mode, set (1) when GCMEN in SMBnCTL1 register is set and the address byte (the first byte transferred after a Start condition) is 00h. It is cleared by a Start Repeated Start or Stop condition (including illegal Start or Stop condition).
2	RO	**MATCH (Address Match).** In Slave mode, set (1) when SAEN in SMBnADDR1 or SMBnADDR2 register is set and the first seven bits of the address byte (the first byte transferred after a Start condition) match the 7-bit ADDR field in the respective register. It is cleared by Start, Repeated Start or Stop condition (including illegal Start or Stop condition).
1	R/W1C	**BB (Bus Busy).** When set (1), indicates the bus is busy. It is set either when the bus is active (i.e., a low level on either SDA or SCL) or by a Start condition. It is cleared when the module is disabled or on detection of a valid Stop condition or by writing 1 to this bit; see Section 4.14.9 on page 166 for a description of the use of this bit.

Bit	Type	Description
0	RO	**BUSY.** When set (1), indicates that the SMB module is in one of the following states: – Generating a Start condition. – In Master mode (MASTER in SMBnST register is set). – In Slave mode (MATCH, GMATCH or ARPMATCH in SMBnCST register is set). – In the period between detecting a Start condition and completing the reception of the address byte; after this, the SMB either becomes not busy or enters Slave mode. The BUSY bit is cleared by the completion of any of the above states or by disabling the module. It should always be written 0.

SMB Control Register 1 (SMBnCTL1)

SMBnCTL1 is byte-wide, read/write. It configures and controls the SMB module. It is cleared (00h) on reset.

Locations: SMB1 - FF F506h
SMB2 - FF F546h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**STASTRE**	**NMINTE**	**GCMEN**	**ACK**	**Reserved**	**INTEN**	**STOP**	**START**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**STASTRE (Stall After Start Enable).** When set (1), enables the Stall After Start mechanism. In this case, the SMB stalls the bus after the address byte. When STASTRE is cleared, STASTR bit in SMBnST register cannot be set.
6	**NMINTE (New Match Interrupt Enable).** When set, enables the interrupt on a new match (i.e., when NMATCH in SMBnST register is set). The interrupt is issued only if INTEN in SMBnCTL1 register is set.
5	**GCMEN (Global Call Match Enable).** When set, enables the matching of an incoming address byte to the general call address (Start condition followed by address byte of 00h) while the SMB is in Slave mode. When cleared, the SMB does not respond to a global call.
4	**ACK (Acknowledge).** When acting as a receiver, holds the value of the next acknowledge cycle. It should be set when a negative acknowledge must be issued on the next byte. Cleared (0) after each acknowledge cycle. It cannot be reset by software. In Transmit mode, ACK is ignored. It is also ignored when it is generated for the Address Byte, in Slave Receive mode.
3	**Reserved.**
2	**INTEN (Interrupt Enable).** When cleared (0), the SMB interrupt is disabled. When set, interrupts are enabled. An interrupt is generated (the interrupt signals to the ICU are high) on one of the following events: – An address match is detected (NMATCH in SMBnST register is set to1 and NMINTE in SMBnCTL1 register is set to 1). – A bus error occurs (BER in SMBnST register is set to 1). – A negative acknowledge is received after sending a byte (NEGACK in SMBnST register is set to 1). – Acknowledgment of each transaction (same as the hardware set of SDAST in SMBnST). – In Master mode, if STASTRE in SMBnCTL1 register is set to 1 after a successful start (STASTR in SMBnST register is set to 1). – Detection of a Stop condition while in Slave mode (SLVSTP in SMBnST register is set to 1).
1	**STOP.** In Master mode, setting this bit generates a Stop condition, which completes or aborts the current message transfer. This bit clears itself after the Stop condition is generated.

4.0 Embedded Controller Modules (Continued)

Bit	Description
0	**START.** Should be set to generate a Start condition on the SMBus. – If the WPC8763L is not the active bus master (MASTER in SMBnST register is set to 0), setting START generates a Start condition as soon as the SMBus is free (BB in SMBnCST register is set to 0). An address transmission sequence should then be performed. – If the WPC8763L is the active master of the bus (MASTER in SMBnST register is set to 1), when START is set, a write to SMBnSDA register generates a Start condition. SMBnSDA data, containing the slave address and the transfer direction, is then transmitted. In case of a Repeated Start condition, the set bit can be used to switch the direction of the data flow between the master and the slave or to choose another slave device. without requiring a Stop condition in either case. This bit is cleared either when the Start condition is sent or on detection of a bus error (BER in SMBnST register is set to 1). START should be set only when in Master mode or when requesting Master mode.

SMB Own Address Register (SMBnADDR1, SMBnADDR2)

SMBnADDR1 and SMBnADDR2 hold the module SMBus addresses. They are cleared (00h) on reset.

SMBnADDR1, SMBnADDR2 (respectively):

Locations: SMB1 - FF F508h, FF F50Ch

SMB2 - FF F548h, FF F54Ch

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**SAEN**	**ADDR**						
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**SAEN (Slave Address Enable).** When set (1), indicates that the ADDR field holds a valid address and enables the match of ADDR to an incoming address byte. When cleared (default), the SMB does not check for an address match.
6-0	**ADDR (Address).** Holds the 7-bit SMBus address of the WPC8763L. When in Slave mode, the first seven bits received after a Start condition are compared to this field (the first bit received is compared to bit 6, the next bit to bit 5 and so on until the last bit, which is compared to bit 0). If the address field matches the received data and SAEN in SMBnADDR register is set to 1, a match is declared. If SAEN is 1, ADDR value in each SMBnADDR register must be different; equal ADDR values may produce unpredictable results.

4.0 Embedded Controller Modules (Continued)

SMB Control Register 2 (SMBnCTL2)

SMBnCTL2 enables/disables the module and determines the SMB clock rate. It is cleared (00h) on reset and while the module is disabled (i.e., when ENABLE in SMBnCTL2 register is 0).

Locations: SMB1 - FF F50Ah

SMB2 - FF F54Ah

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**SCLFRQ6-0**							**ENABLE**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-1	**SCLFRQ6-0 (SCL Frequency bits 6 through 0).** This field, together with SCLFRQ8-7 in SMBCTL3 register, defines the SCL period (low time and high time) when the WPC8763L serves as a bus master. The clock low time and high time are defined as follows: $t_{SCL} = 4 * SCLFRQ * t_{CLK}$ $t_{SCLL} = t_{SCLH} = t_{SCL} / 2$ where t_{CLK} is the WPC8763L clock cycle when in Active mode; see Section 8.5.3 on page 290. SCLFRQ may be programed to values in the range of 00 0001000b (8) through 11 1111111b (511). Values outside this range give unpredictable results.
0	**ENABLE.** When set, the SMB module is enabled. When cleared, the SMB module is disabled, SMBnCTL1, SMBnST and SMBnCST are cleared and the internal clock are halted.

SMB Control Register 3 (SMBnCTL3)

SMBnCTL3 expands the clock prescaler field and enables the match to ARP addresses. It is cleared (00h) on reset.

Locations: SMB1 - FF F50Eh

SMB2 - FF F54Eh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**ARPMEN**	**SCLFRQ8-7**	
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-3	**Reserved.**
2	**ARPMEN (ARP Match Enable).** When set, enables the matching of an incoming address byte to the SMBus ARP address (110 0001b) while the SMBus is in Slave mode. When cleared, the SMBus does not respond to an ARP address.
1-0	**SCLFRQ8-7 (SCL Frequency bits 8 and 7).** Extends SCLFRQ field and is concatenated with bits 0-6, which are part of SMBnCTL2 register; see the SCLFRQ6-0 description in SMBnCTL2 for detailed use of SCLFRQ.

4.0 Embedded Controller Modules (Continued)

4.14.9 Usage Hints

1. When the SMB is disabled, BB in SMBnCST register is cleared. After the SMB is enabled (by setting ENABLE in SMBnCTL2 register), the SMB must synchronize with the bus activity status before issuing a request to become the bus master for the first time. To do this, the software should check that there is no activity on the bus by checking the BB bit after the time-out period allowed by the bus.
2. When waking up from Idle or Deep Idle mode, before checking MATCH in SMBnCST register, check BUSY in SMBnCST register to make sure that the address transaction is completed.
3. The BB bit can help solve a deadlock in which two or more devices detect a usage conflict on the bus and both cease being bus masters at the same time. In this situation, the BB bits of both devices are active (because each “detects” another master currently performing a transaction, while in fact there is no transaction). This potentially causes the bus to stay locked until a device on the bus sends a Stop condition (through STOP in SMBnCTL1 register).

   The BB bit allows the software to monitor bus usage so that it can detect whether the bus remains unused over a certain period of time (BB remains cleared). It also prevents sending a STOP signal in the middle of the transaction of another device on the bus.
4. In some cases, the bus may get stuck with the SCL and/or SDA lines active, such as when an erroneous Start or Stop condition occurs in the middle of a slave receive session.

   If the SCL line is stalled active, the module that holds the bus must release it.

   If the SDA line is stalled active, the sequence below releases the bus (note: Normally, SCL may be toggled only by the bus master; this sequence is a recovery mechanism which is an exception that should only be used if there is no other master on the bus):
   a. Disable and re-enable the SMB module to put it in non-addressed Slave mode.
   b. Set START in SMBnCTL1 register to attempt to issue a Start condition.
   c. Check if the SDA line is active (low) by reading TSDA in SMBnCST register. If it is active, issue a single SCL cycle by writing 1 to TGSCL in the same register. If it is not active, go to step “e”.
   d. Check if MASTER in SMBnST register is set, which indicates that the Start condition was sent. If it is not set, repeat steps “c” and “d” until the SDA is released.
   e. Clear BB. This enables START to be executed. Continue according to “Bus Idle Error Recovery” on page 159.

4.0 Embedded Controller Modules (Continued)

4.15 SENSORPATH BUS INTERFACE (SPB)

4.15.1 Overview

The SPB module contains the hardware necessary for efficient communication with SensorPath devices. The SensorPath transactions are initiated by the core. The SPB module handles all the physical aspects of the transaction including symbol encoding and decoding, attention request detection, reset generation and detection.

4.15.2 Functional Description

SensorPath Bus

The SPB module communicates with SensorPath devices such as GPIO expanders and Non-Volatile memories via the SensorPath bus, as defined in the *SensorPath Specification Revision TBD, date TBD*. The SPB module is the bus master on the SensorPath bus and supports Normal (asynchronous) mode.

The SensorPath bus uses timing encoding based on the SPB_CLK internal clock to provide the following types of bit signals (symbols) over the bus (the SPB module supports all five types):

- Data bit 0
- Data bit 1
- Start bit
- Attention request
- Reset

The SPB module supports SensorPath bus speeds of x1 and x4. Adjust to the maximum SensorPath bus speed allowed by the connected SensorPath devices by changing the SPB_CLK internal clock accordingly. The SPB_CLK clock is generated from the FCLK signal (HFCG module) by dividing it in two stages, as follows:

1. By the Clock prescaler – The division factor is set by CLK_PSC field in CLK_CFG register. Its value must be set by firmware according to the current FCLK frequency (see Table 21 on page 182) to generate a 2.88 MHz clock signal; see Table 19 on page 175.
2. By the Bus Speed Divider – The division factor is set by BSPEED field in SPB_CFG register. Its value must be set by firmware according to the current speed of the SensorPath bus (x1 and x4) to generate the SPB_CLK clock signal (the basic clock of the SPB module).

   Since the “Reset” timing is fixed, regardless of the current speed of the SensorPath bus, the Bus Speed Divider is forced to generate the x1 SPB_CLK (360 KHz) during the transmission of the “Reset” symbol.

In terms of the number of SPB_CLK cycles, the detection levels and transmit timing of the SensorPath symbols are independent of the current speed of the SensorPath bus.

To filter out noise from the SWD signal, the received signal is debounced. The debouncer ignores any level changes shorter than three cycles of the DBC_CLK clock. The DBC_CLK clock is generated from the FCLK signal (HFCG module) by dividing it in two stages, as follows:

1. By the Debounce prescaler – The division factor is set by the DB_PSC bit in CLK_CFG register. Its value must be set by firmware according to the current FMCLK frequency; see Table 19 on page 175.
2. By the Bus Speed Divider – The division factor is set by the BSPEED field in SPB_CFG register. Its value must be set by firmware according to the current speed of the SensorPath bus (x1 and x4). For more information, see Table 19 on page 175.

The decoded “Start Bit”, “Attention Request” and “Reset” symbols, as well as the “Acknowledge” (ACK) and Parity (EP) data bit symbols, are handled by the SPB Controller; the “Data Bit 0” and “Data Bit 1” symbols are stored in the 32-bit shift register. In a similar way, the encoded “Start Bit” and “Reset” symbols, as well as the “Acknowledge” (ACK) and Parity (EP) data bit symbols, are generated by the SPB Controller; the “Data Bit 0” and “Data Bit 1” symbols come from the 32-bit shift register.

If a SensorPath data bit or “Start Bit” collides with an “Attention Request”, the data bit or “Start Bit” is resent automatically.

A “Reset” operation on the bus can be initiated by either the SPB module or a SensorPath device by generating a “Reset” symbol. After sending a “Reset” symbol or after detecting a “Reset” symbol sent by a SensorPath device, the SPB module sends a sequence of 16 data bits with a value of 0 without a preceding start bit.

SensorPath Device Access

The access to the internal registers of the SensorPath devices is initiated by the core via the SPB module.

Write Access. For a write access, the firmware must write the operation code (in OPER field), the number of the target SensorPath device (in DEV_NUM field), the register address within the device (in the INT_ADDR field), the number of relevant data bytes (in the DAT_SIZE field) and the data to be written (in the SPB_DAT0 - SPB_DAT3 registers). Then, the firmware must set the OP_EXEC bit to 1 to trigger the execution of the write transaction.

Note that since the data in the SPB_DAT0 - SPB_DAT3 registers is right-aligned, only the relevant data bytes (as specified in DAT_SIZE field) must be written; the remaining data bytes are ignored.

4.0 Embedded Controller Modules (Continued)

When the SensorPath write transaction is finished, the OP_DONE flag in SPB_BUST register is set to 1. In addition, if errors were detected, one or more error flags in SPB_BUST register are set to 1.

Read Access. For a read access, the firmware must write the operation code (in OPER field), the number of the target SensorPath device (in DEV_NUM field), the register address within the device (in the INT_ADDR field) and the number of relevant data bytes (in the DAT_SIZE field). Then, the firmware must set the OP_EXEC bit to 1 to trigger the execution of the read transaction.

When the SensorPath read transaction is finished, the OP_DONE flag in SPB_BUST register is set to 1. In addition, if errors were detected, one or more error flags in SPB_BUST register are set to 1.

Then, the read data can be retrieved from the SPB_DAT0 - SPB_DAT3 registers. Note that since the data in the SPB_DAT0 - SPB_DAT3 registers is right-aligned, only the relevant data bytes (as specified in DAT_SIZE field) may be read; the remaining bytes contain undefined data.

Bus Reset. To reset the bus interface of the SensorPath devices, the firmware can use one of the following options:

- To generate a bus reset while the bus is idle, the firmware must write the operation code (in OPER field) and set the OP_EXEC bit to 1 to trigger the execution of the reset operation.
- To generate a bus reset while another transaction is being performed (the bus reset overrides the current transaction), the firmware must set the FRESET bit to 1 to trigger the execution of the reset operation. Then, FRESET returns automatically to 0 and the “Reset” symbol is sent.

In both cases, after both the “Reset” symbol and sequence of 16 ‘0’ bits are sent, BRST bit in SPB_BUST register is set to 1. In this case OP_DONE bit remains 0.

Attention Request. When a SensorPath device either contains new data or has detected an error, it sends an “Attention Request” symbol to the SPB module. When the SPB module detects the “Attention Request”, it sets the ATTN bit in SPB_BUST register to 1. In response, the firmware should scan all the devices connected to the SensorPath bus to identify the source of the “Attention Request”.

Selecting Bus Speed

After the SPB module is reset, the SensorPath bus speed is set to the basic speed, x1 (360 KHz clock). This speed is selected by BSPEED field in SPB_CFG register (this field is reset to ‘000’).

To set a higher SensorPath bus speed, the firmware must proceed as follows:

1. Reset the SensorPath bus by setting the FRESET bit in SPC_CTL register.
2. Wait for BRST bit in SPB_BUST register to become 1 (this indicates that the “Reset” operation was completed).
3. Optionally, read the *Device Number Register* in all the connected SensorPath devices to identify the highest speed supported by **all** the devices. This operation should be performed only if it is not known which bus speed is supported by devices connected to the SensorPath bus.
4. Write this speed (or the new desired speed) to BSPEED field in SPB_CFG register.
5. Reset for the second time the SensorPath bus by writing 1 to FRESET bit in SPC_CTL register. The SensorPath devices use the sequence of 16 ‘0’ bits to tune their clocks to the new SensorPath bus speed.

SPB Module Power Control

To minimize power consumption when the SPB module is idle, the module supports a low power operation mode. This is the default operation mode after the SPB module is reset. The firmware must enable the SPB module (set it to Active mode) by setting the SPB_EN bit in SPB_CFG register to 1.

When the SPB module is enabled:

- All the SPB module registers can be accessed by the core.
- The SPB module can perform transactions over the SensorPath bus.
- An SPB event can be generated only by IRQ (SPB event generation by wake-up is disabled).

Entering Low Power Mode. To enter this mode from Active mode, the firmware must disable the SPB module by setting SPB_EN bit in SPB_CFG register to 0. If a transaction is currently being performed, SPB_EN bit remains 1.

Note: If SPB_EN bit cannot be set to 0, the firmware must retry writing 0 to SPB_EN bit until it accepts this value (this indicates that the SPB module was disabled).

When the SPB module is disabled:

- The module clocks (FCLK and its derivatives) are stopped. The core clock (CLK), used for registers access, and LFCLK clock, used for wake-up, continue to run.
- Only SPB_WKST, SPB_CFG and CLK_CFG registers can be accessed by the core.
- The output buffer of the SWD signal is floated (TRI-STATE mode).
- An SPB event can be generated only by wake-up (SPB event generation by IRQ is disabled).

Exiting Low Power Mode. To return from the low power operation mode, the SPB module senses the SWD signal to detect a low level. This is caused by either an “Attention Request” or a “Reset” sent by one of the SensorPath devices.

When a low level is detected, the SPB module performs the following:

- Generates an active wake-up signal at the SPB event output. This is connected to the MIWU, which both wakes up the system (from Idle or Deep Idle states) and generates an interrupt to the ICU module.
- Sets the WK_BUSY bit in SPB_WKST register to 1.
- Checks the duration of the low level at SWD input to identify the cause of the wake-up (either “Attention Request” or “Reset”).

After identifying the source of the wake-up, the SPB module clears the WK_BUSY bit and updates WK_RST bit in SPB_WKST register, accordingly. In response to the interrupt caused by the SPB wake-up event, the firmware must do the following:

1. Wait for the WK_BUSY bit to become 0.
2. Check the cause of the wake-up by reading WK_RST bit.
3. Clear WK_RST bit (if set) by writing 1 to it.
4. Enable the SPB module by setting SPB_EN bit to 1.

If the wake-up was caused by a “Reset” (WK_RST bit was 1), the firmware must set FRESET bit in SPB_CTL register to 1, thus triggering a reset operation.

If the wake-up was caused by an “Attention Request” (WK_RST bit was 0), the firmware must scan the SensorPath devices, to identify which device requested “attention”.

SPB Module Event Generation

An SPB event is generated either by an SPB wake-up (when the module is disabled) or by an SPB IRQ (when the module is enabled). The SPB event is connected to WUI41 input of the MIWU (see Section 4.2 on page 66), which both wakes up the system and generates an interrupt to the ICU module (INT7).

The SPB IRQ is generated if one of the following events occurs:

- SensorPath operation completed (i.e., flagged by OP_DONE bit in SPB_BUST register).
- “Attention Request” was sent by one of the SensorPath devices (i.e., flagged by ATTN bit in SPB_BUST register).
- “Reset” operation, initiated by either the SPB module or a SensorPath device, was completed –(i.e., flagged by BRST bit in SPB_BUST register).
- SensorPath bus failure detection (i.e., flagged by BUS_FAIL, LONG_RST or ERR_DET bit in SPB_BUST register).

Each of the above events has an enable bit in SPB_CFG register, which controls interrupt generation.

4.15.3 SPB Module Registers

For a summary of the abbreviations used for Register Type, see “Register Abbreviations and Access” on page 20.

All SPB registers are powered by V_{CC} and are set to their default values by Core Domain reset.

Register Map

Location	Mnemonic	Register Name	Size	Type
FF F140h	SPB_CTL	SPB Control	Byte	Varies per bit or RUP
FF F141h	SPB_BUST	SPB Bus Status	Byte	R/W1C or RUP
FF F142h	SPB_WKST	SPB Wake Status	Byte	Varies per bit
FF F143h	SPB_INTA	SPB Internal Address	Byte	R/W or RUP
FF F144h	SPB_DAT_0	SPB Data Byte 0	Byte	R/W or RUP
FF F145h	SPB_DAT_1	SPB Data Byte 1	Byte	R/W or RUP
FF F146h	SPB_DAT_2	SPB Data Byte 2	Byte	R/W or RUP
FF F147h	SPB_DAT_3	SPB Data Byte 3	Byte	R/W or RUP
FF F148h	SPB_CFG	SPB Configuration	Byte	R/W
FF F149h	CLK_CFG	SPB Clock Configuration	Byte	R/W

4.0 Embedded Controller Modules (Continued)

SPB Control Register (SPB_CTL)

SPB_CTL controls the SPB module operation and selects the accessed SensorPath device number.

Location: FF F140h

Type: Varies per bit or RUP

Bit	7	6	5	4	3	2	1	0
Name	**FRESET**	**OP_EXEC**	**OPER**			**DEV_NUM**		
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	W1S	**FRESET (SensorPath Force Reset).** Initiates a “Reset” operation on the SensorPath bus even if a SensorPath transaction is currently being executed (the “Reset” overrides the transaction). In this case, OP_EXEC bit and OPER field are ignored. Writing 1 to this bit initiates a “Reset” operation; writing 0 is ignored. Always returns 0 when read. 0: No action (default). 1: Force a “Reset” operation (“Reset” symbol and sequence of 16 data bits with a value of 0).
6	WO	**OP_EXEC (Operation Execute).** Writing 1 to this bit triggers the execution of the operation selected by OPER field. If the SensorPath bus is busy executing another operation, setting OP_EXEC bit to 1 is ignored. Always returns 0 when read. 0: No action (default). 1: Execute the operation set in OPER field.
5-3	R/W	**OPER (Operation Select).** Selects the SensorPath bus operation to be executed. **Bits 5 4 3 Operation** 0 0 0: Reserved (default). 0 0 1: Write data to SensorPath Device register. 0 1 0: Read data from SensorPath Device register. 0 1 1: Reserved. 1 0 0: Reset the SensorPath bus (“Reset” symbol and sequence of 16 data bits with a value of 0). Other: Reserved.
2-0	R/W	**DEV_NUM (Target Device Number).** The SensorPath device number to be sent on the SensorPath bus during the transaction operation selected by OPER field. 0: Reserved (default). 1-7: Device number on the SensorPath bus.

SPB Bus Status Register (SPB_BUST)

SPB_BUST indicates the status of the SensorPath bus transaction.

Location: FF F141h

Type: R/W1C or RUP

Bit	7	6	5	4	3	2	1	0
Name	**BUS_FAIL**	**OP_DONE**	**PERR**	**ACK_ERR**	**LONG_RST**	**ERR_DET**	**ATTN**	**BRST**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W1C	**BUS_FAIL (Bus Failure).** When set to 1, indicates that the SWD signal is “stuck” at high level. If this bit cannot be cleared to 0, the firmware must flag it as a major failure of the SensorPath bus. Writing 1 clears this bit; writing 0 is ignored. 0: Normal bus operation (default). 1: SWD signal is “stuck” at high level.

4.0 Embedded Controller Modules (Continued)

Bit	Type	Description
6	R/W1C	**OP_DONE (Operation Done Status).** When set to 1, indicates that the execution of the operation selected by OPER field was performed (finished). Writing 1 clears this bit; writing 0 is ignored. 0: Bit reset, or operation in process (default). 1: Operation executed.
5	R/W1C	**PERR (Parity Error).** When set to 1, indicates that a parity error was detected during a transaction on the SensorPath bus. Writing 1 clears this bit; writing 0 is ignored. 0: Correct transaction (default). 1: Parity error detected during transaction execution.
4	R/W1C	**ACK_ERR (Acknowledge Error).** When set to 1, indicates that ACK was not received by the SPB module during a transaction on the SensorPath bus. Writing 1 clears this bit; writing 0 is ignored. 0: Correct transaction (default). 1: ACK missing at the end of the transaction.
3	R/W1C	**LONG_RST (Long "Reset" Detected).** When set to 1, indicates that a "Reset" symbol with a duration longer than the minimum specified value was detected on the SensorPath bus. The firmware must then check if BRST bit is set to 1 (indicating that the "Reset" operation ended and the SensorPath bus was released). If BRST remains 0 after more than 500 ms, the SWD signal is "stuck" at low level and the firmware must flag this as a major failure of the SensorPath bus. Writing 1 clears this bit; writing 0 is ignored. 0: Bit reset, or no "Reset" operation was detected on the SensorPath bus (default). 1: A "Reset" operation (initiated either by SPB module or by a SensorPath device) was detected on the SensorPath bus.
2	R/W1C	**ERR_DET (Bus Error Detect).** When set to 1, indicates that the detected SWD symbol is different from the sent symbol. ERR_DET is not affected by an "Attention Request" or a slave-generated "Reset". It therefore indicates an invalid state of the SensorPath bus. Writing 1 clears this bit; writing 0 is ignored. 0: Normal bus operation (default). 1: SensorPath bus error detected.
1	R/W1C	**ATTN (Attention Request).** When set to 1, indicates that an "Attention Request" was received by the SPB module from a SensorPath device. Writing 1 clears this bit; writing 0 is ignored. 0: Normal operation (default). 1: "Attention Request" was received.
0	R/W1C	**BRST (SensorPath Bus Reset).** When set to 1, indicates that the SensorPath bus was reset either by the SPB module or by a SensorPath device. Set only at the end of the SensorPath bus "Reset" operation (after the sequence of 16 '0' bits was sent), indicating that the reset process was completed and the bus is active. Writing 1 clears this bit; writing 0 is ignored. 0: Normal operation (default). 1: SensorPath bus was reset.

4.0 Embedded Controller Modules (Continued)

SPB Wake Status Register (SPB_WKST)

SPB_WKST indicates the wake-up status of the SPB module.

Location: FF F142h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**WK_BUSY**	**WK_RST**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	RO	**WK_BUSY (Busy After Wake-Up).** When set to 1, indicates that following a wake-up condition at SWD signal, the wake-up detector is in the process of identifying the source of the wake-up (either “Attention Request” or “Reset”). 0: No wake-up, or WK_RST flag is valid (default). 1: Wake-up detector busy (the WK_RST flag is not valid yet).
6	R/W1C	**WK_RST (Wake-Up Source Detect).** Indicates the source of the last wake-up condition. Not valid while WK_BUSY bit is 1. Writing 1 clears this bit; writing 0 is ignored. 0: Bit reset, or wake-up by “Attention Request (default). 1: Wake-up by “Reset”.
5-0		**Reserved.**

SPB Internal Address Register (SPB_INTA)

SPB_INTA selects the transaction data size and the register address within the accessed SensorPath device.

Location: FF F143h

Type: R/W or RUP

Bit	7	6	5	4	3	2	1	0
Name	**DAT_SIZE**		**INT_ADDR**					
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-6	R/W	**DAT_SIZE (Data Size).** The data size (number of active bytes) of the target SensorPath device register accessed by the transaction. **Bits 7 6 Data Size** 0 0: 1 active data byte (8 bits) - data in SPB_DAT_0 (default). 0 1: 2 active data bytes (16 bits) - data in SPB_DAT_0, SPB_DAT_1. 1 0: 3 active data bytes (24 bits) - data in SPB_DAT_0, SPB_DAT_1, SPB_DAT_2. 1 1: 4 active data bytes (32 bits) - data in SPB_DAT_0, SPB_DAT_1, SPB_DAT_2, SPB_DAT_3.
5-0	R/W	**INT_ADDR (Internal Address).** The internal address of the target SensorPath device accessed by the transaction. Valid values are 00h (default) to 3Fh.

4.0 Embedded Controller Modules (Continued)

SPB Data Byte 0 Register (SPB_DAT_0)

SPB_DAT_0 contains data byte 0, which is written to the target SensorPath device or read from the target.

Location: FF F144h

Type: R/W or RUP

Bit	7	6	5	4	3	2	1	0
Name	**Data Bits 7-0**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Data Bits 7-0.** Bits 7-0 of the SensorPath Device register. The data bits are right-aligned (data bit 0 maps to the LSB of the SensorPath Device register). Only the number of bytes defined in DAT_SIZE field in SPB_INTA register is required to be written or read by the firmware. When read, the bytes beyond the number defined in DAT_SIZE might contain invalid data.

SPB Data Byte 1 Register (SPB_DAT_1)

SPB_DAT_1 contains data byte 1, which is written to the target SensorPath device or read from the target.

Location: FF F145h

Type: R/W or RUP

Bit	7	6	5	4	3	2	1	0
Name	**Data Bits 15-8**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Data Bits 15-8.** Bits 15-8 of the SensorPath Device register.

SPB Data Byte 2 Register (SPB_DAT_2)

SPB_DAT_2 contains data byte 2, which is written to the target SensorPath device or read from the target.

Location: FF F146h

Type: R/W or RUP

Bit	7	6	5	4	3	2	1	0
Name	**Data Bits 23-16**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Data Bits 23-16.** Bits 23-16 of the SensorPath Device register.

4.0 Embedded Controller Modules (Continued)

SPB Data Byte 3 Register (SPB_DAT_3)

SPB_DAT_3 contains data byte 3, which is written to the target SensorPath device or read from the target.

Location: FF F147h

Type: R/W or RUP

Bit	7	6	5	4	3	2	1	0
Name	**Data Bits 31-24**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**Data Bits 31-24.** Bits 31-24 of the SensorPath Device register.

SPB Configuration Register (SPB_CFG)

SPB_CFG configures the SPB module.

Location: FF F148h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**SPB_EN**	**DONE_IEN**	**ATT_IEN**	**BRST_IEN**	**FAIL_IEN**	**BSPEED**		
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W	**SPB_EN (SPB Module Enable).** When set to 1, enables the operation of the SPB module; see "SPB Module Power Control". When set to 0, only the SPB_WKST, SPB_CFG and CLK_CFG registers can be accessed by the core. If 0 is written to SPB_EN while a transaction is currently being performed, its value does not change. If 0 is written when the bus is idle, its value changes to 0. 0: SPB module disabled (default). 1: SPB module enabled.
6	R/W	**DONE_IEN ("Done" Interrupt Enable).** When set to 1, enables interrupt generation (IRQ) by a set OP_DONE bit in SPB_BUST register. 0: Interrupt disabled (default). 1: Interrupt enabled.
5	R/W	**ATT_IEN ("Attention" Interrupt Enable).** When set to 1, enables interrupt generation (IRQ) by a set ATTN bit in SPB_BUST register. 0: Interrupt disabled (default). 1: Interrupt enabled.
4	R/W	**BRST_IEN ("Bus Reset" Interrupt Enable).** When set to 1, enables interrupt generation (IRQ) by a set BRST bit in SPB_BUST register. 0: Interrupt disabled (default). 1: Interrupt enabled.
3	R/W	**FAIL_IEN ("Failure" Interrupt Enable).** When set to 1, enables interrupt generation (IRQ) when one of the following bits in SPB_BUST register is set: BUS_FAIL, LONG_RST, ERR_DET. 0: Interrupt disabled (default). 1: Interrupt enabled.

Bit	Type	Description
2-0	R/W	**BSPEED (Bus Speed).** Selects the current speed of the SensorPath bus by controlling the value of the Bus Speed Divider. **Bits 2 1 0 Speed** 0 0 0: Basic speed - SPB_CLK frequency is 360 KHz (default). 0 1 0: Speed x4 - SPB_CLK frequency is 1.44 MHz. Other: Reserved.

SPB Clock Configuration Register (CLK_CFG)

CLK_CFG configures the SPB module clock dividers.

Location: FF F149h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**DB_PSC**	**Reserved**	**CLK_PSC**				
Reset	0	0	0	0	0	1	1	0

Bit	Type	Description
7		**Reserved.**
6	R/W	**DB_PSC (Debounce Prescaler).** Controls the division factor of the Debounce Prescaler; see Table 19. 0: Divide FCLK by 1 (default). 1: Divide FCLK by 2.
5		**Reserved.**
4-0	R/W	**CLK_PSC (Clock Prescaler).** Controls the division factor of the Clock Prescaler to generate a 2.88 MHz clock signal from the FCLK clock. FCLK is divided by CLK_PSC + 1 (see Table 19). The default value is 6 (i.e., FCLK is divided by 7).

Table 19. Clock Prescaler Settings

FMCLK Freq.	50 MHz			40 MHz		33 MHz	
FPRED Field[1,2]	0h	1h	2h	0h	1h	0h	1h
FCLK Freq.	50 MHz	25 MHz	16.67 MHz	40 MHz	20 MHz	33 MHz	16.5 MHz
DB_PSC Bit	1	0	0	1	0	0	0
CLK_PSC Field	10h	08h	05h	0Dh	06h	0Ah	05h

1. FPRED field in HFCGP register; see Section 4.19.3 on page 183.
2. Other FPRED field values are allowed but their use is **not** recommended when the SPB module is active.

4.0 Embedded Controller Modules (Continued)

4.16 ON-CHIP RAM

The WPC8763L contains 4 Kbytes of on-chip RAM.

The system RAM can be byte, word or double-word accessed. Each system RAM read or write is one cycle long and does not include any wait states; see Section 1.4.1 on page 19 for the system RAM memory map.

4.17 ON-CHIP ROM

The WPC8763L contains 1 Kbyte of on-chip ROM. The core code execution starts at address 0 in the ROM.

The on-chip ROM contains the core boot code (the Booter program; see Chapter 7 on page 275). ROM contents cannot be modified.

4.0 Embedded Controller Modules (Continued)

4.18 POWER MANAGEMENT CONTROLLER (PMC)

The Power Management Controller (PMC) module improves the efficiency of WPC8763L operation by adjusting the chip's power consumption to the level of activity required by the application. This module works together with the High-Frequency Clock Generator (HCFG) and the core to control the activity of the WPC8763L. It also interacts with the Multi-Input Wake-Up (MIWU), Interrupt Control Unit (ICU) and the debugger interface for wake-up events.

4.18.1 Features

- Core domain power modes:
  - Active
  - Idle
  - Deep Idle
  - Power Off
- Power mode switching by software and/or hardware control.
- High-frequency clock source Enable/Disable control.
- The other core domain modules are controlled via power mode indications.

4.18.2 The Core Domain Power Modes

Table 20 summarizes the main properties of the various modes and shows the activity levels of the various clocks while in these power states.

Table 20. Core Domain Power Mode Summary

Mode	V_{CC} Supply	LFCG	LFCLK	HFCG Module	FMCLK, CLK	RAM and Registers	I/O Pins
Active[1]	On	On	Active	On	Active	Functional	Functional
Idle	On	On	Active	On	Stopped	Preserved[2]	Most preserved; a few functional
Deep Idle	On	On	Active	Off	Off	Preserved[2]	Most preserved; a few functional
Power Off	Off	On	Off	Off	Off	Undefined	High impedance

1. When no core activity is required, the core can execute the WAIT instruction while in Active mode to reduce power consumption. This state is referred to as "Active Executing WAIT" in some places in the specification, but it is not a separate power state as far as WPC8763L clocks are concerned.
2. MIWU, TWD and PMC modules are functional; RAM and the registers of other modules are preserved.

Active Mode

In Active mode, the core and the external flash operate at the frequency generated by the HFCG. This frequency may be changed dynamically using the HFCGP register in the HFCG module.

In this mode, power consumption can be lowered by selectively disabling modules and/or by the core executing the WAIT instruction. When WAIT is executed, the core stops executing new instructions until it receives an interrupt signal.

After Core Domain reset, Watchdog Warm reset or Debugger Warm reset, the WPC8763L is in Active mode.

Idle Mode

In Idle mode, the clock is stopped for most of the core domain modules, but the HFCG is enabled. Only the PMC and a limited number of core domain modules (such as MIWU and TWD) continue to operate at the LFCLK rate. When required, they can wake up the core domain and resume instruction execution by the core.

For modules that are active in Idle mode, details of their activity are included in each module description.

Wake-up events are generated by the MIWU according to the enabled internal and external events.

Deep Idle Mode

The Deep Idle mode is similar to the Idle mode, with the difference that the HFCG is disabled (thus the clock is stopped for most of the core domain).

4.0 Embedded Controller Modules (Continued)

Power Off Mode

When V_{CC} and AV_{CC} power are both turned off, the core domain reaches its lowest activity level. The contents of the memories (except for the on-chip ROM) and registers are lost. When both V_{CC} and AV_{CC} power return, a V_{CC} Power-Up reset is generated.

4.18.3 Switching Between Power Modes

The switching from one power mode to another is performed as described below. Figure 42 shows the power modes of the core domain and the transitions between them.

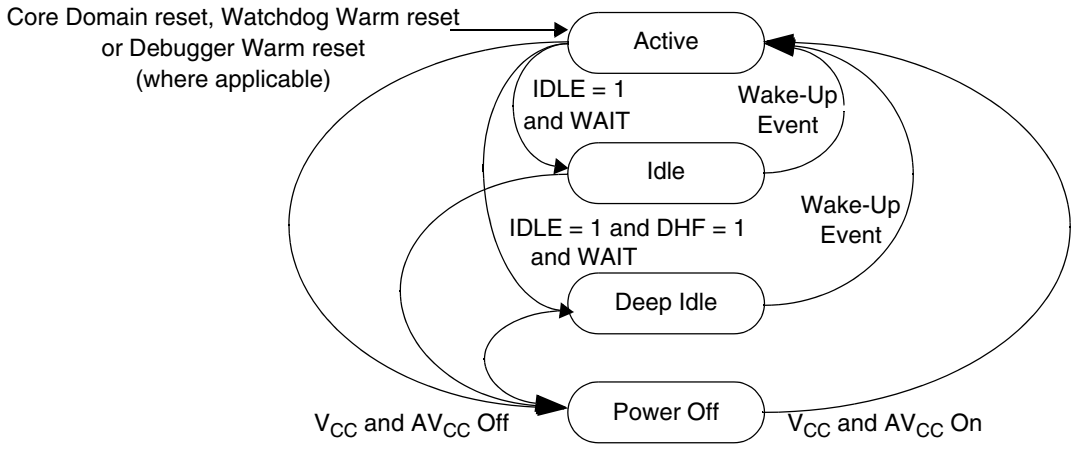


Figure 42. Power Modes and Transitions

Wake-Up Event

The wake-up event has three sources:

- A maskable wake-up event (from MIWU)
- A Non-Maskable Interrupt (NMI)
- A debugger wake-up event (from the debugger interface)

Once a wake-up event is detected, it is latched until an interrupt acknowledge bus cycle is detected or a reset is applied.

$\overline{\text{VCC\_POR}}$ Input Reset

An asserted $\overline{\text{VCC\_POR}}$ Input reset causes a transition to Active mode from any power mode (except Power Off mode).

Decreasing Power Consumption

Entering Idle Mode

To enter Idle mode, perform the following:

1. Set both I and E bits in the core PSR register to 1, to enable maskable interrupts.
2. Set the bits in the ICU IENAM0-1 registers to 1, according to the required wake-up events.
3. Set the bits in the MIWU WAKEN1-5 registers to 1, according to the required wake-up events.
4. Set IDLE bit in PMCSR register to 1.
5. Execute the WAIT instruction to start Idle mode.

Entering Deep Idle Mode

To enter Deep Idle mode, perform the same steps as in “Entering Idle Mode”, adding to step 4, “and set DHF bit in PMCSR register to 1”.

Entering Power Off Mode

Switch to Power Off mode by turning off the supply to the WPC8763L VCC and AVCC pins. Note that V_{DD} must also be turned off.

4.0 Embedded Controller Modules (Continued)

Increasing Activity

Wake-Up from Idle and Deep Idle Modes to Active Mode

A hardware wake-up event causes the core domain to switch directly from Idle or Deep Idle mode to Active mode. The following sequence is performed:

1. DHF in PMCSR register is cleared, thus enabling the high-frequency clock (if it was disabled).
2. After waiting for the high-frequency clock to become active (i.e., OHFC in PMCSR register is set), the core domain switches to Active mode.

When in Active mode, IDLE bit in PMCSR register is cleared.

Since the core was executing a WAIT instruction, it resumes operation by entering an interrupt routine (an enabled maskable interrupt, NMI or ISE).

Exit from Power Off Mode

When in Power Off mode, activity can be resumed only by switching to Active mode. This is done by applying both V_{CC} and AV_{CC} power to the WPC8763L, which performs a V_{CC} Power-Up reset, as described in Section 3.2 on page 48.

Power Mode Switch Protection

The PMC module includes a mechanism that protects the WPC8763L from malfunctions caused by missing or unstable clock signals.

Clock Toggling Indication

OHFC and OLFC bits in PMCSR register indicate the current status of the high- and low-frequency clock inputs, respectively. The current status is based on indications from the HFCG module.

The PMC does not use the high-frequency clock when OHFC bit is 0 and does not use the low-frequency clock when OLFC is 0.

During reset, the core clock domains do not toggle until OHFC is 1. During power mode change, if there is a request to switch to a non-stable or non-toggling clock, the power mode change is stalled until the respective clock is stable.

4.18.4 PMC Registers

For a summary of the abbreviations used for Register Type, see “Register Abbreviations and Access” on page 20.

PMC Register Map

All PMC registers are powered by V_{CC}.

Location	Mnemonic	Register Name	Size	Type
FF F100h	PMCSR	Power Management Controller Status	Byte	Varies per bit

Power Management Controller Status Register (PMCSR)

PMCSR is set to its default value by Core Domain reset, Watchdog Warm reset or Debugger Warm reset.

Location: FF F100h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**OLFC**	**OHFC**	**Must be 1**	**Reserved**		**IDLE**	**DHF**	**Reserved**
Reset	0	0	1	0	0	0	0	0

Bit	Type	Description
7	RO	**OLFC (Oscillating Low-Frequency Clock).** 0: Indicates that the low-frequency clock (LFCLK) received by the PMC is disabled, not available or not stable. When OLFC is 0, the PMC does not switch from Active mode to Idle or Deep Idle mode (default). 1: Indicates that the low-frequency clock received by the PMC is available and stable.

Bit	Type	Description
6	RO	**OHFC (Oscillating High-Frequency Clock).** 0: Indicates that the high-frequency clock (FMCLK) received by the PMC is disabled, not available or not stable. When OHFC is 0, the PMC does not switch to Active mode (default). 1: Indicates that the high-frequency clock received by the PMC is available and stable.
5		**Reserved.** The bit must be 1.
4-3		**Reserved.**
2	R/W	**IDLE.** When set to 1, the core domain enters Idle or Deep Idle mode on the execution of a WAIT instruction. Can be set and cleared by software; it is cleared by hardware when a wake-up event is detected.
1	R/W	**DHF (Disable High-Frequency Oscillator).** When set to 0, the HFCG is enabled. In Active mode, the HFCG is enabled regardless of the DHF value. In Deep Idle mode, DHF is used to reduce power consumption. When set to 1, the HFCG is disabled and the high-frequency clock is not generated. In Power off mode, the HFCG is disabled regardless of the DHF value. Can be set and cleared by software; it is cleared by the hardware when a hardware wake-up event is detected.
0		**Reserved.**

4.18.5 Usage Hints

Entering Idle or Deep Idle

Before entering Idle or Deep Idle mode, the firmware must check if an LPC transaction to the Shared BIOS memory or to the Shared RAM Access windows is not currently being executed. To do this, the firmware must perform the following:

1. Read bit 6 of WKPND5 register; see Section 4.2.3 on page 69.
2. If the bit is set to 1, clear it by writing 1 to bit 6 of WKPCL5 register.
3. Repeat items “1” and “2” as long as bit 6 of WKPND5 register remains set to 1.
4. When the bit is read 0, the firmware can configure the PMC module to enter Idle or Deep Idle mode; see “Decreasing Power Consumption” on page 178.

4.0 Embedded Controller Modules (Continued)

4.19 HIGH-FREQUENCY CLOCK GENERATOR (HFCG)

The HFCG generates the high-frequency clock (FMCLK), which is based on the 32 KHz Clock domain (32.768 KHz, LFCLK clock signal). Based on the FMCLK, the HFCG also generates the flash clock (FCLK) and the core clock (CLK). Clock generation is controlled by the PMC module; see Section 4.18 on page 177.

To generate the FMCLK, the HFCG includes a programmable frequency multiplier. The FCLK and CLK clocks are derived from FMCLK via programmable prescalers.

4.19.1 Features

- Programmable frequency multiplier for a wide range of output frequencies.
- Programmable prescaler to generate the core and flash clocks from FMCLK.
- The default core clock domain frequency is set on V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset.

4.19.2 Functional Description

Figure 43 shows the block diagram of the HFCG module.

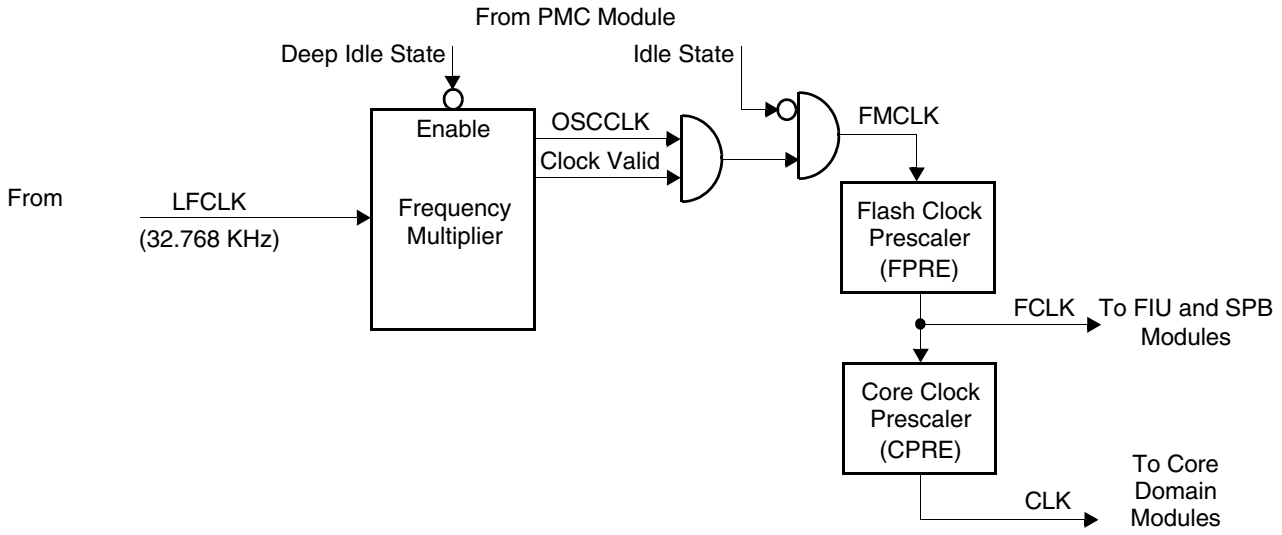


Figure 43. HFCG Module Block Diagram

The frequency multiplier includes a 6-bit programmable (N) value and a 16-bit programmable (M) value; these values define the OSCCLK frequency. During a frequency change, the FMCLK output is held low to prevent the system from using an unstable clock.

The HFCG operation is tightly coupled with the PMC. The HFCG receives Idle state and Deep Idle state signals from the PMC. Idle state disables the FMCLK, FCLK and CLK signals. The Deep Idle state disables the operation of the frequency multiplier.

FMCLK Clock Frequency Setting

To change the FMCLK frequency:

1. Write the N value to HFCGN register.
2. Write the low byte of the M value to HFCGML register.
3. Write the upper byte of the M value to HFCGMH register.
4. Set LOAD in HFCGCTRL1 register to 1.

Table 21 shows the available settings and their corresponding M and N values. After reset, the Booter updates the HFCGML, HFCGMH and HFCGN registers from the flash.

4.0 Embedded Controller Modules (Continued)

Table 21. Frequency Multiplier Setting

FMCLK Frequency	HFCGMH	HFCGML	HFCGN	Notes
50 MHz	0Bh	ECh	02h	
40 MHz	09h	89h	02h	Default
33 MHz	07h	DEh	02h	

Note: When read, the HFCGML, HFCGMH and HFCGN registers return the last M and N values loaded into the frequency multiplier (i.e., loaded either by setting the LOAD bit in HFCGCTRL register to 1 or by reset). Therefore, reading the HFCGML, HFCGMH and HFCGN registers after reset, returns the default M and N values.

Flash Clock (FCLK) Generation

The flash clock (FCLK) is derived from FMCLK via a 4-bit prescaler (FPRE). The FCLK frequency may be set in the range of 10 MHz to 50 MHz. Valid division factor for FCLK ranges from 1 to 4; see Table 22. After reset, the Booter updates FPRED field in HFCGP register from the flash.

Core Clock (CLK) Generation

The core clock (CLK) is derived from FCLK via a 4-bit prescaler (CPRE). The CLK frequency may be set in the range of 4 MHz to 25 MHz. Valid division factors for CLK are shown in Table 22. After reset, the Booter updates CPRED field in HFCGP register from the flash.

Table 22. Flash and Core Clock Frequency Settings

FMCLK Freq.	FPRED Field	FCLK Freq.	CPRED Field[1]	CLK Freq.	Notes
50 MHz	0h	50 MHz	1h	25 MHz	
			2h	16.67 MHz	
			3h	12.5 MHz	
			7h	6.25 MHz	
	1h	25 MHz	0h	25 MHz	
			1h	12.5 MHz	
			2h	8.33 MHz	
			3h	6.25 MHz	
	2h	16.67 MHz	0h	16.67 MHz	
			1h	8.33 MHz	
			2h	5.56 MHz	
			3h	4.17 MHz	
	3h	12.5 MHz	0h	12.5 MHz	Avoid these CLK frequencies if SPB module is used.
			1h	6.25 MHz	
			2h	4.17 MHz	
40 MHz	0h	40 MHz	1h	20 MHz	
			2h	13.33 MHz	
			3h	10 MHz	
			7h	5 MHz	

4.0 Embedded Controller Modules (Continued)

Table 22. Flash and Core Clock Frequency Settings (Continued)

FMCLK Freq.	FPRED Field	FCLK Freq.	CPRED Field[1]	CLK Freq.	Notes
40 MHz	1h	20 MHz	0h	20 MHz	
			1h	10 MHz	
			2h	6.67 MHz	
			3h	5 MHz	
	2h	13.33 MHz	0h	13.33 MHz	Avoid this CLK frequency if SPB module is used.
			1h	6.67 MHz	Avoid this CLK frequency if SPB module is used (default).
			2h	4.44 MHz	Avoid these CLK frequencies if SPB module is used.
	3h	10 MHz	0h	10 MHz	
			1h	5 MHz	
33 MHz	0h	33 MHz	1h	16.5 MHz	
			2h	11 MHz	
			3h	8.25 MHz	
			7h	4.13 MHz	
	1h	16.5 MHz	0h	16.5 MHz	
			1h	8.25 MHz	
			2h	5.5 MHz	
			3h	4.13 MHz	
	2h	11 MHz	0h	11 MHz	Avoid these CLK frequencies if SPB module is used.
			1h	5.5 MHz	
	3h	8.25 MHz	0h	8.25 MHz	
			1h	4.13 MHz	

1. CPRED values that do not appear in the table generate a CLK frequency outside of the 4 MHz to 25 MHz range.

4.19.3 HFCG Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

HFCG Register Map

All HFCG registers are powered by V_{CC} and are set to their default values by V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset. The HFCGCTRL and HFCGP registers are set to their default values by Core Domain reset.

Location	Mnemonic	Register Name	Size	Type
FF F080h	HFCGCTRL	HFCG Control	Byte	Varies per bit
FF F082h	HFCGML	M Low Byte Value	Byte	R/W or RO
FF F084h	HFCGMH	M High Byte Value	Byte	R/W or RO
FF F086h	HFCGN	N Value	Byte	R/W or RO
FF F088h	HFCGP	HFCG Prescaler	Byte	R/W

4.0 Embedded Controller Modules (Continued)

HFCG Control Register (HFCGCTRL)

HFCGTRL is reset by Core Domain reset.

Location: FF F080h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**LOCK**	**Reserved**	**LOAD**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-2		**Reserved.**
2	W1S	**LOCK (Disable Writing to all HCFG registers).** Write 1 to disable write access to all HCFG registers except HFCGP register. Only reset clears this bit. 0: Write access is enabled (default). 1: Write access is disabled.
1		**Reserved.**
0	WO	**LOAD (Load M and N Values).** Write 1 to set the frequency multiplier frequency by loading the HFCGML, HFCGMH and HFCGN data into the frequency multiplier. This bit always returns 0.

HFCGM Low Value Register (HFCGML)

HFCGML contains the lower eight bits of the frequency multiplier M value. It is reset by V_{CC} Power-Up reset and $\overline{\text{VCC\_POR}}$ Input reset.

Location: FF F082h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**HFCGM7-0**							
Reset	See text							

Bit	Description
7-0	**HFCGM7-0.** When written, holds the lower eight bits of the M value **to be** loaded into the frequency multiplier. When read, this field returns the value **currently** in the frequency multiplier. HFCGM7-0 is updated by the Booter from the flash after reset. See Table 21 on page 182 for valid HFCGM7-0 values. The default value is 89h.

HFCGM High Value Register (HFCGMH)

HFCGMH contains the upper eight bits of the frequency multiplier M value. It is reset by V_{CC} Power-Up reset and $\overline{\text{VCC\_POR}}$ Input reset.

Location: FF F084h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**HFCGM15-8**							
Reset	See text							

Bit	Description
7-0	**HFCGM15-8.** When written, holds the upper eight bits of the M value **to be** loaded into the frequency multiplier. When read, this field returns the value **currently** in the frequency multiplier. HFCGM15-8 is updated by the Booter from the flash after reset. See Table 21 on page 182 for valid HFCGM7-0 values. The default value is 09h.

4.0 Embedded Controller Modules (Continued)

HFCGN Value Register (HFCGN)

HFCGN contains six bits of the frequency multiplier N value. It is reset by V_{CC} Power-Up reset and $\overline{VCC\_POR}$ Input reset.

Location: FF F086h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**HFCGN5-0**					
Reset	0	0	0	0	0	0	1	0

Bit	Description
7-6	**Reserved.**
5-0	**HFCGN5-0.** When written, holds the six bits of the N value **to be** loaded into the frequency multiplier. When read, returns the value **currently** in the frequency multiplier. HFCGN5-0 is updated by the Booter from the flash after reset. See Table 21 on page 182 for valid HFCGN5-0 values. The default value is 02h.

HFCG Prescaler Register (HFCGP)

HFCGP is reset by Core Domain reset.

Location: FF F088h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**FPRED**				**CPRED**			
Reset	0	0	1	0	0	0	0	1

Bit	Description
7-4	**FPRED (Flash Prescaler Divider Value).** The divider of FMCLK is the number defined by FPRED + 1, e.g., a value of 00h results in a divide by 1; a value of 3h results in a divide by 4. Valid FPRED values are 0h to 3h; see Table 22 on page 182. FPRED is updated by the Booter from the flash after reset. The default value is 2h.
3-0	**CPRED (Core Prescaler Divider Value).** The divider of the flash clock (FCLK divided by CPRED) is the number defined by CPRED + 1, e.g., a value of 0h results in a divide by 1; a value of 3h results in a divide by 4. Valid division factors for CLK are shown in Table 22 on page 182. CPRED is updated by the Booter from the flash after reset. The default value is 1h.

4.0 Embedded Controller Modules (Continued)

4.19.4 Usage Hints

Changing the FCLK and/or CLK Frequency

To change both FPRED and CPRED fields in HFCGP register, perform two write transactions to HFCGP register. Each transaction must change only one of the fields; the other must remain unchanged. The order in which the FPRED and CPRED fields are changed must ensure that the maximum allowed CLK frequency is not exceeded.

For example, at FMCLK = 50 MHz, to change the FCLK and CLK frequencies from FCLK = 16.67 MHz and CLK = 16.67 MHz, to FCLK = 50 MHz and CLK = 16.67 MHz, perform:

1. Change CPRED from 0h to 2h (FCLK = 16.67 MHz and CLK = 5.56 MHz).
2. Wait at least 16 CLK cycles for the CLK frequency to change.
3. Change FPRED from 2h to 0h (FCLK = 50 MHz and CLK = 16.67 MHz).

Changing the FMCLK Frequency

To increase the FMCLK frequency, first decrease the FCLK and/or CLK frequency (see section above) to ensure that they do not exceed the maximum allowed value during the change.

For example, to change from FMCLK = 33 MHz, FCLK = 33 MHz and CLK = 16.5 MHz, to FMCLK = 50 MHz, FCLK = 25 MHz and CLK = 25 MHz, perform:

1. Change FPRED from 0h to 1h (FMCLK = 33 MHz, FCLK = 16.5 MHz and CLK = 8.25 MHz).
2. Wait at least 16 CLK cycles for the FCLK frequency to change.
3. Change CPRED from 1h to 0h (FMCLK = 33 MHz, FCLK = 16.5 MHz and CLK = 16.5 MHz).
4. Wait at least 16 CLK cycles for the CLK frequency to change.
5. Change M and N from 07DEh and 02h to 0BECh and 02h, respectively, as explained in “FMCLK Clock Frequency Setting” on page 181 (FMCLK = 50 MHz, FCLK = 25 MHz and CLK = 25 MHz).

4.0 Embedded Controller Modules (Continued)

4.20 DEVELOPMENT AND DEBUG SUPPORT

4.20.1 Debugger Interface (SDI+)

The SDI+ is an on-chip debug interface, compliant with the Nexus 5001 Forum TM Standard for Global Embedded Processor Debug Interfaces (Revision 1.0, 15th December 1999), Class 1+. It implements all basic functions for application development and testing. The debugger interface can also be used to program the flash memory.

The debugger interface is enabled in OBD mode only. For the SDI+ module architecture, refer to the *CR16C SDI+ Nexus Class 1+ compliant CR16C Debug Interface Architecture Specification, Revision 1.1, July 2003*.

Note: Depending on $\overline{\text{JEN0}}$ and $\overline{\text{JENK}}$ strap setting, it is **not** possible to use either GPIO42-44, GPIO46, GPIO50 and GPIO52 pins or KBSOUT1-3 and KBSOUT5-6 pins while using the debugger interface because these pins are used by the debugger interface.

Features

- Communication between SDI+ and a debugging tool via a standard IEEE1149.1 (JTAG) port.
- Eight hardware breakpoints/watchpoints, triggered on instruction execution or data access.
- Single-stepping of instructions.
- Access to memory mapped resources, by means of SDI+ -based Direct Memory Access (DMA) functionality.
- Access to core internal registers.
- Supports flash or other memory programing that use the SDI+ -based DMA.
- Wakes up the WPC8763L from Idle or Deep Idle mode.

4.20.2 Debugger Reset Types

The debugger interface can generate two types of reset: Debugger Cold reset and Debugger Warm reset. Both reset types are generated by sending a SYSRST_ON command, followed by a SYSRST_OFF command, via JTAG. DBGRST_MODE bit in RSTCTL register (see page 37) selects the type of reset generated by the SDI+ in response to these JTAG commands.

Debugger Cold Reset

Debugger Cold reset is generated if DBGRST_MODE bit in RSTCTL register is set to 0. This reset type loads default values to most of the core-controlled WPC8763L modules; see Section 3.2.3 on page 49.

A Debugger Cold reset sets DBGRST_STS bit in RSTCTL register to 1.

Debugger Warm Reset

A Debugger Warm reset is generated if DBGRST_MODE bit in RSTCTL register is set to 1. This type of reset loads default values only to the core registers and to the status registers in the Debugger Interface module (JTAG interface). It also wakes up the core system to Active mode.

A Debugger Warm reset sets DBGRST_STS bit in RSTCTL register to 1.

4.20.3 Freezing Events

The WPC8763L can prevent real-time events from interfering with the operation of the debugger by changing the status of the WPC8763L. This is done by disabling maskable interrupts and using SDI+ Automatic Freeze mode. If enabled, Automatic Freeze mode “freezes” the watchdog counter and disables destructive reads.

When the WPC8763L is set to IRE environment, the SDI+ module is held in reset; therefore, Automatic Freeze mode is disabled.

Disabling Maskable Interrupts

Clearing the core I or E bits in PSR register disables the maskable interrupts. The I bit is cleared automatically when a trap, interrupt or reset occurs.

Freezing the Watchdog Counter

To freeze the watchdog counter, use SDI+ Automatic Freeze mode. This mode prevents the watchdog from generating the reset that occurs if the watchdog is not “touched” in time; see Section 4.11 on page 133. The watchdog counter holds its value while it is frozen and resumes counting after “freeze” ends.

If an application fails to touch the watchdog in time, a reset event might be generated before or immediately after the “freeze” onset. As a result, the WPC8763L is reset.

Automatic Freeze mode must be disabled before returning to the application.

4.0 Embedded Controller Modules (Continued)

Disabling Additional Modules

The Keyboard and Mouse Controller, Power Management Channels 1 and 2, all SMB modules, all MFT16 modules and the CR_UART module can be frozen by SDI+. Freezing is enabled when the respective bits in DBGFRZEN1 register is set; the bits can be set to meet the specific needs of different applications.

Disabling Destructive Reads

When “freeze” is enabled, destructive reads do not change the system state (i.e., they return the read data but do not clear bit, set bits or send signals). This allows the debugger to check the values of these bits. NMISTAT register is an exception to this rule and is not affected by “freeze”. Core access to host domain registers are destructive but are not affected by “freeze”. Note that host operations continue without being affected by “freeze”.

4.20.4 Debugger Interface, Core Registers

For a summary of the abbreviations used for Register Type, see “Register Abbreviations and Access” on page 20.

All debugger interface registers are powered by V_{CC} and are set to their default values by Core Domain reset.

Register Map

Location	Mnemonic	Register Name	Size	Type
FF F030h	DBG_DID	Debugger Interface JTAG Device ID	Double-Word	RO
FF F034h	DBGFRZEN1	Debug Freeze Enable 1	Byte	R/W

Debugger Interface JTAG Device ID (DBG_DID)

The value of DBG_DID is scanned by the JTAG interface during the IDCODE instruction.

Location: FF F030h

Type: RO

Bit	31 – 28	27 – 12	11 – 0
Name	**VER**	**Part Number**	**Manufacturer ID**
Reset	0h	FE22h	01Fh

Bit	Description
31-28	**VER (JTAG Version).**
27-12	**JTAG Part Number.** The WPC8763L Part Number field value is FE22h.
11-0	**JTAG Manufacturer ID.** The Winbond JTAG Manufacturer ID is 01Fh.

Debug Freeze Enable 1 Register (DBGFRZEN1)

If Automatic Freeze mode is activated in the SDI+, and the relevant bit in DBGFRZEN1 is set, freezing of the respective module is enabled during debug. DBGFRZEN1 is set to FFh on reset.

Location: FF F034h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**HIFEN**	**Reserved**	**UARTFEN**	**SMB2FEN**	**SMB1FEN**	**MFT2FEN**	**MFT1FEN**
Reset	1	1	1	1	1	1	1	1

Bit	Description
7	**Reserved.**
6	**HIFEN (Host Interface Freeze Enable).** Controls the freezing of the Host Interface modules (Keyboard and Mouse Controller interface and Power Management Channels 1 and 2). When HIFEN is set, in Automatic Freeze mode, the IBF bit in HIKMST register is not cleared by a core read from the input buffer (HIKMDI). 0: Automatic Freeze mode has no effect on the Host Interface modules. 1: Freezes the Host Interface modules during Automatic Freeze mode (default).

Bit	Description
5	**Reserved.**
4	**UARTFEN (CR_UART Freeze Enable).** Controls the freezing of the CR_UART module. When UARTFEN is set, in Automatic Freeze mode, CR_UART data transmission and reception is frozen, the PE, FE, DOE and BKD bits are not cleared by a read from USTAT register and RBF bit is not cleared by a read from URBUF register. 0: Automatic Freeze mode has no effect on the CR_UART module. 1: Freezes the CR_UART module during Automatic Freeze mode (default).
3	**SMB2FEN (SMB2 Freeze Enable).** Controls the freezing of the SMB2 module. When SMB2FEN is set, in Automatic Freeze mode, the SDAST bit in SMB2ST register is not cleared by a core access to the SMB2SDA register. 0: Automatic Freeze mode has no effect on the SMB2 module. 1: Freezes the SMB2 module during Automatic Freeze mode (default).
2	**SMB1FEN (SMB1 Freeze Enable).** Same as SMB2FEN bit, but for SMB1 module.
1	**MFT2FEN (MFT16-2 Freeze Enable).** Controls the freezing of the MFT16-2 module. When set, during Automatic Freeze mode, all timer counter clocks are stopped, and the current values of timer/counter registers T2CNT1 and T2CNT2 are frozen. When Automatic Freeze mode becomes inactive, counting resumes from the previous value. 0: Automatic Freeze mode has no effect on the MFT16-2. 1: Freezes the MFT16-2 module during Automatic Freeze mode (default).
0	**MFT1FEN (MFT16-1 Freeze Enable).** Same as MFT2FEN bit, but for MFT16-1 module.

4.20.5 Usage Hints

Flash Programing. To program the flash memory, use the debugger interface-based DMA mechanism, as follows:

a. Set DBGRST_MODE bit in RSTCTL register to 0 (Debugger Cold reset).
b. Send a SYSRST_ON command to put the WPC8763L in reset (except the debugger interface itself). This clears the flash lock bits in the FIU.
c. Send a SYSRST_OFF command (release the modules from reset), leaving the core in Debug mode.
d. Open Core Bus write access to the flash array by adjusting the Flash Access Window 3 High and Low limits to the actual flash size; see Section 4.1 on page 55.
e. Use the FIU for programing the flash using standard write accesses to the flash via the Core Bus; see the specific flash datasheet.
f. If the attached flash supports burst write access, 32-bit data may be used by the DMA to speed up the programing process.
g. Use the FIU “UMA” feature to perform “housekeeping” flash operations (erase, write control, read status, etc.).

Exit Debug. To exit from Debug mode, perform the following:

a. Send a SYSRST_ON command to put the WPC8763L in reset (except the Debugger Interface itself).
b. Send a SYSRST_OFF command (release the modules from reset) without Debug mode.

JTAG Synchronization. If the core clock (CLK) frequency is lower than the JTAG clock (TCK) frequency, synchronization may be lost. To maintain synchronization, use the Synchronization Busy (SB) bit, which is shifted out from the JTAG Instruction Shift Register (JINR) via the TDO pin. When the SB status flag returns to 0, this indicates that the read or write operation, performed via the JTAG interface, is complete, i.e.:

- The written data was updated in the WPC8763L registers.
- The status flags affected by the read were updated.

4.0 Embedded Controller Modules (Continued)

4.21 CORE BUS WATCHER

4.21.1 General Description

The Core Bus Watcher ("Watcher"), which is connected to both the Core bus and the Peripheral bus, monitors the Core bus and logs bus errors and illegal accesses by the core. The Watcher status can be checked either by polling certain registers or by interrupt.

The Watcher monitors for the occurrence of an Illegal Address transfer or a Core bus ERROR response. On detection of either event, the Watcher logs the relevant address and control signal values. In addition, an interrupt output is asserted (if enabled).

Features

- Core bus ERROR response and Illegal Address (IAD) transfer detection
- Maskable interrupt generation

4.21.2 Functional Description

Illegal Address (IAD) Transfer

When an IAD transfer on the Core bus occurs, and the result is either an O.K. or an ERROR response, the relevant address and control signals are logged, and the CIA flag in WSTAT register is set.

When enabled (via IEN bit in WCNTRL register), the interrupt output is also asserted, and all subsequent IAD transfers and ERROR responses are ignored until CIA bit in WSTAT register is cleared (via the ICL bit in WICLR register).

ERROR Response

When an ERROR response on the Core bus occurs, the relevant address and control signals are logged and the ERR flag is set.

When enabled (by IEN bit in WCNTRL register), the interrupt output is also asserted, and all IAD transfers and ERROR responses are ignored until ERR bit in WSTAT register is cleared (via ICL bit in WICLR register).

Note: An ERROR response is still performed if it results from an IAD transfer; therefore, both CIA and ERR flags are set from a single transfer.

4.21.3 Core Bus Watcher Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

All Watcher registers are powered by V_{CC} and are set to their default values by Core Domain reset.

Register Map

Location	Mnemonic	Register Name	Size	Type
FF F700h	WADR	Watcher Address	Double-Word	RO
FF F704h	WSTAT	Watcher Status	Double-Word	RO
FF F708h	WICLR	Watcher Interrupt Clear	Double-Word	WO
FF F70Ch	WCNTRL	Watcher Control	Double-Word	Varies per bit

4.0 Embedded Controller Modules (Continued)

Watcher Address Register (WADR)

WADR is used to record the Core bus address when an IAD or an ERROR response on the Core bus occurs.

Location: FF F700h

Type: RO

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**CADR**																															
Reset	0000 0000h																															

Bit	Description
31-0	**CADR (Core Bus Address).** Holds the address value for either a Core bus slave ERROR response or an illegal address transaction.

Watcher Status Register (WSTAT)

WSTAT is used to record the Core bus control signals and the type of error detected.

Location: FF F704h

Type: RO

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Reserved**												**CWR**	**CSI**			**Reserved**														**CIA**	**ERR**
Reset	0000 0000h																															

Bit	Description
31-20	**Reserved.**
19	**CWR (Core Bus Write).** Is set to 1 on a Core bus write transaction, if an IAD transfer or ERROR response was detected.
18-16	**CSI (Core Bus Size).** Records the size of the Core bus transaction if an IAD transfer or ERROR response was detected.
15-2	**Reserved.**
1	**CIA (Core Bus IAD Error).** Is set if an illegal address transfer is detected. Can only be cleared by setting ICL bit in WICLR register. 0: Illegal address transfer was not detected (default). 1: Illegal address transfer was detected.
0	**ERR (ERROR Response).** Is set if an ERROR response is detected on the Core bus. Can only be cleared by setting ICL bit in WICLR register. 0: Error response was not detected (default). 1: Error response was detected.

Watcher Interrupt Clear Register (WICLR)

WICLR is used to clear the interrupt flags.

Location: FF F708h

Type: WO

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Reserved**																															**ICL**
Reset	0000 0000h																															

Bit	Description
31-1	**Reserved.**
0	**ICL (Interrupt Clear).** When set, the Interrupt is deasserted and the CIA and ERR flags are cleared. Always returns 0 when read.

4.0 Embedded Controller Modules (Continued)

Watcher Control Register (WCNTRL)

WCNTRL is used to enable the Watcher module and set the control signals.

Location: FF F70Ch

Type: Varies per bit

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Reserved**																												**LOC**	**MEN**	**EDE**	**IEN**
Reset	0000 0000h																															

Bit	Type	Description
31-4		**Reserved.**
3	R/W1S	**LOC (Lock Bit).** When set, the WCNTRL register is locked. Write access to WCNTRL is no longer supported; only read access is enabled. Cleared only by reset. 0: WCNTRL is not locked (default). 1: WCNTRL is locked.
2	R/W or RO	**MEN (Module Enable Bit).** When set, the Watcher module is enabled. If cleared, the clock is switched off and only write access to the WCNTRL register is supported. 0: Watcher is off (default). 1: Watcher is on.
1	R/W or RO	**EDE (Error Detection Enable).** Enables the IAD transfer and ERROR response detection. If cleared, no detection is performed. 0: IAD transfer and ERROR response detection is disabled (default). 1: IAD transfer and ERROR response detection is enabled.
0	R/W or RO	**IEN (Interrupt Enable).** When set enables the IAD transfer and ERROR response interrupt. If cleared, no interrupt is generated. 0: IAD transfer and ERROR response interrupt is disabled (default). 1: IAD transfer and ERROR response interrupt is enabled.

5.0 Host-to-EC Interface Modules

This chapter describes the functions that serve as an interface between the host and core domains:

- Keyboard and Mouse Controller interface (legacy 60h, 64h); see Section 5.1
- Two Power Management (PM) channels compliant with ACPI EC specifications; see Section 5.2 on page 201
- Shared Memory mechanism; see Section 5.4 on page 216
- Core Access to Host Modules; see Section 5.3 on page 211
- Mobile System Wake-Up functions; see Section 5.5 on page 232

5.1 KEYBOARD AND MOUSE CONTROLLER INTERFACE

The WPC8763L supports a standard Keyboard and Mouse Controller interface. This interface implements legacy ports 60h and 64h.

5.1.1 Features

- Intel 8051SL-compatible host interface
  - 8042 KBD standard interface (ports 60h, 64h)
  - Legacy IRQ: IRQ1 (KBD) and IRQ12 (mouse) support
  - Fast Gate A20 and Fast Reset via firmware; see Section 5.5.4 on page 235
- Configured using two logical devices: Keyboard and Mouse

5.1.2 General Description

The WPC8763L supports a keyboard/mouse communication channel that uses the standard Command/Status register and data registers. It uses either polling- or interrupt-driven communication mechanisms with the host and/or core. The hardware is designed to allow a race-free interface between the host and the WPC8763L.

The keyboard and mouse channel consists of three registers:

- DBBOUT - can be written by the core and read by the host processor.
- DBBIN - can be written by the host processor and read by the core.
- STATUS - can be read by both core and host processors. It has five bits (2, 4-7) that are written by the core. Three other bits are controlled by the hardware to indicate the status of DBBIN and DBBOUT registers.

Host Addresses

The host processor accesses the WPC8763L Keyboard/Mouse Host interface registers at two addresses in the host address space. These addresses are defined by two internal chip select signals specified in the WPC8763L host configuration registers; see Section 6.1.11 on page 258). Legacy settings of these addresses are 60h and 64h for the status/command and data registers, respectively.

Table 23 describes the register mapping to the host processor I/O space. For simplicity, the Host Interface module specification refers to the legacy addresses.

Table 23. Mapping of the Host Interface Registers to the Host Processor

Port	Legacy Address	Internal Chip Select	Type	Register Name	Mnemonic
Keyboard and Mouse	60h	Keyboard/Mouse Data	Write	Data	DBBIN (A2=0)
	64h	Keyboard/Mouse Command	Write	Command	DBBIN (A2=1)
	60h	Keyboard/Mouse Data	Read	Data	DBBOUT
	64h	Keyboard/Mouse Command	Read	Status	STATUS

Core Interrupts

The Host Interface module generates four level interrupts to the ICU. These can be used by the firmware for interrupt-driven control of the keyboard/mouse and/or PM channels.

Host Interrupts

The WPC8763L host interface supports two interrupts to the host processor: Keyboard interrupt (legacy IRQ1) and Mouse interrupt (legacy IRQ12). These interrupts may be controlled either by hardware, according to the host interface buffer status, or by the WPC8763L firmware toggling the bit value.

5.0 Host-to-EC Interface Modules (Continued)

The host configuration module assigns host interrupts to IRQ numbers; see Section 6.1.11 on page 258. These interrupts are IRQ1 and IRQ12 for keyboard and mouse IRQs, respectively.

When IRQ1 and/or IRQ12 are disabled (OBFKIE and/or OBFMIE bits in HICTRL register are cleared), the firmware can control the IRQ1 and/or IRQ12 signals by writing to the respective bit in HIIRQC register.

When IRQ1 and/or IRQ12 are controlled by hardware (OBFKIE and/or OBFMIE bits in HICTRL register are set to 1), interrupts to the host are generated according to the status of Output Buffer Full (OBF) flag.

In **Normal** Polarity mode (IRQNPOL in HIIRQC register is set to 0), the interrupt signal is usually low (0); it is asserted (1) to indicate that the respective OBF flag is set. The signal is deasserted (0) when the output buffer is read (i.e., OBF flag is cleared).

Note that IRQ1 and IRQ12 have the same OBF flag but are not asserted together. Either IRQ1 or IRQ12 is set depending on the internal register written (HIKDO or HIMDO, respectively).

In **Negative** Polarity mode (IRQNPOL in HIIRQC register is set to1), the IRQ signal behavior is inverted from the behavior described above.

The WPC8763L firmware can read the values of the IRQ1 and IRQ12 signals by performing a read from IRQ1B and IRQ12B bits in HIIRQC register.

Figure 44 shows the effect of the different control bits on the IRQ signals.

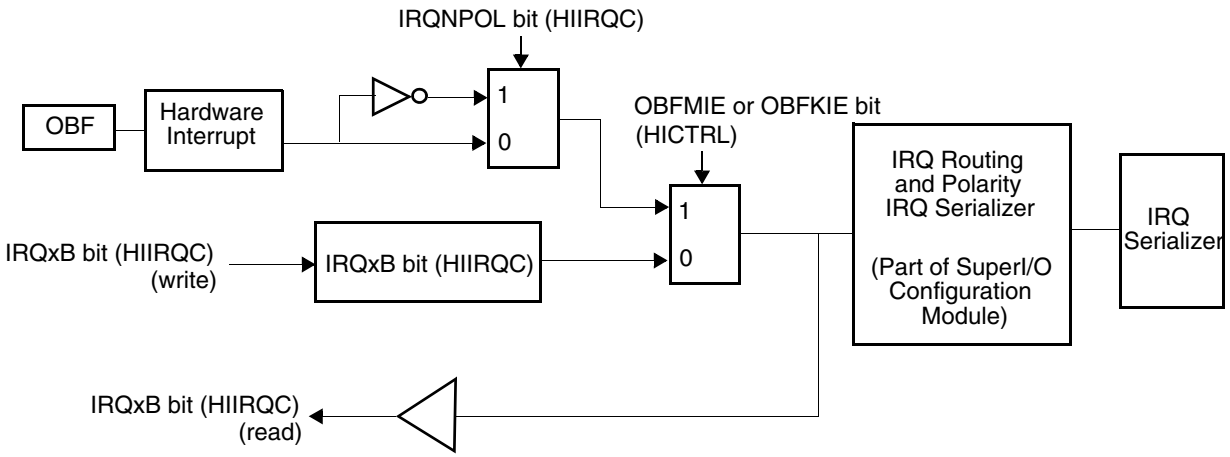


Figure 44. IRQx (IRQ1, IRQ11 or IRQ12) Control Diagram

Keyboard/Mouse Channel (60h, 64h)

The host interface of the WPC8763L is compatible with the legacy 8042 host interface. It is based on two registers: Command/Data and Status. The host interface logic generates interrupts to the host processor and core according to the status of the input and output data buffers. Figure 45 shows a schematic description of the Host interface Keyboard/Mouse channel.

Status Read

Both the host and the core can read the status of the KBC data buffers. Bits 2 and 4-7 can be written by the core. The host processor should read address 64h to obtain the contents of the Status register. The core software can obtain this information by reading/writing the HIKMST register. The format of the Status register is identical for both the host and core; see "Host Interface Keyboard/Mouse Status Register (HIKMST)" on page 199.

Host Data Write to Host Interface Keyboard/Mouse Channel

The data buffer has two latches; one serves as an input buffer, the other as an output buffer. When writing to address 60h or 64h, the following sequence of events occurs: the data is written to the Data In latch (DBBIN), IBF bit in the Status register is set and bit 3 (A2) in the Status register indicates to the core which address (command or data) was written to. When writing to address 60h (legacy A2=0), bit 3 of the Status register is cleared. When writing to address 64h (legacy A2=1), bit 3 of the Status register is set.

The core identifies that data is present in the input buffer by either polling IBF bit of the Status register or acknowledging an interrupt when the input buffer interrupt is enabled (IBFCIE in HICTRL register is set to 1).

When the input buffer is full, reading the Status register identifies which address was written to (i.e., check A2 of HIKMST register). The core can then read the data from the input buffer (HIKMDI). The IBF status bit is cleared when the data input buffer is read by the core.

5.0 Host-to-EC Interface Modules (Continued)

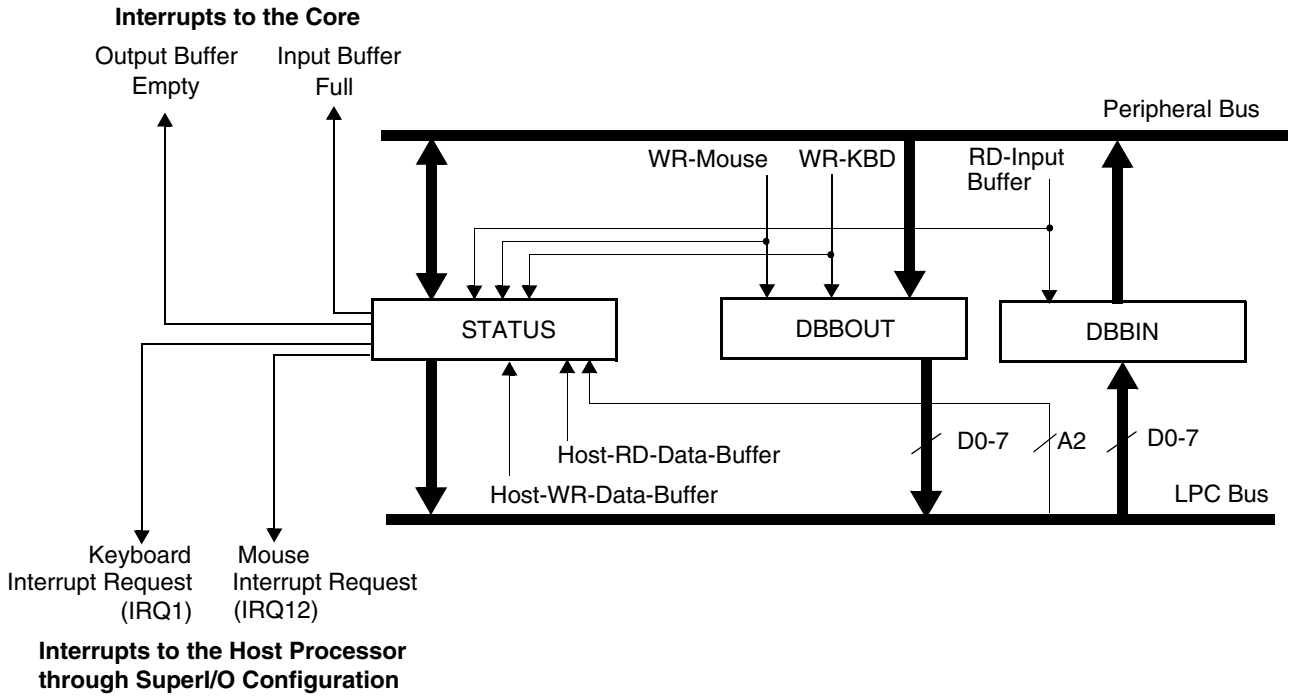


Figure 45. Host Interface Keyboard/Mouse Channel (Ports 60,64) Block Diagram

Host Data Read from Host Interface Keyboard/Mouse Channel

The output data latch (DBBOUT) is written by the core when the core needs to send data to the host. The OBF flag in the Status register (OBF in HIKMST register) is set to indicate that data is available in DBBOUT. DBBOUT should be written only when this bit is cleared.

The WPC8763L supports polling and interrupt communication mechanisms with the host. Both Keyboard interrupt (IRQ1) and Mouse interrupt (IRQ12) are supported.

The core firmware writes to HIKDO register data addressed to the keyboard driver (i.e., generate IRQ1). A write to HIKDO stores the data to DBBOUT and sets OBF bit. In addition, if the IRQ1 interrupt is enabled (OBFKIE in HICTRL register is set to 1), an IRQ1 interrupt is generated according to the interrupt mode (IRQM field and IRQNPOL bit in HIIRQC register).

The core firmware writes data addressed to the mouse driver (IRQ12) to HIMDO register. A write to HIMDO stores the data in DBBOUT and sets OBF bit. In addition, if the IRQ12 interrupt is enabled (OBFMIE in HICTRL is set to 1), an IRQ12 interrupt is generated according to the interrupt mode (IRQM field and IRQNPOL bit in HIIRQC register).

The host processor identifies that data is present in the output buffer by either polling the Status register (reading address 64h) or responding to IRQ1 or IRQ12. When this data is available, the host can read it from address 60h. Reading from address 60h clears the OBF flag. In addition, when the hardware interrupt is enabled, IRQ1 or IRQ12 are deasserted (low if IRQNPOL in HIIRQC register is set to 0).

The core can read OBF bit to identify when the output buffer is empty and ready for a new data transfer. When the Output Buffer Empty interrupt to the core is enabled (OBECIE in HICTRL register is set to 1), the interrupt signal to the ICU is set high if the output buffer is empty (OBF=0).

5.0 Host-to-EC Interface Modules (Continued)

5.1.3 Host Interface Registers

The module has four registers, described below. The base address for each may be configured individually. For legacy operation, they should be configured to 60h and 64h; see Section 6.1.11 on page 258.

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20. All the registers are set to their default values by Core Domain reset. If the HIPRST_MODE bit in RSTCTL register is set to 1, they are also set to their default values by V_{DD} Power-Up reset or Host Hardware reset; see page 37.

Host Interface Register Map

Offset	Mnemonic	Register Name	Type
60h	DBBOUT	Data Out Buffer	R
64h	STATUS	Status	R
60h	DBBIN	Data In Buffer	W
64h	COMAND	Command In Buffer	W

Data Out Buffer Register (DBBOUT, Legacy 60h)

Reading DBBOUT clears the OBF flag in the STATUS register and the interrupt is deasserted. If the core interrupt on output buffer empty is enabled (OBECIE in HICTRL register is set to 1), reading DBBOUT asserts the interrupt (high).

Location: As defined in LDN 06h registers, index 60h and 61h

Type: R

Bit	7	6	5	4	3	2	1	0
Name	**Keyboard/Mouse DBBOUT Data**							

Bit	Description
7-0	**Keyboard/Mouse DBBOUT Data.**

Status Register (STATUS, Legacy 64h)

STATUS provides the status of the host interface keyboard channel buffers (DBBIN and DBBOUT) and the value of messages sent by the core. This register can also be read by the core as HIKMST. It is cleared (00h) on reset.

Location: As defined in LDN 06h registers, index 62h and 63h

Type: R

Bit	7	6	5	4	3	2	1	0
Name	**ST3-ST0**				**A2**	**F0**	**IBF**	**OBF**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-4	**ST3-ST0 (Status).** Four general-purpose flags that can be set or cleared by the core firmware.
3	**A2 (A2 Address).** Holds the value of the A2 signal in the last host write to the keyboard/mouse channel's input buffer (i.e., A2=0 for Data In Buffer write and A2=1 for Command In Buffer write).
2	**F0 (Flag 0).** A general-purpose flag that can be set or cleared by the core firmware.
1	**IBF (Input Buffer Full).** Is set when the keyboard/mouse channel's DBBIN is written by the host processor (i.e., writing to either address 60h, data or address 64h, control). Cleared when the core reads the input buffer (HIKMDI).
0	**OBF (Output Buffer Full).** Is set when the keyboard/mouse channel's DBBOUT is written by the core (i.e., writing to HIKDO or HIMDO registers). Cleared by a host processor read from the keyboard/mouse channel output buffer (60h).

5.0 Host-to-EC Interface Modules (Continued)

Data In Buffer Register (DBBIN, Legacy 60h)

If the core interrupt on IBF is enabled (IBFCIE in HICTRL register is set to 1), writing to DBBIN asserts it (high).

Location: As defined in LDN 06h registers, index 60h and 61h

Type: W

Bit	7	6	5	4	3	2	1	0
Name	**Keyboard/Mouse DBBIN Data**							

Bit	Description
7-0	**Keyboard/Mouse DBBIN Data.**

Command In Buffer Register (COMAND, Legacy 64h)

If the core interrupt on IBF is enabled (IBFCIE in HICTRL register is set to 1), writing to COMAND asserts it (high).

Location: As defined in LDN 06h registers, index 62h and 63h

Type: W

Bit	7	6	5	4	3	2	1	0
Name	**Keyboard/Mouse COMAND Data**							

Bit	Description
7-0	**Keyboard/Mouse COMAND Data.**

5.1.4 Core Interface Registers

Certain bits in PM channel 1 are used to achieve firmware compatibility with the PC87570.

The core interface registers are set to their default values by Core Domain reset. If the HIPRST_MODE bit in RSTCTL register is set to 1, they are also set to their default values by V_{DD} Power-Up reset or Host Hardware reset; see page 37. For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Core Interface Register Map

Location	Mnemonic	Register Name	Size	Type
FF F780h	HICTRL	Host Interface Control	Byte	R/W
FF F782h	HIIRQC	Host Interface IRQ Control	Byte	R/W
FF F784h	HIKMST	Host Interface Keyboard/Mouse Status	Byte	Varies per bit
FF F786h	HIKDO	Host Interface Keyboard Data Out Buffer	Byte	WO
FF F788h	HIMDO	Host Interface Mouse Data Out Buffer	Byte	WO
FF F78Ah	HIKMDI	Host Interface Keyboard/Mouse Data In Buffer	Byte	RO

Host Interface Control Register (HICTRL)

HICTRL is used to set host interface mechanism options. The non-reserved bits are cleared on reset.

Location: FF F780h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**PMICIE**	**PMIOCIE**	**PMIHIE**	**IBFCIE**	**OBECIE**	**OBFMIE**	**OBFKIE**
Reset	0	0	0	0	0	0	0	0

5.0 Host-to-EC Interface Modules (Continued)

Bit	Description
7	**Reserved.**
6	**PMICIE (PM Channel 1 Input Buffer Full Core Interrupt Enable).** 0: Interrupt signal is low (default). 1: Enables the PM input buffer full interrupt to the ICU for PM channel 1 in PC87570 Compatible mode. The interrupt signal is active when the input buffer is full (IBF bit is set in the PM Channel Status register).
5	**PMOCIE (PM Channel 1 Output Buffer Empty Core Interrupt Enable).** 0: Interrupt signal is low (default). 1: Enables the PM Output Buffer Empty interrupt to the ICU for PM channel 1 in PC87570 Compatible mode. The interrupt signal is active when the output buffer is empty (OBF bit is cleared in the PM Channel Status register).
4	**PMIHIE (PM Channel 1 Host Interrupt Enable).** 0: IRQ11 is controlled by IRQ11B bit in HIIRQC register (default). 1: Enables the Output Buffer Full interrupt of PM channel 1 in PC87570 Compatible mode to drive the host processor interrupt. The interrupt is noted as IRQ11 and may be routed to any of the IRQs to either $\overline{\text{SMI}}$ or $\overline{\text{ECSCI}}$ events. The interrupt is triggered by a core write to HIPM0DO register and sent according to IRQNPOL bit in HIIRQC register.
3	**IBFCIE (Input Buffer Full Core Interrupt Enable).** 0: Interrupt signal is low (default). 1: Enables the Input Buffer Full interrupt to the ICU for the keyboard/mouse channel. The interrupt signal is active when the input buffer is full (i.e., the interrupt signal is set (1) when IBF bit is set).
2	**OBECIE (Output Buffer Empty Core Interrupt Enable).** 0: Interrupt signal is low (default). 1: Enables the Output Buffer Empty interrupt to the ICU for the keyboard/mouse channel. The interrupt signal is active when the output buffer is empty (i.e., the interrupt signal is set (1) when OBF bit is cleared).
1	**OBFMIE (Output Buffer Full Mouse Interrupt Enable).** 0: IRQ12 is controlled by IRQ12B bit in HIIRQC register (default). 1: Enables the Output Buffer Full interrupt to the mouse driver in the host processor (IRQ12). The interrupt is triggered by the core write to HIMDO register and sent according to IRQNPOL bit in HIIRQC register.
0	**OBFKIE (Output Buffer Full Keyboard Interrupt Enable).** 0: IRQ1 is controlled by IRQ1B bit in HIIRQC register (default). 1: Enables the Output Buffer Full interrupt to the keyboard driver of the host processor (IRQ1). The interrupt is triggered by a core write to HIKDO register and sent according to IRQNPOL bit in HIIRQC register.

Host Interface IRQ Control Register (HIIRQC)

HIIRQC controls the mode of operation of the IRQ signals. It is set to 07h on reset.

Location: FF F782h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**IRQNPOL**	**Reserved (must be '000')**			**IRQ11B**	**IRQ12B**	**IRQ1B**
Reset	0	0	0	0	0	1	1	1

Bit	Description
7	**Reserved.**
6	**IRQNPOL (Negative Polarity).** When cleared, the IRQ (IRQ1, IRQ11, IRQ12) signal polarity is compatible with the standard ISA bus interface. When hardware IRQ generation is enabled (HICTRL register bits OBFKIE for IRQ1 and IRQ12; PMHIE for IRQ11), the interrupt output is inverted if IRQNPOL is set.
5-3	**Reserved.** Must be '000'.

5.0 Host-to-EC Interface Modules (Continued)

Bit	Description
2	**IRQ11B (Host Interrupt Request 11 Control Bit).** When PM channel 1 is in PC87570 Compatible mode and its host interrupt is configured for direct control by the firmware (PMHIE in HICTRL register is 0), IRQ11B bit is output to the IRQ11 signal. When read, IRQ11B bit returns the current value of the IRQ11 signal. The IRQ11 signal value can be read regardless of the state of PMHIE bit; see Section 5.2 on page 201 for details about the PM channel 1 interrupt mechanism.
1	**IRQ12B (Host Interrupt Request 12 Control Bit).** When the IRQ12 signal is configured for direct control by the firmware (OBFMIE in HICTRL register is 0), IRQ12B bit is output to the IRQ12 signal. When read, IRQ12B bit returns the current value of the IRQ12 pin. The IRQ12 signal value can be read regardless of the state of the OBFMIE bit.
0	**IRQ1B (Host Interrupt Request 1 Control Bit).** When the IRQ1 signal is configured for direct control by the firmware (OBFKIE in HICTRL register is 0), IRQ1B bit is output to the IRQ1 signal. When read, IRQ1B bit returns the current value of the IRQ1 pin. The IRQ1 signal value can be read regardless of the state of OBFKIE bit.

Host Interface Keyboard/Mouse Status Register (HIKMST)

HIKMST provides the status of the host interface keyboard channel buffers (DBBIN and DBBOUT) and a way for the WPC8763L to send status bits to the host. It can also be read by a host processor read from address 64h. It is cleared on reset.

Location: FF F784h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**ST3-ST0**				**A2**	**F0**	**IBF**	**OBF**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-4	R/W	**ST3-ST0 (Status Bits).** Four general-purpose flags that can be set or cleared by the core firmware.
3	RO	**A2 (A2 Address).** Holds the value of the A2 signal in the last host write to the keyboard/mouse channel's input buffer (i.e., indicates A2 value during write to address 60h or 64h). This read-only bit is ignored when writing to HIKMST register.
2	R/W	**F0 (Flag 0).** A general-purpose flag that can be set or cleared by the core firmware.
1	RO	**IBF (Input Buffer Full).** Is set when the keyboard/mouse channel's DBBIN is written by the host processor, i.e., writing to either address 60h (data) or address 64h (control). Cleared by a core read of the input buffer (HIKMDI). This read-only bit is ignored when writing to HIKMST register.
0	RO	**OBF (Output Buffer Full).** Is set when the keyboard/mouse channel's DBBOUT is written by the core (i.e., writing to HIKDO or HIMDO register). Cleared by a host processor read from the keyboard/mouse channel output buffer (60h). This read-only bit is ignored when writing to HIKMST register.

Host Interface Keyboard Data Out Buffer Register (HIKDO)

When the core firmware writes to HIKDO, OBF bit in the Status register is set, an IRQ1 interrupt is sent, if enabled, and the core interrupt on output buffer empty is deasserted, if enabled (i.e., if OBECIE in HICTRL register is 1).

Location: FF F786h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**Keyboard DBBOUT Data**							

Bit	Description
7-0	**Keyboard DBBOUT Data.**

5.0 Host-to-EC Interface Modules (Continued)

Host Interface Mouse Data Out Buffer Register (HIMDO)

When the core firmware writes to HIMDO, OBF bit in the Status register is set, an IRQ12 interrupt is sent, if enabled, and the core interrupt on output buffer empty is deasserted, if enabled (i.e., if OBECIE in HICTRL register is 1).

Location: FF F788h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**Mouse DBBOUT Data**							

Bit	Description
7-0	**Mouse DBBOUT Data.**

Host Interface Keyboard/Mouse Data In Buffer Register (HIKMDI)

When the core firmware reads from HIKMDI, IBF bit in the Status register is cleared and the core interrupt on IBF is deasserted, if enabled (i.e., if IBFCIE in HICTRL register is 1).

Location: FF F78Ah

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Keyboard/Mouse DBBIN Data**							

Bit	Description
7-0	**Keyboard/Mouse DBBIN Data.**

5.0 Host-to-EC Interface Modules (Continued)

5.2 POWER MANAGEMENT (PM) CHANNELS

The WPC8763L implements two PM communication channels, both of which are compliant with ACPI specifications for an EC interface. The WPC8763L may be configured to work in either Private Interface or Shared Interface mode. The PM interface has two identical channels, which are independently configured and used. If only one channel is used, Shared Interface mode must be used; if both channels are used, Private Interface mode may be used.

The description below refers to a single channel. The suffix 'n' in the signal names indicate the channel number (either 1 or 2). Registers in the core domain have the prefix HIPMn to indicate the channel number. Host domain registers are identified by the logical device to which they belong.

Note: When working in PC87570 Compatible mode, only PM channel 1 may be used.

5.2.1 Features

- Two operation modes:
  - PC87570 Compatible
  - Enhanced PM
- ACPI EC interface-compliant support for:
  - Shared interface mode
  - Private interface mode
- PM channel registers (channel 1: legacy 62h, 66h; channel 2: legacy 68h, 6Ch):
  - Command/Status
  - Data
- PM interrupt using:
  - IRQ
  - $\overline{\text{SMI}}$
  - $\overline{\text{ECSCI}}$

5.2.2 General Description

The PM channel has two modes of operation:

- PC87570 Compatible (available for channel 1 only) supports software previously written for the PC87570.
- Enhanced PM includes a mechanism that facilitates easier generation of $\overline{\text{ECSCI}}$ and $\overline{\text{SMI}}$ interrupts to the host.

Figure 46 is a schematic diagram of the PM channel.

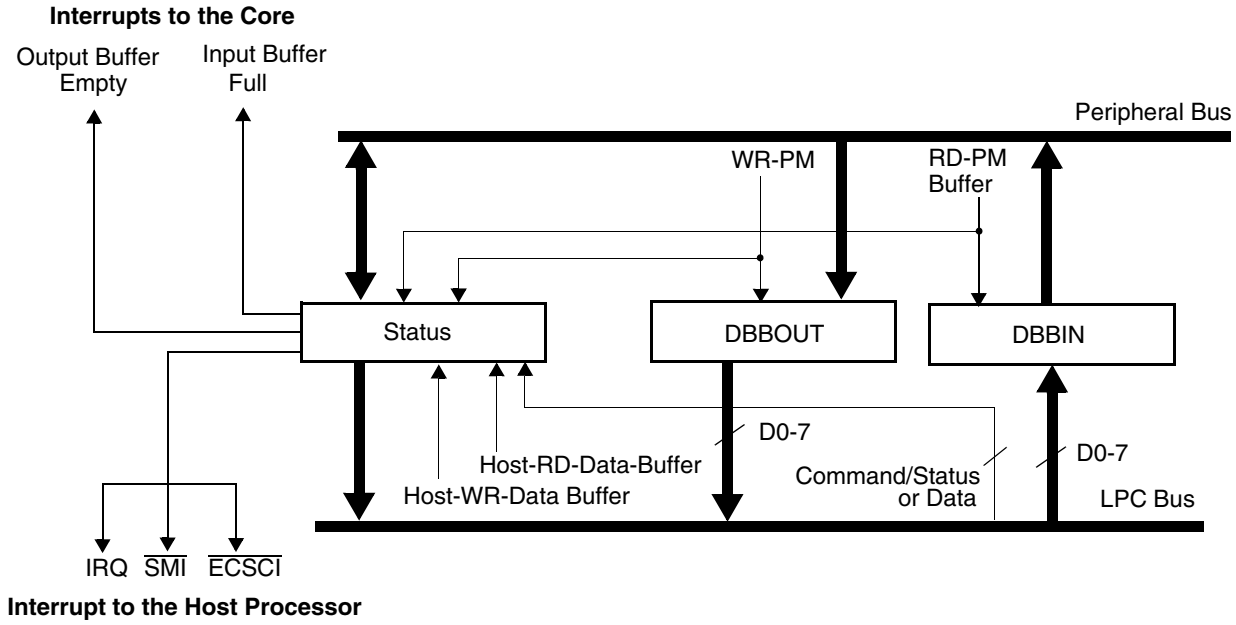


Figure 46. Host Interface PM Channel n Block Diagram

5.0 Host-to-EC Interface Modules (Continued)

Data Registers

The PM channel has three registers.

- DBBOUT - can be written to by the core and read by the host processor. Multiple addresses in the core address space enable generating an IRQ, SMI or ECSCI interrupt on Output Buffer Full (OBF).
- DBBIN - can be written to by the host processor and read by the core.
- STATUS - can be read by both the core and the host processor. It has five bits (bits 2 and 4-7) that are written to directly by the core or, in Enhanced PM mode, via the control and configuration register. Three other bits are controlled by hardware to indicate the status of DBBIN and DBBOUT registers.

Host Addresses

The host processor accesses the WPC8763L PM channel interface registers at two addresses in the host address space. These addresses are defined by two internal chip select signals specified in the WPC8763L host configuration registers; see Section 6.1.14 on page 269 and Section 6.1.15 for channels 1 and 2, respectively. The Legacy setting of these addresses is 62h and 66h for channel 1 status/command and data registers, respectively.

Table 24 shows the register mapping to the host processor I/O space. For simplicity, the Host Interface module specification refers to the legacy addresses.

Table 24. Host Interface Registers to Host Processor Mapping

Port	Legacy Address[1]	Configuration Register Index	PM Internal Chip Select	Type	Register Name	Mnemonic
PM Channel n	62h	Index 60h, 61h	Data	Write	Data	DBBIN
	66h	Index 62h, 63h	Command/Status	Write	Command	DBBIN
	62h	Index 60h, 61h	Data	Read	Data	DBBOUT
	66h	Index 62h, 63h	Command/Status	Read	Status	STATUS

1. The legacy address serves as an example only. Do not assign the same address to both channels.

Core Interrupts

For each channel, the Host Interface module generates two level interrupts to the ICU; see Figure 47. The firmware can use these for interrupt-driven control of the PM channels.

In PC87570 Compatible mode (EME in HIPMnCTL register is set to 0), interrupts are enabled using HICTRL register bits PMOCIE and PMICIE for output buffer empty and input buffer full interrupts, respectively.

In Enhanced PM mode (EME in HIPMnCTL register is set to1), interrupts are enabled using HIPMnCTL register bits OBEIE and IBFIE for output buffer empty and input buffer full interrupts, respectively.

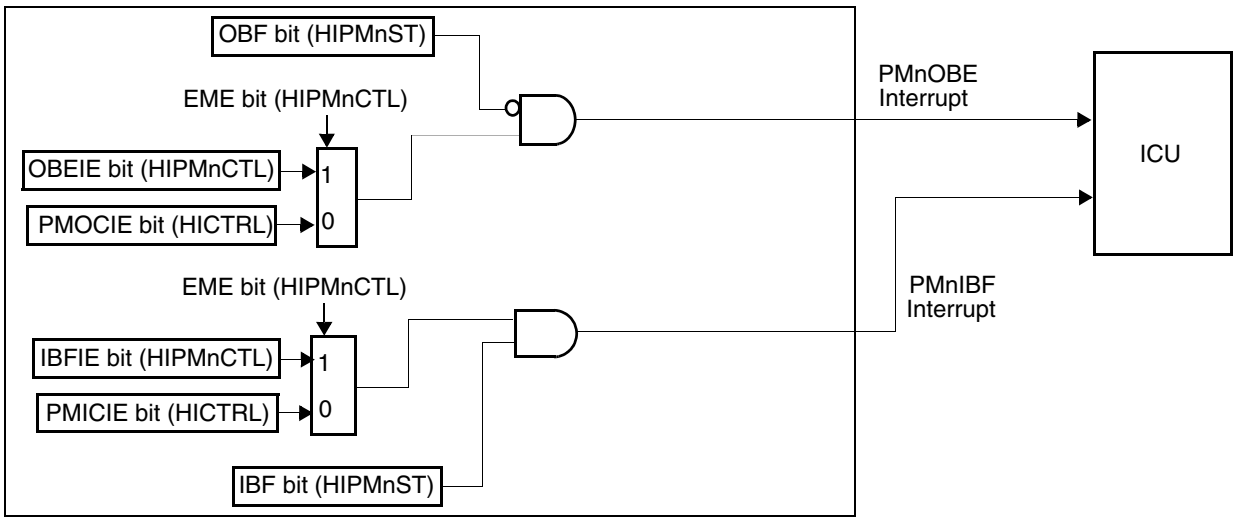


Figure 47. Core Interrupt Request for PM Channel n

5.0 Host-to-EC Interface Modules (Continued)

Host Interrupt Generation Modes

The Host Interface module generates three types of interrupts to the host: regular IRQ, $\overline{\text{SMI}}$ and $\overline{\text{ECSCI}}$. The interrupt mechanisms are designed to meet ACPI requirements for host interrupts. Two interrupt modes are supported: PC87570 Compatible and Enhanced PM.

PC87570 Compatible Mode

PC87570 Compatible mode uses the same method for IRQ generation as the PC87570. It is available only for PM channel 1 and is enabled when EME in HIPM1CTL register is set to 0. Figure 48 shows this mechanism.

The host configuration module assigns host interrupts to IRQ numbers; see Section 6.1.14 on page 269. IRQ11 is used as an example interrupt and for the default bit names (the actual interrupt number used is determined by the IRQ routing logic). When hardware-driven IRQ11 is disabled (PMHIE in HICTRL register is cleared), the firmware can control the IRQ11 signal by writing to the signal's respective bit in HIIRQC register. When hardware-driven IRQ11 is enabled (PMHIE is set to 1), interrupts to the host are generated according to the status of the OBF flag.

In **Normal** Polarity mode (IRQNPOL in HIIRQC register is cleared), the interrupt signal is usually low (0). It is asserted (1) to indicate that the respective OBF flag has been set. The signal is deasserted (0) when the output buffer is read (i.e., OBF flag is cleared).

In **Negative** Polarity mode (IRQNPOL in HIIRQC register is set to 1), IRQ signal behavior is inverted from the behavior described for normal polarity.

The WPC8763L firmware can read the value of the IRQ11 signal by performing a read of IRQ11B bit in HIIRQC register.

The core can also control the routing of interrupts generated by the PM channel to one of the following:

- IRQ signal, when IRQE bit in HIPM1IE register is set.
- $\overline{\text{SMI}}$ output, when SMIE bit in HIPM1IE register is set.
- SCI event, using the $\overline{\text{ECSCI}}$ output, when SCIE bit in HIPM1IE register is set.

The core firmware should not enable more than one of these interrupts simultaneously. It should also update ST0 and ST1 bits to indicate the type of host interrupt used.

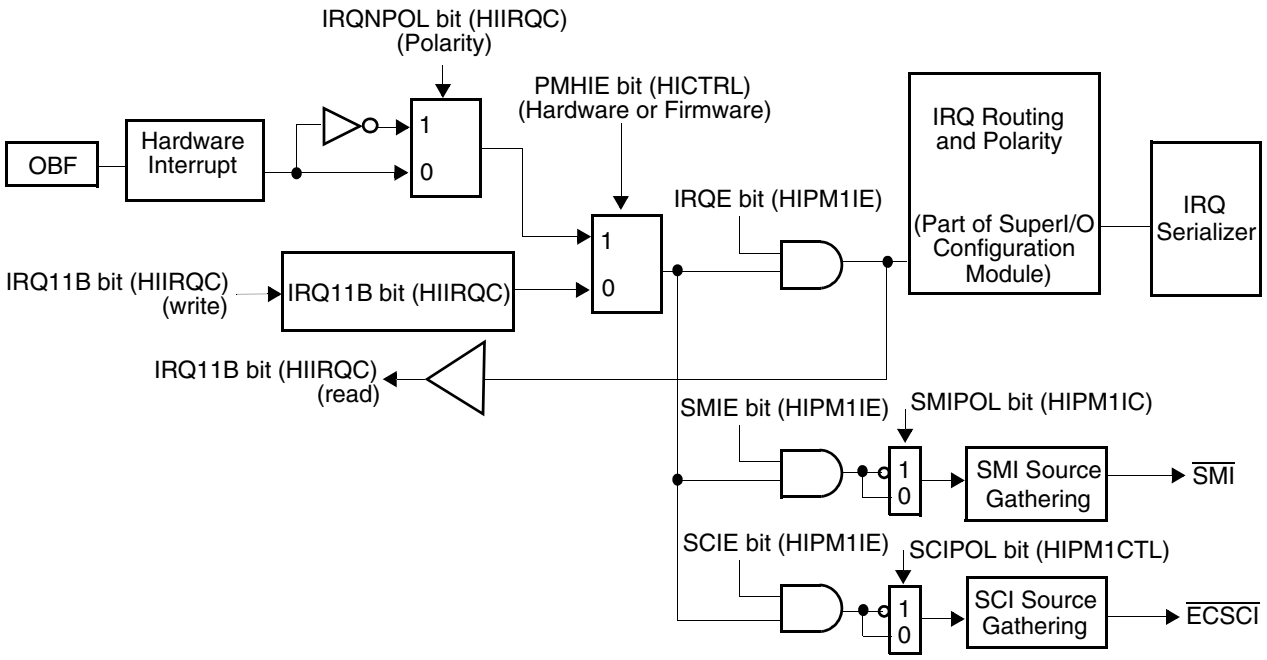


Figure 48. IRQ, $\overline{\text{ECSCI}}$ and $\overline{\text{SMI}}$ Control in PC87570 Compatible Mode (PM Channel 1 Only)

Enhanced PM Mode

Enhanced PM mode is available for both PM channels. It is enabled when EME in HIPMnCTL register is set to 1. Figure 49 shows interrupt generation in this mode.

An IRQ, $\overline{\text{SMI}}$ or $\overline{\text{ECSCI}}$ interrupt may be generated under software control or by using hardware. Using hardware reduces software overhead and simplifies procedures.

5.0 Host-to-EC Interface Modules (Continued)

The mechanism that generates the IRQ is identical to that used in PC87570 Compatible mode. To enable identical control of both channels, the bits that are used for channel 1 are separated from the keyboard/mouse channel's registers; see Figure 49 for bit usage.

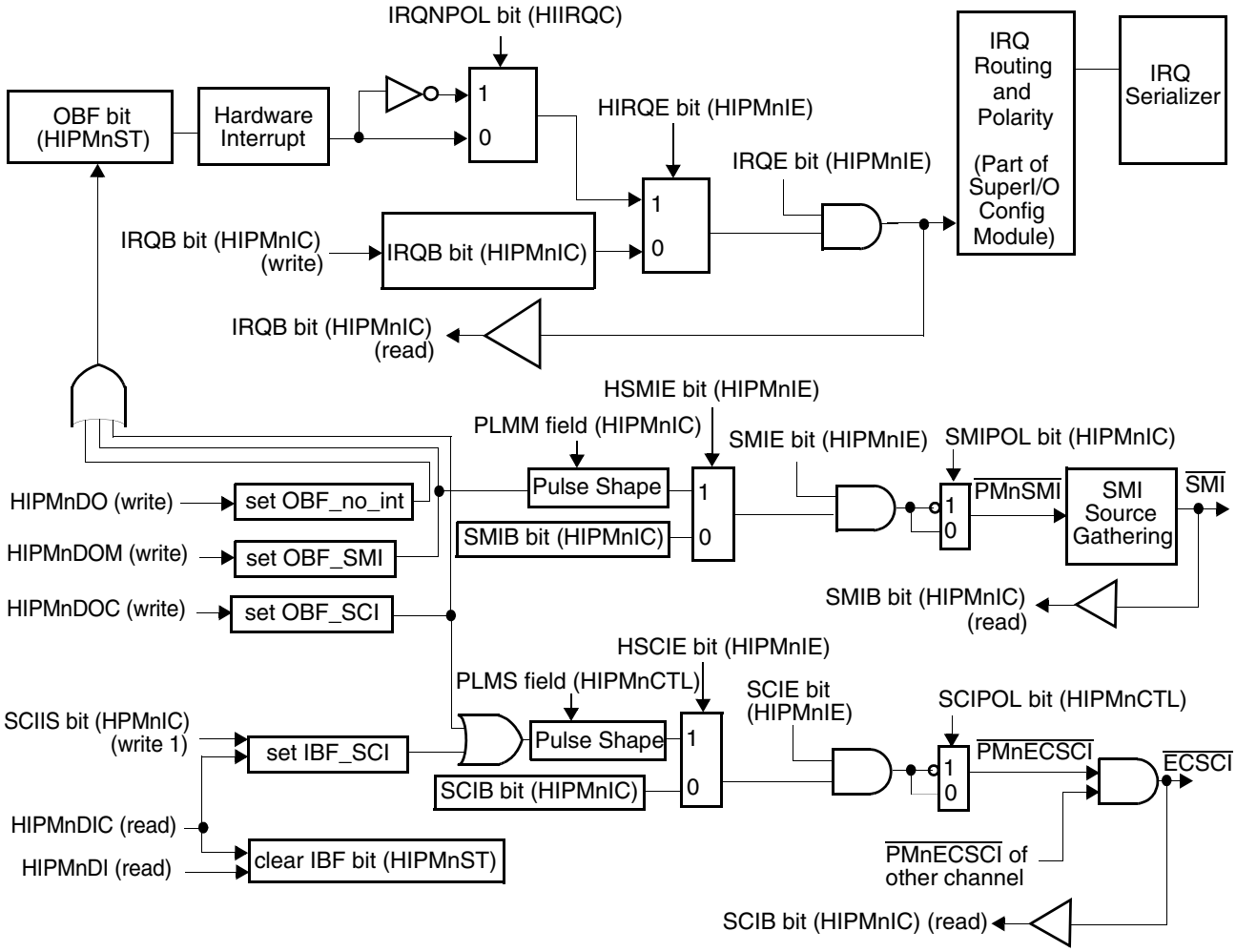


Figure 49. IRQ, $\overline{\text{ECSCI}}$ and $\overline{\text{SMI}}$ Control, Enhanced PM Mode

IRQE in HIPMnIE register determines if an IRQ is sent from PM Channel n.

Enhanced PM mode supports direct generation of $\overline{\text{ECSCI}}$ and $\overline{\text{SMI}}$ on core writes to the Data output buffer; it supports generation of $\overline{\text{ECSCI}}$ on core reading of the Data Input buffer. The core decides whether to generate an interrupt and which type of interrupt to generate by selecting the data register address in use.

When data is written to HIPMnDO register, the OBF flag in HIPMnST register is set and neither $\overline{\text{SMI}}$ nor $\overline{\text{ECSCI}}$ is generated.

When data is written to HIPMnDOM register, the OBF flag and the internal OBF_SMI flag are set and an $\overline{\text{SMI}}$ interrupt is generated. The OBF_SMI flag is cleared when OBF flag is cleared. The $\overline{\text{SMI}}$ is generated as a pulse whose width is defined by PLMM in HPIMnIC register.

The $\overline{\text{SMI}}$ interrupt is routed to the $\overline{\text{SMI}}$ pin only if both HSMIE and SMIE bits in HIPMnIE register are set. When SMIE is set and HSMIE is cleared, SMIB in HIPMnIC register is used as the PMnSMI signal value. When SMIE is cleared, the PMnSMI signal is inactive (high).

When data is written to HIPMnDOC register, the OBF flag and OBF_SCI internal flag are set and an $\overline{\text{ECSCI}}$ interrupt is generated. OBF_SCI is cleared when OBF is cleared. The $\overline{\text{ECSCI}}$ is generated as a pulse whose width is defined by PLMS in HPIMnCTL register.

When data is read from HIPMnDIC register, the IBF flag is cleared, the IBF_SCI Internal flag is set and an $\overline{\text{ECSCI}}$ interrupt is generated. The IBF_SCI flag is cleared when the IBF is set again. The $\overline{\text{ECSCI}}$ is generated as a pulse whose width is defined by PLMS in HPIMnCTL register. Reading from HIPnDI register clears the IBF flag but does not generate an $\overline{\text{ECSCI}}$ interrupt.

Note that IBF_SCI flag may also be set by writing a 1 to SCIIS bit in HIPMnIC register. This is done to start an $\overline{\text{ECSCI}}$ interrupt on input buffer empty without having to read from the input buffer.

The $\overline{\text{ECSCI}}$ interrupt is routed to the $\overline{\text{ECSCI}}$ pin only if HSCIE and SCIE in HIPMnIE register are set. When SCIE is set and HSCIE is cleared, the value of SCIB in HIPMnIC register is used as the PMnSCI signal value. When SCIE is cleared, PMnSCI is inactive (high).

Status Read

The status of the PM channel data buffers can be read by both the host and the core. Bits 2 and 4-7 can be written by the core.

The host processor should read the Status register I/O address (legacy 66h, for channel 1) to obtain the contents of the Status register. The core software should read/write the HIPMnST register to access the same information. The format of the Status register is identical for both the host and the core; see "Host Interface PM n Status Register (HIPMnST)" on page 206.

Host Data Write to Host PM Channel

The data buffer has two latches: one serves as an input buffer; the other serves as an output buffer. When writing to the Command (legacy address 66h) or Data (legacy address 62h) registers, the following sequence occurs: data is written to the Data In latch (DBBIN), IBF bit in the Status register is set and bit 3 (CMD) in the Status register indicates to the core which of the two registers was written to. When writing to the data register address, CMD bit of the Status register is cleared (0). When writing to the command register address, CMD bit in the Status register is set (1).

The core identifies that data is present in the input buffer by either polling IBF bit in the Status register or acknowledging an interrupt when the input buffer interrupt is enabled (IBFCIE bit in HICTRL register is set to 1).

When the input buffer is full, check CMD bit in HIPMnST register to identify which registers were written to. The core can then read the data from the input buffer (HIPMnDI or HIPMnDIC registers). The IBF status bit is cleared when the data input buffer is read by the core.

Host Data Read from Host Interface Power Management Channel

The core writes to the Output Data latch (DBBOUT) when it needs to send data to the host. The OBF flag in the Status register (HIPMnST) is set to indicate that data is available in DBBOUT. DBBOUT should be written to only when OBF in HIPMnST register is cleared.

The WPC8763L supports polling and interrupt communication mechanisms with the host. IRQ, $\overline{\text{SMI}}$ or $\overline{\text{ECSCI}}$ interrupts may be used. The core firmware writes data addressed to the PM drivers to the HIPMnDO register. When working in Enhanced PM mode, writes to HIPMnDOC and HIPMnDOM may be used to automatically generate $\overline{\text{ECSCI}}$ and $\overline{\text{SMI}}$, respectively. Refer to "Host Interrupt Generation Modes" on page 203 for details of the interrupt generation mechanism.

The host processor identifies the presence of data in the output buffer by either polling the Status register or responding to an IRQ, $\overline{\text{SMI}}$ or $\overline{\text{ECSCI}}$ event. When data is available, the host can read it from the data register (legacy address 62h for channel 1). Reading the data register clears the OBF flag (HIPMnST). In addition, the hardware interrupt is enabled and the IRQ signal is deasserted (low).

The core can read OBF in HIPMnST register to identify when the output buffer is empty and ready for a new data transfer. When the output buffer is empty (OBF is set to 0), if the output buffer empty interrupt to the core is enabled (PMOCIE bit in HICTRL register is 1 when EME bit in HIPMnCTL register is 0), the interrupt signal to the ICU is set high.

5.2.3 Host PM Registers

These registers are similar to the KBC Host Interface registers, in Section 5.1.3 on page 196.

5.0 Host-to-EC Interface Modules (Continued)

5.2.4 Core PM Registers

These registers are reset by Core Domain reset. They are also set to their default values by V_{DD} Power-Up reset or Host Hardware reset if the HIPRST_MODE bit in RSTCTL register is set to 1; see page 37. For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Core PM Register Map

Location	Mnemonic	Register Name	Size	Type
FF F78Ch, FF F79Eh	HIPMnST[1]	Host Interface PM n Status	Byte	Varies per bit
FF F78Eh, FF F7A0h	HIPMnDO[1]	Host Interface PM n Data Out Buffer	Byte	WO
FF F790h, FF F7A2h	HIPMnDI[1]	Host Interface PM n Data In Buffer	Byte	RO
FF F792h, FF F7A4h	HIPMnDOC[1]	Host Interface PM n Data Out Buffer with SCI	Byte	WO
FF F794h, FF F7A6h	HIPMnDOM[1]	Host Interface PM n Data Out Buffer with SMI	Byte	WO
FF F796h, FF F7A8h	HIPMnDIC[1]	Host Interface PM n Data In Buffer with SCI	Byte	RO
FF F798h, FF F7AAh	HIPMnCTL[1]	Host Interface PM n Control	Byte	R/W
FF F79Ah, FF F7ACh	HIPMnIC[1]	Host Interface PM n Interrupt Control	Byte	R/W
FF F79Ch, FF F7AEh	HIPMnIE[1]	Host Interface PM n Interrupt Enable	Byte	R/W

1. Where n stands for PM Channel 1 or 2.

Host Interface PM n Status Register (HIPMnST)

HIPMnST contains the status of the host interface PM channel buffers (DBBIN and DBBOUT). It also provides a means for the WPC8763L to send status bits to the host. It is read by a host processor read from address 66h. It is cleared to 00h on reset.

Location: PM1 - FF F78Ch
PM2 - FF F79Eh

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**ST3-ST0**				**CMD**	**F0**	**IBF**	**OBF**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-4	R/W	**ST3-ST0 (Status).** Four general-purpose flags that can be used for signaling between the host and core. When used as an EC interface channel for ACPI, a predefined meaning is assigned to bits ST0, ST1 and ST2. The default meaning is BURST, $\overline{\text{ECSCI}}$ event and $\overline{\text{SMI}}$ event, respectively.
3	RO	**CMD (Command Register).** Indicates whether the last host write to the PMn channel was to the data register or the command register. Writing to this bit is ignored. 0: Last write was to the data register (pointed to by configuration register index 60h and 61h) (default). 1: Last write was to the command register (pointed to by configuration register index 62h and 63h).
2	R/W	**F0 (Flag 0).** General-purpose flag that can be set or cleared by the core firmware.
1	RO	**IBF (Input Buffer Full).** Is set when the PM channel's DBBIN register is written to by the host processor (writing to either address 62h or address 66h). Cleared by a core read of the PM input buffer (HIPMnDI or HIPMnDIC).
0	RO	**OBF (Output Buffer Full).** Is set when the PM channel's DBBOUT register is written to by the core (writing to HIPMnDO, HIPMnDOM or HIPMnDOC register). Cleared by a host processor read of the output buffer (62h). Writing to this bit is ignored.

5.0 Host-to-EC Interface Modules (Continued)

Host Interface PM n Data Out Buffer (HIPMnDO)

HIPMnDO allows the core firmware to write to the PM port DBBOUT register while setting the PM port OBF bit in the Status register. If enabled, an IRQ11 (and/or $\overline{\text{ECSCI}}$ and/or $\overline{\text{SMI}}$, in PC87570 Compatible mode) interrupt is sent at that time. If the core interrupt on PM port output buffer empty is enabled, writing to HIPMnDO deasserts it (low).

Location: PM1 - FF F78Eh
PM2 - FF F7A0h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**PM Channel DBBOUT Data**							

Bit	Description
7-0	**PM Channel DBBOUT Data.**

Host Interface PM n Data In Buffer (HIPMnDI)

HIPMnDI allows the core firmware to read the PM port DBBIN register while clearing the PM port IBF bit in the Status register. If the core interrupt on IBF for the PM channel is enabled, reading from HIPMnDI deasserts it (low).

Location: PM1 - FF F790h
PM2 - FF F7A2h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**PM Channel DBBIN Data**							

Bit	Description
7-0	**PM Channel DBBIN Data.**

Host Interface PM n Data Out Buffer with SCI (HIPMnDOC)

HIPMnDOC has the same function as the HIPMnDO register. In addition, it generates an $\overline{\text{ECSCI}}$ interrupt when OBF is set and hardware $\overline{\text{ECSCI}}$ generation is enabled.

Location: PM1 - FF F792h
PM2 - FF F7A4h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**PM Channel DBBOUT Data**							

Bit	Description
7-0	**PM Channel DBBOUT Data.**

Host Interface PM n Data Out Buffer with SMI (HIPMnDOM)

HIPMnDOM has the same function as the HIPMnDO register. In addition, it generates an $\overline{\text{SMI}}$ interrupt when OBF is set and hardware $\overline{\text{SMI}}$ generation is enabled.

Location: PM1 - FF F794h
PM2 - FF F7A6h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**PM Channel DBBOUT Data**							

Bit	Description
7-0	**PM Channel DBBOUT Data.**

5.0 Host-to-EC Interface Modules (Continued)

Host Interface PM n Data In Buffer with SCI (HIPMnDIC)

HIPMnDIC has the same function as the HIPMnDI register. In addition, it generates an $\overline{\text{ECSCI}}$ interrupt when IBF is cleared and when hardware $\overline{\text{ECSCI}}$ generation is enabled.

Location: PM1 - FF F796h
PM2 - FF F7A8h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**PM Channel DBBIN Data**							

Bit	Description
7-0	**PM Channel DBBIN Data.**

Host Interface PM n Control Register (HIPMnCTL)

HIPMnCTL controls the operation mode and configuration of the PM channel. It includes the Enhanced mode enable bit and control bits for Enhanced mode operation. It is set to 40h on reset.

Location: PM1 - FF F798h
PM2 - FF F7AAh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**EME**	**SCIPOL**	**PLMS**			**Reserved**	**OBEIE**	**IBFIE**
Reset	see text	1	0	0	0	0	0	0

Bit	Description
7	**EME (Enhanced Mode Enable).** 0: PM channel is used in Legacy mode. HIPMnST status bits are controlled by direct writes to the bit; interrupts are controlled by HICTRL and HIIRQC register bits (default for HIPM1CTL). 1: Enables enhanced control of the PM channel. The bits in HICTRL and HIIRQC registers are ignored in this case (default for HIPM2CTL). In HIPM2CTL (i.e., for PM channel 2), EME is a read-only bit that holds the value 1. Writes to this bit are ignored.
6	**SCIPOL (SCI Negative Polarity).** 0: $\overline{\text{ECSCI}}$ output inactive value is low and its active (asserted) value is high. 1: Inverted polarity is used. When SCIPOL is set, the $\overline{\text{ECSCI}}$ signal is the inverse of either what is stored in SCIB or the output of the $\overline{\text{ECSCI}}$ pulse shaper (default). This bit affects the $\overline{\text{ECSCI}}$ signal polarity in both PC87570 Legacy and Enhanced modes. Note: For the correct operation of SCI Source Gathering (see Figure 49 on page 204), SCIPOL bit must be set to 1 (i.e., to generate an active-low output.
5-3	**PLMS (Pulse Level Mode SCI).** Sets the hardware-controlled $\overline{\text{ECSCI}}$ signal mode to be Level or Pulse and sets the pulse width. When PLMS = 000b, the $\overline{\text{ECSCI}}$ signal functions in Level mode. In this mode, the $\overline{\text{ECSCI}}$ pulse shaper output value is low, and a high level is set to issue an interrupt (the respective OBF is set). When PLMS ≠ 0, the host interrupts are in Pulse mode. In this mode, the $\overline{\text{ECSCI}}$ pulse shaper output value is low, and it toggles high to issue an interrupt (i.e., when the respective output buffer register is written). **Bits** **5 4 3 Pulse Width** 0 0 0: Level interrupt (default). 0 0 1: 1 core clock (CLK) cycle pulse. 0 1 0: 2 core clock (CLK) cycle pulse. 0 1 1: 4 core clock (CLK) cycle pulse. 1 0 0: 8 core clock (CLK) cycle pulse. 1 0 1: 16 core clock (CLK) cycle pulse. Other: Reserved.

Bit	Description
2	**Reserved.**
1	**OBEIE (Output Buffer Empty Interrupt Enable).** 0: Interrupt to the core by OBF is disabled (default). 1: Enables an interrupt to the core when OBF bit in HIPMnST register is cleared to 0 (i.e., the output buffer is empty).
0	**IBFIE (Input Buffer Full Interrupt Enabler).** 0: IBF interrupt to the core is disabled (default). 1: Enables an interrupt to the core when IBF bit in HIPMnST register is set to 1.

Host Interface PM n Interrupt Control Register (HIPMnIC)

HIPMnIC affects operation in Enhanced mode only (i.e., when EME bit in HIPMnCTL register is set). In PC87570 Legacy mode, the bits in this registers are ignored. HIPMnIC controls the PM n interrupt signals mode of operation. It is set to 41h on reset.

Location: PM1 - FF F79Ah

PM2 - FF F7ACh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**SCIIS**	**SMIPOL**	**PLMM**			**SCIB**	**SMIB**	**IRQB**
Reset	0	1	0	0	0	0	0	1

Bit	Description
7	**SCIIS (SCI on IBF Start).** A write of 1 starts an $\overline{\text{ECSCI}}$ interrupt when IBF is cleared. A write of 0 is ignored. When read, always returns 0.
6	**SMIPOL (SMI Negative Polarity).** 0: $\overline{\text{SMI}}$ output inactive value is low and its active (asserted) value is high. 1: Inverted polarity is used. When SMIPOL is set, the $\overline{\text{SMI}}$ signal is either the inverse of what is stored in SMIB or the output of the $\overline{\text{SMI}}$ pulse shaper (default). This bit affects the $\overline{\text{SMI}}$ signal polarity in both PC87570 Legacy and Enhanced mode. Note: For the correct operation of SMI Source Gathering (see Figure 49 on page 204), SMIPOL bit must be set to 1 (i.e., to generate an active-low output.
5-3	**PLMM (Pulse Level Mode SMI).** Sets the hardware-controlled $\overline{\text{SMI}}$ signal mode to Level or Pulse and sets the pulse width. When PLMM = 000b, the $\overline{\text{ECSCI}}$ signal functions in Level mode. In this mode, the $\overline{\text{SMI}}$ pulse shaper output value is low, and a high level is set to issue an interrupt (i.e., the respective OBF is set). When PLMM ≠ 0, the host interrupts are in Pulse mode. In this mode, the $\overline{\text{SMI}}$ pulse shaper output value is low, and it toggles high to issue an interrupt (i.e., when the respective output buffer register is written). **Bits** **5 4 3 Pulse Width** 0 0 0: Level interrupt (default). 0 0 1: 1 core clock (CLK) cycle pulse. 0 1 0: 2 core clock (CLK) cycle pulse. 0 1 1: 4 core clock (CLK) cycle pulse. 1 0 0: 8 core clock (CLK) cycle pulse. 1 0 1: 16 core clock (CLK) cycle pulse. Other: Reserved.
2	**SCIB (Host SCI Request Control Bit).** When the $\overline{\text{ECSCI}}$ signal is configured for direct control by the firmware (HSCIE in HIPMnIE register is 0), SCIB bit is output to the PMnSCI signal (if SCIPOL=0, SCIB is output; if SCIPOL=1, SCIB is inverted before output). When read, SCIB bit returns the current value of the $\overline{\text{ECSCI}}$ pin. The $\overline{\text{ECSCI}}$ signal value can be read regardless of the state of HSCIE bit in HIPMnIE register.

5.0 Host-to-EC Interface Modules (Continued)

Bit	Description
1	**SMIB (Host SMI Request Control Bit)**. When the $\overline{SMI}$ signal is configured for direct control by the firmware (HSMIE in HIPMnIE register is 0), SMIB bit is output to the PMnSMI signal (if SMIPOL=0, SMIB is output; if SMIPOL=1, SMIB is inverted before output). When read, SMIB bit returns the current value of the $\overline{SMI}$ pin. The $\overline{SMI}$ signal's value can be read regardless of the state of HSMIE in HIPMnIE register.
0	**IRQB (Host Interrupt Request Control Bit).** When the IRQ signal is configured for direct control by the firmware (HIRQE in HIPMnIE register is 0), IRQB bit is output to the PMnIRQ signal. When read, IRQB bit returns the current value of the PMnIRQ signal. IRQn signal's value can be read regardless of the state of HIRQE in HIPMnIE register.

Host Interface PM n Interrupt Enable Register (HIPMnIE)

HIPMnIE controls the PM n interrupt signals that enable $\overline{SMI}$, $\overline{ECSCI}$ and IRQ interrupts. It is cleared on reset.

Location: PM1 - FF F79Ch
PM2 - FF F7AEh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**HSTA**	**HSMIE**	**HSCIE**	**HIRQE**	**SMIE**	**SCIE**	**IRQE**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Reserved.**
6	**HSTA (Host Stall Access).** Enables stalling of the host if the host attempts to access a buffer that does not contain valid data. This mode eliminates the need to have the buffer status bits read by the host before reading/writing data to the PM Channel. 0: Host transaction is completed even if the buffer is not ready. Data loss may occur in this case (default). 1: Host read transaction from an empty output buffer or write transaction to a full input buffer is extended until the buffer is ready (i.e., input buffer is empty or output buffer holds valid data).
5	**HSMIE (Hardware SMI Enable).** Works only in Enhanced PM mode. 0: SMIB bit in HIPMnIC register controls the value of the $\overline{SMI}$ (default). 1: Enables the generation of $\overline{SMI}$ events by hardware control based on the status of the OBF flag.
4	**HSCIE (Hardware SCI Enable).** Works only in Enhanced PM mode. 0: SCIB bit in HIPMnIC register controls the value of the $\overline{ECSCI}$ (default). 1: Enables the generation of $\overline{ECSCI}$ events by hardware control based on the status of the OBF and IBF flags.
3	**HIRQE (Hardware IRQ Enable).** Works only in Enhanced PM mode. 0: IRQB bit of HIPMnIC register controls the value of the IRQ (default). 1: Enables the generation of IRQ events by hardware control based on the status of the OBF flag.
2	**SMIE (SMI Enable).** 0: PMnSMI signal is set to its default value (high) and no interrupts are issued (default). 1: Enables the generation of $\overline{SMI}$ events by the PM Channels module.
1	**SCIE (SCI Enable).** 0: PMnSCI signal is set to its default value (high) and no interrupts are issued (default). 1: Enables PM generation of $\overline{ECSCI}$ events by the PM Channels module.
0	**IRQE (IRQ Enable).** 0: PMnIRQ signal is set to its default value (low) and no interrupts are issued (default). 1: Enables PM generation of IRQ events by the PM Channels module.

5.0 Host-to-EC Controller Interface Modules (Continued)

5.3 CORE ACCESS TO HOST MODULES

The WPC8763L enables the core to access the host module registers (i.e., Host Configuration, Keyboard/Mouse interface, Power Management Channels 1 and 2, GPIO and MSWC).

Host Module Register Arbitration. Since the host processor software and the WPC8763L firmware cannot access a host module simultaneously, they must communicate to prevent conflicts in host module register usage.

Access to the host modules is controlled via a lock bit for each module in LKSIOHA register. When the relevant lock bit is cleared, access to the host modules registers by the host processor is enabled. When the relevant lock bit is set, access to the host module registers by the host processor is blocked. The WPC8763L returns a Long Wait SYNC code until the required module is released by the core. Any attempt by the host to access the locked register is flagged by setting the respective bit in SIOLV register.

Arbitration. The host and core should access the host modules only after preventing host access to the module (using lock bits, as explained in the previous paragraph). The module arbitrates access between the host and core. If a core transaction starts after an LPC transaction (to a different, unlocked module) has started, the core transaction waits for the completion of the LPC transaction. If a core transaction starts before an LPC transaction, the LPC transaction is stalled by Long Wait SYNC response until the core transaction ends.

The following sequence minimizes conflicts between the host and core when both use the host modules.

1. After arbitrating the use of the specific host modules with the host, set the corresponding lock bit; see LKSIOHA register.
2. Read and save all host module registers required for proper operation of the host. Beware of destructive reads (i.e., when status bits are cleared when registers are read).
3. After the host module access is complete, restore the host module registers saved in step 2.
4. Clear the corresponding lock bit to allow the host to access the host module.

The core can access the V_{CC}-powered registers in the host modules (including the Host configuration registers), regardless of their Activation bit in index 30h or of the state of V_{DD} power supply.

Core Read Access. To perform a core read access from a host module register:

1. Set CSAE bit in SIBCTRL register, if not already set.
2. Verify that both CSRD and CSWR bits in SIBCTRL register are cleared.
3. Select the module to be accessed by setting its respective bit in CRSMAE register, if not already set. All other bits in the register must be cleared.
4. In IHIOA register, specify the offset of the register from the module base address, if not already specified.
5. Write 1 to CSRD bit in SIBCTRL register.
6. Read CSRD bit in SIBCTRL until it returns 0.
7. Read the data from IHD register.

Core Write Access. To perform a core write access to a host module register:

1. Set CSAE bit in SIBCTRL register, if not already set.
2. Verify that both CSRD and CSWR bits in SIBCTRL register are cleared.
3. Select the module to be accessed by setting its respective bit in CRSMAE register, if not already set. All other bits in the register must be cleared.
4. In IHIOA register, specify the offset of the register from the module base address, if not already specified.
5. Write the data to IHD register; this starts the write access to the device.
6. Read CSWR bit in SIBCTRL until it returns 0; this indicates the completion of the write transaction.

Interrupts. The IRQ lines of all the host modules are routed to the core (via the MSWC, MIWU and ICU modules) as a single “Module IRQ”; see “Module IRQ Event” on page 233. Each module interrupt may be masked by the core. The interrupt request is cleared by either the host or the core clearing the pending interrupts in the host modules.

5.0 Host-to-EC Controller Interface Modules (Continued)

5.3.1 Core Access to Host Module Registers

The following set of registers is accessible only by the core. The registers are powered by V_{CC} and are set to their default values by Core Domain reset.

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Core Access to Host Modules Register Map

Location	Mnemonic	Register Name	Size	Type
FF F740h	IHIOA	Indirect Host I/O Address	Word	R/W
FF F742h	IHD	Indirect Host Data	Byte	R/W
FF F744h	LKSIOHA	Lock Host Access	Word	R/W
FF F746h	SIOLV	Access Lock Violation	Word	R/W1C
FF F748h	CRSMAE	Core to Host Modules Access Enable	Word	R/W
FF F74Ah	SIBCTRL	Module Control	Byte	Varies per bit

Indirect Host I/O Address Register (IHIOA)

IHIOA defines the host I/O offset address for read/write transactions from/to the host modules. The I/O address is an offset from the runtime base address of the host module. The accessed host module is selected using the CRSMAE register; see page 214.

Location: FF F740h

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Reserved**								**Indirect Host I/O Offset**							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description
15-8	**Reserved.**
7-0	**Indirect Host I/O Offset.** Only offsets within the runtime address range of the host module are allowed. Other offsets may have unpredictable results.

Indirect Host Data Register (IHD)

IHD holds host data for read/write transactions from/to the host modules.

Location: FF F742h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Indirect Host Data**							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**Indirect Host Data.** Before writing data to IHD register, CSAE bit in SIBCTRL register must be set to 1; otherwise, the write transaction will have unpredictable results.

5.0 Host-to-EC Controller Interface Modules (Continued)

Lock Host Access Register (LKSIOHA)

LKSIOHA locks host access to the host modules. All bits are cleared on reset.

Location: FF F744h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**					**LKSMEM**	**Reserved**	**LKMSWC**
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**LKHGPHA**	**Reserved**						**LKCFG**
Reset	0	0	0	0	0	0	0	0

Bit	Description
15-11	**Reserved.**
10	**LKSMEM (Lock Shared Memory Host Access).** 0: Host access to the Shared Memory registers is enabled (default). 1: Host access to the Shared Memory registers is blocked.
9	**Reserved.**
8	**LKMSWC (Lock Mobile System Wake-Up Control Host Access).** 0: Host access to the MSWC registers is enabled (default). 1: Host access to the MSWC registers is blocked.
7	**LKHGPHA (Lock Host GPIO Host Access).** 0: Host access to the Host GPIO registers is enabled (default). 1: Host access to the Host GPIO registers is blocked.
6-1	**Reserved.**
0	**LKCFG (Lock Configuration Registers Host Access).** 0: Host access to the configuration registers is enabled (default). 1: Host access to the configuration registers is blocked.

Access Lock Violation Register (SIOLV)

SIOLV provides an error indication when a host lock violation occurs on access to host modules.

Location: FF F746h

Type: R/W1C

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**					**SMEMLV**	**Reserved**	**MSWCLV**
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**HGPIOLV**	**Reserved**						**CFGLV**
Reset	0	0	0	0	0	0	0	0

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Description
15-11	**Reserved.**
10	**SMEMLV (Shared Memory Lock Violation).** Is set to1 when the host attempts to access the Shared Memory registers while LKSMEM bit in LKSIOHA register is set.
9	**Reserved.**
8	**MSWCLV (Mobile System Wake-Up Control Lock Violation).** Is set to1 when the host attempts to access the MSWC registers while LKMSWC bit in LKSIOHA register is set.
7	**HGPIOLV (Host GPIO Lock Violation).** Is set to1 when the host attempts to access the Host GPIO registers while LKHGPHA bit in LKSIOHA register is set.
6-1	**Reserved.**
0	**CFGLV (Configuration Register Lock Violation).** Is set to1 when the host attempts to access the configuration registers while LKCFG bit in LKSIOHA register is set.

Core to Host Modules Access Enable Register (CRSMAE)

CRSMAE enables core access to the host modules. At any time, only one of the bits in this register may be set to 1.

Location: FF F748h

Type: R/W

Bit	15	14	13	12	11	10	9	8
Name	**Reserved**					**SMEMAE**	**Reserved**	**MSWCAE**
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	**HGPIOAE**	**Reserved**						**CFGAE**
Reset	0	0	0	0	0	0	0	0

Bit	Description
15-11	**Reserved.**
10	**SMEMAE (Shared Memory Core Access Enable).** 0: Core access to the Shared Memory registers is disabled (default). 1: Core access to the Shared Memory registers is enabled.
9	**Reserved.**
8	**MSWCAE (Mobile System Wake-Up Control Access Enable).** 0: Core access to the MSWC registers is disabled (default). 1: Core access to the MSWC registers is enabled.
7	**HGPIOAE (Host GPIO Access Enable).** 0: Core access to the Host GPIO registers is disabled (default). 1: Core access to the Host GPIO registers is enabled.
6-1	**Reserved.**
0	**CFGAE (Configuration Registers Core Access Enable).** Enables access to the PnP Configuration index/data registers, with A0 of the offset used to differentiate between them as follows: — When A0 = 0 (e.g., offset = 00h), the index register is accessed. — When A0 = 1 (e.g., offset = 01h), the data register is accessed. 0: Core access to the configuration registers is disabled (default). 1: Core access to the configuration registers is enabled.

5.0 Host-to-EC Controller Interface Modules (Continued)

Module Control Register (SIBCTRL)

SIBCTRL allows the core to control the module operation.

Location: FF F74Ah

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**					**CSWR**	**CSRD**	**CSAE**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-3		**Reserved.**
2	RO	**CSWR (Core Write to Host Modules).** Is set to 1 by a write to IHD register. It returns to 0, when the write to the host module is completed. The contents of IHIOA and IHD registers must not be modified while CSWR is 1.
1	R/W1S	**CSRD (Core Read from Host Modules).** Writing 1 starts a read from the host module; the read is based on the offset and on the enabled module, specified in CRSMAE register. CSAE bit must be set to 1 before or together with writing 1 to CSRD; otherwise, the read transaction will have unpredictable results. A write of 0 is ignored. This bit returns to 0 when the read access is completed, indicating that the data is ready in IHD register. The contents of IHIOA and IHD registers must not be modified while CSRD is 1.
0	R/W	**CSAE (Core to Host Modules Access Enable).** 0: Core access to the host modules is disabled (default). 1: Core access to the host module is enabled. The module to be accessed is selected by the CRSMAE register.

5.0 Host-to-EC Controller Interface Modules (Continued)

5.4 SHARED MEMORY (SHM)

5.4.1 Overview

The SHM module allows sharing of the flash memory and on-chip RAM by both the core and the host. It maps the host memory address using several modes to support the following features:

- BIOS code storage
- Core/host communication via mailbox
- Core/host bulk data transfer

The SHM further supports a protection mechanism used to secure any data or code stored in the flash or on-chip RAM devices from unauthorized access. Host access protection is configured per 64 Kbyte flash blocks. Each block can be protected by both host and core.

For efficient host read accesses, a configurable prefetch of 16 bytes from flash memory is implemented. The host/core mailbox has a configurable size and base address.

SHM registers are divided into the following groups: core-controlled and host-controlled. The host registers control various parameters that are required for the host to access the different address ranges. Note that some of the host configuration registers are in Section 6.1.13 on page 263. The core registers configure the memory allocation and include status, protection and semaphore registers.

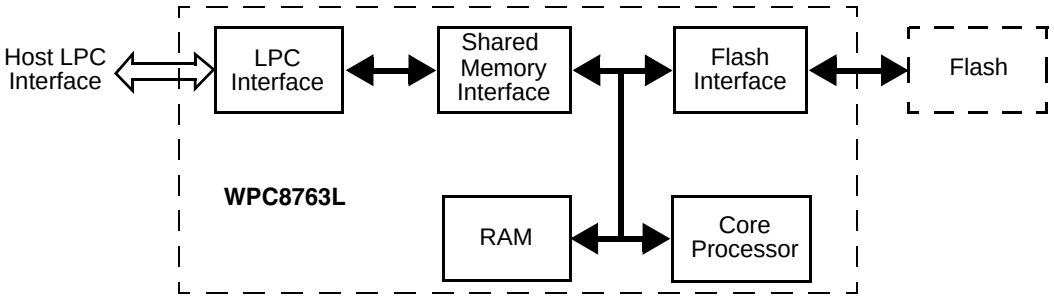


Figure 50. Shared Memory System Diagram

5.4.2 LPC Bus Transactions

The SHM module supports the following host transactions on the LPC bus:

- 8-bit I/O read/write - for the SHM host accessible registers
- 8-bit LPC memory (LPC) read/write - for the flash memory, RAM and semaphore registers
- 8-bit Firmware Memory (FWH) read/write - for the flash memory, RAM and semaphore registers

Up to two values of Firmware Memory transactions (FWH) ID are supported. They are configured via the BIOS_FWH_ID field (for the 386 Mode) and by the SHW_FWH_ID field, respectively. For all the other memory ranges, see SHM_CFG register on page 264.

5.4.3 Address Range Detection and Operation

The SHM supports the host address ranges shown in Table 25. Each range is enabled via a configuration bit; if range settings overlap, the ranges will be interpreted according to the “priority” defined in the table. The SHM returns a SYNC code only on an LPC transaction to a range that is supported and enabled by the SHM. An exception to this is when the LPC transaction is to a protected memory range. In this case, the LPC response is according to HERES field in SMC_CTL register.

Note that the allocated flash size ranges from 128 Kbytes to 4 Mbytes. The actual size is defined by FL_SIZE_P1 field in FLASH_SIZE register and is copied by the Booter from the flash after reset.

5.0 Host-to-EC Controller Interface Modules (Continued)

Table 25. Supported Host Address Ranges

Priority[1]	Range Description	Host Address Range	Allocation	Type	SHBM[2] Strap	Enable
1	386 Mode BIOS	FFC0 0000h - FFFF FFFFh	FL_SIZE_P1	LPC	0	BIOS_LPC_EN = 1
		(ID[3])FC0 0000h - (ID)FFF FFFFh ID = BIOS_FWH_ID	FL_SIZE_P1	FWH	0	BIOS_FWH_EN = 1
	Legacy BIOS	000F 0000h - 000F FFFFh	64 Kbyte	LPC	0	BIOS_LPC_EN = 1
	Extended Legacy BIOS	000E 0000h - 000E FFFFh	64 Kbyte	LPC	0	BIOS_LPC_EN = 1 and BIOS_EXT_EN = 1
2	Shared Flash Access Window[4]	Base = SHAW1BA_3-2 Size = FWIN1_SIZE	FWIN1_SIZE	LPC	X[5]	FLASH_ACC_EN = 1 and SHAW1BA_3-2 ≠ 0 and SHWIN_ACC = 0
		Above with ID[3] = SHW_FWH_ID	FWIN1_SIZE	FWH	X	Above and SHWIN_ACC = 1
–	Shared RAM Access Window 1[4]	Base = SHAW1BA_3-0 Size = RWIN1_SIZE	RWIN1_SIZE	LPC	X	FLASH_ACC_EN = 0 and SHAW1BA_3-0 ≠ 0 and WIN_BASE1 ≠ FFFFh and SHWIN_ACC = 0
		Above with ID[3] = SHW_FWH_ID	RWIN1_SIZE	FWH	X	Above and SHWIN_ACC = 1
–	Shared RAM Access Window 2	Base = SHAW2BA_3-0 Size = RWIN2_SIZE	RWIN2_SIZE	LPC	X	SHAW2BA_3-0 ≠ 0 and WIN_BASE2 ≠ FFFFh and SHWIN_ACC = 0
		Above with ID[3] = SHW_FWH_ID	RWIN2_SIZE	FWH	X	Above and SHWIN_ACC = 1

1. Priority 1 is highest; priority 2 is lowest. The Shared RAM Access windows 1 and 2 are not prioritized.
2. SHBM strap only sets the default value of BIOS_LPC_EN and BIOS_FWH_EN bits.
3. The address in the Firmware Memory space is 28 bits; the upper nibble is replaced by the ID value.
4. Shared Flash Access Window and Shared RAM Access Window 1 cannot be activated at the same time.
5. The SHBM strap setting is irrelevant.

386 Mode BIOS. This range is allocated at the top of the host memory according to the size set by FL_SIZE_P1 field. SHM responds to this range if the SHBM strap pin is sampled 0. The 386 Mode BIOS range can be accessed either by LPC or FWH transactions. Following a Host Domain Hardware reset, the first LPC transaction to the SHM automatically sets the type of transaction (LPC memory or FWH) used in the system. The automatic detection mechanism also sets the BIOS_LPC_EN and BIOS_FWH_EN bits accordingly. In FWH transactions, the required FWH ID is configured via BIOS_FWH_ID field in FWH_ID_LPC register.

When FWH transactions are enabled, the ID field of the transaction is compared with the value of BIOS_FWH_ID field in the core-controlled FWH_ID_LPC register.

Legacy BIOS. This range is allocated the 000F 0000h - 000F FFFFh address range (64 Kbytes) of host memory. To use this BIOS range, the SHBM strap pin must be sampled 0. This range responds only to LPC memory transactions.

Extended Legacy BIOS. This range is allocated the 000E 0000h - 000E FFFFh address range (64 Kbytes) of host memory. To use this BIOS range, the SHBM strap pin must be sampled 0 and BIOS_EXT_EN bit in SHM_CFG register must be set to 1.

Shared Flash Access Window 1. The parameters of this window are defined by the SHAW1BA_3 and SHAW1BA_2 registers (see Section 6.1.13 on page 263), which define the host base address. The host window size is defined by FWIN1_SIZE field of WIN_CFG register.

The SHM responds to LPC memory transactions within this range if FLASH_ACC_EN bit is set to 1 (i.e., RAM Access Window 1 range is disabled), the value of SHAW1BA_3-2 registers is ≠ 0 and SHWIN_ACC bit in WIN_CFG register is set to 0. If FWH transactions are used, SHWIN_ACC bit must be set to 1 and the required FWH ID must be configured via SHW_FWH_ID field in SHM_CFG register.

Shared RAM Access Window 1. The parameters of this window are defined by SHAW1BA_3 - SHAW1BA_0 registers, which define the host base address. The host window size is defined by RWIN1_SIZE field in WIN_SIZE register.

SHM responds to LPC memory transactions within this range if FLASH_ACC_EN bit is set to 0 (i.e., Flash Access Window range is disabled), the value of SHAW1BA_3-0 registers is ≠ 0, the value of WIN_BASE1 register is ≠ FFFFh and SHWIN_ACC bit in WIN_CFG register is set to 0. When FWH transaction are used to access this range, SHWIN_ACC bit must be set to 1 and the required FWH ID must be configured via SHW_FWH_ID field in SHM_CFG register.

5.0 Host-to-EC Controller Interface Modules (Continued)

Shared RAM Access Window 2. The parameters of this window are defined by SHAW2BA_3 - SHAW2BA_0 registers, which define the host base address. The host window size is defined by RWIN2_SIZE field of WIN_SIZE register.

The SHM responds to LPC memory transactions within this range if the value of SHAW2BA_3-0 registers is ≠ 0, the value of WIN_BASE2 register is ≠ FFFFh and SHWIN_ACC bit in WIN_CFG register is set to 0. If FWH transactions are used to access this range, SHWIN_ACC bit must be set to 1 and the required FWH ID must be configured via SHW_FWH_ID field in SHM_CFG register.

Note: Since transactions to Shared RAM Access windows 1 and 2 are not prioritized, their mapping in host address space must **not** be overlapping.

5.4.4 Host to Core Address Mapping

Note that the base address and memory allocation of the flash memory and RAM in the core address space can be found in the Section 1.4.1 on page 19.

386 Mode BIOS Range. Host addresses starting at FFFF FFFFh, ranging down within the flash size defined by FL_SIZE_P1, are mapped to core address space as shown in Figure 51. Address FFFF FFFFh in the host address space is mapped to the top of the flash memory.

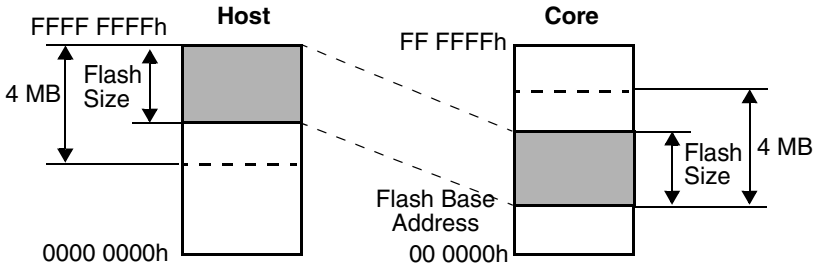


Figure 51. Mapping of 386 BIOS Mode into Flash Memory

Legacy BIOS. In this range, the SHM maps the host address into the top 64 Kbytes of the flash, as shown in Figure 52.

Extended Legacy BIOS. In this range, the SHM maps the host address into the core address space as shown in Figure 52. The host address 000E FFFFh is mapped to 64 Kbytes below the top of the flash memory.

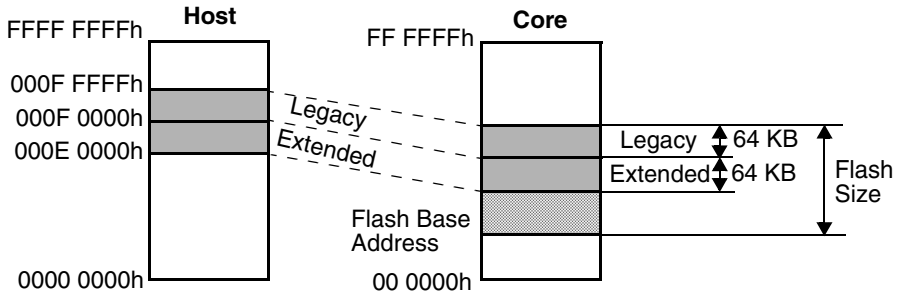


Figure 52. Mapping of BIOS and Extended Legacy BIOS Ranges into Flash Memory

Shared Flash Access Window 1. In this range, the host base address defined by SHAW1BA_2 and SHAW1BA_3 registers is mapped to the flash base address, as shown in Figure 53. The allocated access window size is defined by FWIN1_SIZE field.

5.0 Host-to-EC Controller Interface Modules (Continued)

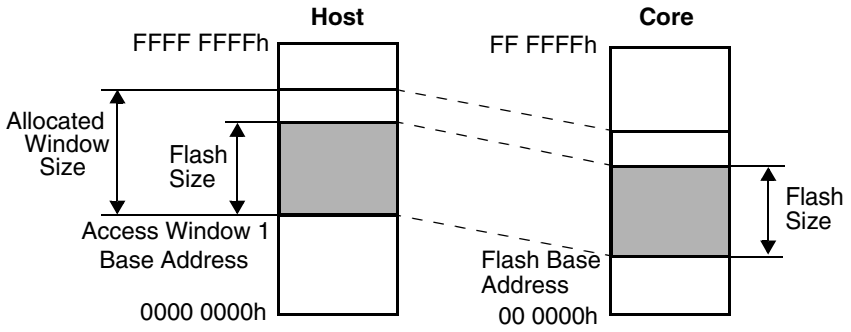


Figure 53. Address Mapping of Shared Flash Access Window 1 Range

Shared RAM Access Window 1. In this range, the SHM translates the address from the host address space to the core address space, as shown in Figure 54. The host base address defined by SHAW1BA_3-0 registers is mapped to the RAM base address defined by WIN_BASE1 register, the window size is defined by RWIN1_SIZE field in WIN_SIZE register.

Shared RAM Access Window 2. In this range, the SHM translates the address from the host address space to the core address space, as shown in Figure 54. The host base address defined by SHAW2BA_3-0 registers is mapped to the RAM base address defined by WIN_BASE2 register; the window size is defined by RWIN2_SIZE field in WIN_SIZE register.

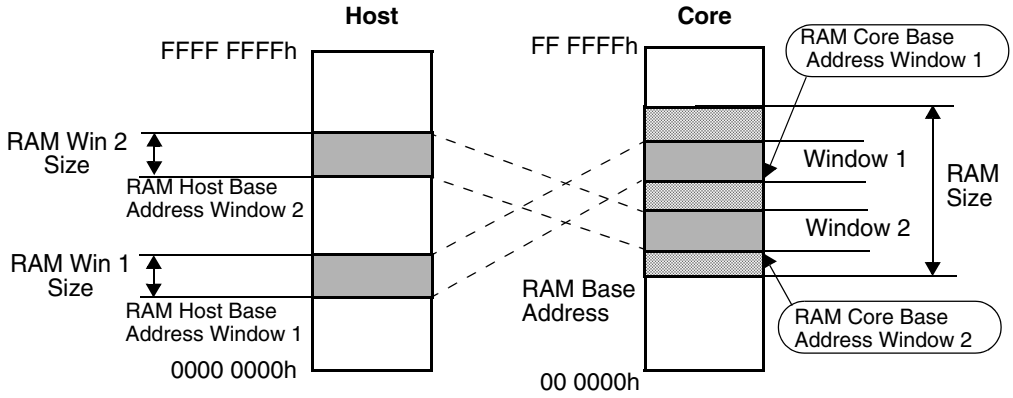


Figure 54. Address Mapping of Shared RAM Access Windows 1 and 2 Ranges

5.4.5 Access Protection

The SHM provides protection mechanisms for both flash memory and on-chip RAM against inadvertent access by the host.

Shared Flash Memory Access Protection

To protect specified flash memory areas from being read and/or written by the host, the SHM provides a dual-control mechanism as follows:

- Flash write protection is controlled by both host and core. There are separate bits that can be set independently by the host and the core. When either the host or core protection bit is set, the respective memory block becomes write protected. The core-controlled protection bits act as an override to the host-controlled protection bits.
  - Host-controlled write protection bits. These read/write bits can be locked independently. Once set, the lock bits can be cleared only by V_{DD} Power-Up reset or Host Hardware reset.
  - Core-controlled (override) write protection bits. These bits are write-once (i.e., when a bit is set to protect a memory block, it can be cleared only by Core Domain reset).
- Flash read protection uses a similar mechanism to the write protection described above.

Host-Controlled Read/Write Protection. The 4 Mbyte area allocated to the flash memory is divided into 64 memory blocks, of 64 Kbytes each. Each block is protected by separate read and write protection bits. In addition, lock bit is implemented to prevent unauthorized write to the protection bits.

The protection bits are accessed using an indexed mechanism, via registers SMHAP0-3. Each register controls the protection for a 1 Mbyte memory segment (16 memory blocks).

5.0 Host-to-EC Controller Interface Modules (Continued)

To modify the protection and lock bits:

1. Read and save the Index field.
2. Write the new Index field with Index Write bit set to 1.
3. Read and modify the Read/Write Protection and Lock bits.
4. Write both the new Index field and the modified Read/Write Protection and Lock bits with Index Write bit set to 0.
5. Restore the saved Index field with Index Write bit set to 1.

Core-Controlled (Override) Write Protection. The write protection bits are accessed via registers SMCOWP0-3. Each register controls the protection for 1 Mbyte memory segment (16 memory blocks). It is not necessary to perform read-modify-write to set write protection bits. Bits are write-once (i.e., a set bit can be altered only by reset).

Core-Controlled (Override) Read Protection. The read protection bits are accessed via registers SMCORP0-3. The registers' functionality is similar to the core-controlled write protection described above.

On-Chip RAM Access Protection

To protect the on-chip RAM from inadvertent host access, each RAM access window has four protection bits in WIN_PROT register: RWnL_RP and RWnL_WP (where n =1 or 2) protect the lower half of the RAM from unauthorized reads and writes, respectively. RWnH_RP and RWnH_WP protect the upper half of the RAM from unauthorized access.

Note that the Semaphore registers, which are mapped at offset 00h of each on-chip RAM window, are excluded from this protection mechanism.

5.4.6 Response to Host Access

Host Access to Protected Area. If the host tries to access a protected shared memory area (either flash or RAM), the SHM performs the following:

- Sets the respective HWERR or HRERR bit in SMC_STS register to 1. If the HERR_IE bit in SMC_CTL register is set, an interrupt is also generated to the ICU.
- Responds to the LPC transaction according to the setting of the HERES field in SMC_CTL register; see page 224.

Host Access to Unprotected Area. If the host accesses an unprotected shared memory area (either flash or RAM, not including the semaphore registers), the SHM performs the following:

- Sets the SHM_ACC bit in SMC_STS register to 1. If the SHM_ACC_IE bit in SMC_CTL register is set, an interrupt is also generated to the ICU.
- Responds to the LPC transaction according to the setting of the STALL_HOST bit in FLASH_SIZE register; see page 225.

5.4.7 Transaction Types

If the current transaction is valid (i.e., an access to a supported, enabled, not-protected SHM address), the SHM module performs one of the following types of transactions:

- Normal transactions, as follows:
  - One-byte LPC **read** or **write** transactions.
- Transactions with prefetch must be 16-byte aligned. This type is supported **only** when reading from the flash memory (not from core RAM). Prefetch transactions are disabled, when PREF_EN bit in SMC_CTL register is set to 0. Since prefetch speeds up host reads but slows down core operations, PREF_EN bit should be set to 1 only during BIOS boot.

The SHM responds with one or more cycles of Long Wait SYNC until the internal data transfer is completed.

5.0 Host-to-EC Controller Interface Modules (Continued)

5.4.8 Host-Core Synchronization

To synchronize the host and core operation, host access should be enabled only after the initialization of the core-controlled registers; see HOSTWAIT bit in SHM_CTL register and "Host Configuration Address Selection" on page 235. In addition, the SHM supports the LPC transaction stall operation (see STALL_HOST bit in FLASH_SIZE register) to prevent host access to any SHM address range during specific core operations (for example, flash erase).

Semaphore Interface

The Host-Core Semaphore interface may be used for communication associated with flash memory updates or other data transfer. This interface is based on two semaphore registers. Each register is each located in two places: at the base address of the respective Shared RAM Access Window in the host memory space, and at a fixed address in the core memory space. The signaling interface is designed mainly for polling-based operations, however it supports interrupting the core when the semaphore is written by the host.

Each 8-bit semaphore register can be read by both host and core. The host can write HSEM3-0 bits only and the core can write CSEM3-0 bits only, as shown in Figure 55. A host write to the semaphore sets HSEMnW bit (where n =1 or 2) in SMC_STS register to 1. If the respective HSEMn_IE bit in SMC_CTL register is set to 1, an interrupt is generated to the ICU.

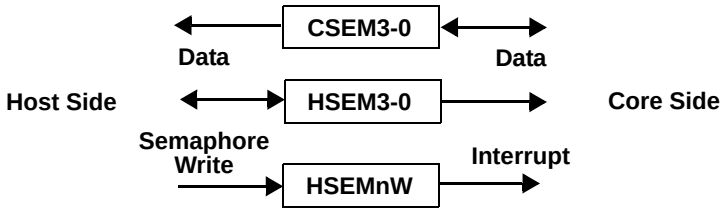


Figure 55. Semaphore Interface

5.4.9 SHM Host Registers

The following set of registers is accessible only by the host. All the registers are powered by V_{DD} and are set to their default values by V_{DD} Power-Up or Hardware reset except for the SHAW1-2_SEM registers; see page 223.

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

SHM Host Register Map

Base Address Registers	Offset	Mnemonic	Register Name	Type
Base Address at index 60h, 61h (I/O access)	00h	SMHAP0	Shared Memory Host Access Protect 0	Varies per bit
	01h	SMHAP1	Shared Memory Host Access Protect 1	Varies per bit
	02h	SMHAP2	Shared Memory Host Access Protect 2	Varies per bit
	03h	SMHAP3	Shared Memory Host Access Protect 3	Varies per bit
SHAW1BA_3-0 (Memory access)	00h[1]	SHAW1_SEM	Shared Access Window 1, Semaphore	Varies per bit
SHAW2BA_3-0 (Memory access)	00h[2]	SHAW2_SEM	Shared Access Window 2, Semaphore	Varies per bit

1. Only if Shared Access Window 1 is enabled when FLASH_ACC_EN = 0.
2. Only if Shared Access Window 2 is enabled.

5.0 Host-to-EC Controller Interface Modules (Continued)

Shared Memory Host Access Protect 0-3 Register (SMHAP0-3)

These registers hold the shared memory read/write host-controlled protection and lock information. Each register is mapped into a 1 Mbyte memory segment; each index value is mapped into a one 64 Kbyte memory block. In each register, index 0 is mapped to the first (lowest) memory block and index 15 (Fh) is mapped to the last block (highest) of each memory segment, respectively. For more information, see "Host-Controlled Read/Write Protection." on page 219.

Protection Mapping

- SMHAP0 controls the first memory segment, blocks 0-15 (flash addresses 0-1 Mbyte).
- SMHAP1 controls the second memory segment, blocks 16-31 (flash addresses 1-2 Mbytes).
- SMHAP2 controls the third memory segment, blocks 32-47 (flash addresses 2-3 Mbytes).
- SMHAP3 controls the fourth memory segment, blocks 48-63 (flash addresses 3-4 Mbytes).

Location: Offset 00h-03h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Host Access Protection Index**				**Index Write**	**Host Lock Protection**	**Host Write Protection**	**Host Read Protection**
Reset	0	0	0	0	0	0	1	0

Bit	Type	Description
7-4	R/W or RO	**Host Access Protection Index.** Holds the index number used to access bits 3-0 of this register. Index = Block_First_Address / 64 Kbytes. In SMHAP0: 0h-Fh for blocks 0-15, respectively. In SMHAP1: 0h-Fh for blocks 16-31, respectively. In SMHAP2: 0h-Fh for blocks 32-47, respectively. In SMHAP3: 0h-Fh for blocks 48-63, respectively.
3	WO	**Index Write.** When set to 1, indicates an index write to SMHAPn register; therefore, bits 1-2 of this register are ignored. When read, always returns 0. 0: Writes to SMHAPn register affect all fields of the register (writes to bits 0-2 use the newly written index) (default). 1: Writes to SMHAPn register affects only the index; bits 1-2 are ignored.
2	R/W or RO	**Host Lock Protection.** When set to 1, locks the values of the Host Read Protection bit, Host Write Protection bit, and Host Lock Protection bit for the block number pointed to by the index field (bits 7-4). Once set to 1, Host Lock Protection bit can be cleared to 0 only by V_{DD} Power-Up reset or Host Hardware reset. 0: Bits 0-2 of this register can be changed (default for all the blocks). 1: Bits 0-2 of this register are locked (i.e., cannot be changed).
1	R/W or RO	**Host Write Protection.** When set to 1, disables write transactions to the block pointed to by the index field (bits 7-4). Note that the Core Override write protect bits may disable writes even when writes are enabled by the Host Write Protection bit. 0: Shared memory write accesses are enabled (flash program and erase) for this block. 1: Shared memory write accesses (flash program and erase) are disabled for this block (default for all the blocks).
0	R/W or RO	**Host Read Protection.** When set to 1, disables read transactions from the block pointed to by the index field (bits 7-4). Note that the Core Override read protect bits can disable reads even when reads are enabled by the Host Read Protection bit. 0: Shared memory reading is enabled for this block (default for all the blocks). 1: Shared memory reading is disabled for this block.

5.0 Host-to-EC Controller Interface Modules (Continued)

Shared Access Window 1-2, Semaphore Register (SHAW1-2_SEM)

Bit	Type	Description
7-0	R/W	**IMA23-16.** Holds address bits 23-16 of the core address space, for Indirect Memory Access core transactions.

SHAW1-2_SEM provide eight semaphore bits between the core and host. Four of the bits may be set by the host and four may be set by the core. These registers are accessible by the host only while the respective RAM window is enabled.

Location: Offset 00h (from base address in SHAW1-2BA3-0 registers)

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**CSEM3**	**CSEM2**	**CSEM1**	**CSEM0**	**HSEM3**	**HSEM2**	**HSEM1**	**HSEM0**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-4	RO	**CSEM3-0 (Core Semaphore Bits 3-0).** These bits can be written by the core and can be read by both the host and core. They are reset by Core Domain reset.
3-0	R/W	**HSEM3-0 (Host Semaphore Bits 3-0).** These bits can be written by the host and can be read by both the host and core. They are reset by V_{DD} Power-Up reset or Host Hardware reset.

5.4.10 SHM Core Registers

The following set of registers is accessible only by the core. All the registers are powered by V_{CC} and are reset by Core Domain reset except for the SHAW1-2_SEM registers; see page 228.

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Shared Memory Core Register Map

Location	Mnemonic	Register Name	Size	Type
FF 0400h	SMC_STS	Shared Memory Core Status	Byte	R/W1C
FF 0401h	SMC_CTL	Shared Memory Core Control	Byte	Varies per bit
FF 0402h	FLASH_SIZE	Flash Memory Size	Byte	R/W
FF 0403h	FWH_ID_LPC	FWH ID on LPC Bus	Byte	R/W
FF 0406h	WIN_PROT	RAM Windows Protection	Byte	R/W
FF 0407h	WIN_SIZE	RAM Windows Size	Byte	R/W
FF 0408h[1]	SHAW1_SEM	Shared Access Window 1, Semaphore	Byte	Varies per bit
FF 0409h[2]	SHAW2_SEM	Shared Access Window 2, Semaphore	Byte	Varies per bit
FF 040Ah	WIN_BASE1	RAM Window 1 Base Address	Word	R/W
FF 040Ch	WIN_BASE2	RAM Window 2 Base Address	Word	R/W
FF 0410h	SMCORP0	Shared Memory Core Override Read Protect 0	Word	R/W1S
FF 0412h	SMCORP1	Shared Memory Core Override Read Protect 1	Word	R/W1S
FF 0414h	SMCORP2	Shared Memory Core Override Read Protect 2	Word	R/W1S
FF 0416h	SMCORP3	Shared Memory Core Override Read Protect 3	Word	R/W1S
FF 0418h	SMCOWP0	Shared Memory Core Override Write Protect 0	Word	R/W1S
FF 041Ah	SMCOWP1	Shared Memory Core Override Write Protect 1	Word	R/W1S
FF 041Ch	SMCOWP2	Shared Memory Core Override Write Protect 2	Word	R/W1S

5.0 Host-to-EC Controller Interface Modules (Continued)

Location	Mnemonic	Register Name	Size	Type
FF 041Eh	SMCOWP3	Shared Memory Core Override Write Protect 3	Word	R/W1S

1. Only if Shared Access Window 1 is enabled when FLASH_ACC_EN = 0.
2. Only if Shared Access Window 2 is enabled.

Shared Memory Core Status Register (SMC_STS)

Location: FF 0400h

Type: R/W1C

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**SHM_ACC**	**HSEM2W**	**HSEM1W**	**Reserved**		**HWERR**	**HRERR**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7		**Reserved.**
6	RW/1C	**SHM_ACC (Host Access to Shared Memory).** Is set to 1 on host access to any supported and unprotected shared memory address (except for the two semaphore registers, SHAW1_SEM and SHAW2_SEM). When SHM_ACC changes to 1, an SHM interrupt to the ICU is generated if SHM_ACC_IE bit in SMC_CTL register is set to 1. Writing 1 clears this bit; writing 0 is ignored. 0: No host access to unprotected shared memory address occurred (default). 1: The host accessed a valid and unprotected shared memory address.
5	RW/1C	**HSEM2W (Host Semaphore 2 Written).** Is set to 1 when the host writes to the SHAW2_SEM register. Writing 1 clears this bit; writing 0 is ignored. 0: No semaphore write (default). 1: SHAW2_SEM semaphore register was written by host.
4	RW/1C	**HSEM1W (Host Semaphore 1 Written).** Is set to 1 when the host writes to the SHAW1_SEM register. Writing 1 clears this bit; writing 0 is ignored. 0: No semaphore write (default). 1: SHAW1_SEM semaphore register was written by host.
3-2		**Reserved.**
1	RW/1C	**HWERR (Host Write Access Error).** Is set to 1 when the host attempts to write to a write-protected block. Writing 1 clears this bit; writing 0 is ignored. 0: No write attempt to write-protected block (default). 1: Host tried to write to a write-protected block.
0	RW/1C	**HRERR (Host Read Access Error).** Is set to 1, when the host attempts to read from a read-protected block. Writing 1 clears this bit; writing 0 is ignored. 0: No read attempt from read-protected block (default). 1: Host tried to read from a read-protected block.

Shared Memory Core Control Register (SMC_CTL)

Location: FF 0401h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**HOSTWAIT**	**PREF_EN**	**SHM_ACC _IE**	**HSEM2_IE**	**HSEM1_IE**	**HERR_IE**	**HERES**	
Reset	1	1	0	0	0	0	0	0

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Type	Description
7	R/W1C	**HOSTWAIT (Host Access Stall).** When set to 1, any LPC transaction to the WPC8763L device is extended by a Long Wait SYNC field. HOSTWAIT is set to 1 after reset to allow the WPC8763L to be initialized by the Booter. When the device initialization process is finished, the firmware should clear HOSTWAIT to 0. The extended LPC transaction (if applicable) is thus released, and the host can freely communicate with the device. Writing 1 clears this bit; writing 0 is ignored. 0: Normal LPC access to WPC8763L. 1: LPC access to WPC8763L during initialization is extended by Long Wait SYNC (default).
6	R/W	**PREF_EN (Prefetch Enable).** When set to 1, enables the SHM to perform a 16-byte read prefetch following a host read from an address on a 16-byte boundary. 0: Disabled. 1: Enabled (default).
5	R/W	**SHM_ACC_IE (Enable Interrupt by Host Access to Shared Memory).** When set to 1, and SHM_ACC bit is also set, the SHM generates an interrupt to the ICU. 0: Interrupt disabled (default). 1: Interrupt enabled.
4	R/W	**HSEM2_IE (Enable Interrupt by Host Semaphore 2 Written).** When set to 1, and HSEM2W bit is also set, the SHM generates an interrupt to the ICU. 0: Interrupt disabled (default). 1: Interrupt enabled.
3	R/W	**HSEM1_IE (Enable Interrupt by Host Semaphore 1 Written).** When set to 1, and HSEM1W bit is also set, the SHM generates an interrupt to the ICU. 0: Interrupt disabled (default). 1: Interrupt enabled.
2	R/W	**HERR_IE (Enable Interrupt by Host Access Errors).** When set to 1, and HWERR or HRERR bit is also set, the SHM generates an interrupt to the ICU. 0: Interrupt disabled (default). 1: Interrupt enabled.
1-0	R/W	**HERES (Host Error Response).** Controls the WPC8763L response to an LPC/FWH transaction attempting either to write to a write-protected block or read from a read-protected block. When HERES field is updated, the new value takes effect starting from the next LPC transaction. **Bits 1 0 Transaction Response** 0 0: WPC8763L does **not** return SYNC (default). 0 1: WPC8763L ignores write transactions and returns 00h data on read transactions. 1 0: WPC8763L returns Long Wait SYNC until the block protection is removed (changing the HERES field does not terminate the Long Wait SYNC). 1 1: Reserved.

Flash Memory Size Register (FLASH_SIZE)

Location: FF 0402h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**STALL_ HOST**	**FL_SIZE_P1**					
Reset	0	0	1	0	0	0	0	1

Bit	Type	Description
7		**Reserved.**

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Type	Description
6	R/W	**STALL_HOST (Stall Host Access to Shared Memory).** When set to 1, any host access to a supported and unprotected shared memory address (except for the two semaphore registers, SHAW1_SEM and SHAW2_SEM) is extended ("stalled") by a Long Wait SYNC field. The transaction is resumed (by a Ready SYNC field) when STALL_HOST bit is set back to 0. 0: A host access to any supported **and** unprotected shared memory address terminates normally (default). 1: A host access to any supported **and** unprotected shared memory address is stalled by Long Wait SYNC.
5-0	R/W	**FL_SIZE_P1 (Flash Size Plus 1).** Selects the size of the connected flash for SHM operations. Must be configured to the same value as FL_SIZE_P1 field in FIU_CFG register; see page 59. The FL_SIZE_P1 in this register is updated by the Booter from the flash after reset. FL_SIZE_P1 = (Number of 128 Kbyte blocks contained by the flash) + 1. Valid values for FL_SIZE_P1 range from 02h to 21h (default); all other values are reserved.

FWH ID on LPC Bus Register (FWH_ID_LPC)

Location: FF 0403h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**				**BIOS_FWH_ID**			
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-4		**Reserved.**
3-0	R/W	**BIOS_FWH_ID (BIOS FWH ID Value).** Configures the FWD ID value used for 386 Mode BIOS range access.

RAM Windows Protection Register (WIN_PROT)

Location: FF 0406h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**RW2H_WP**	**RW2H_RP**	**RW2L_WP**	**RW2L_RP**	**RW1H_WP**	**RW1H_RP**	**RW1L_WP**	**RW1L_RP**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W	**RW2H_WP (RAM Access Window 2 High, Write Protect).** When set to 1, host write to the upper half of the Shared Access RAM Window 2 is disabled (write protected). 0: Enabled (default). 1: Disabled.
6	R/W	**RW2H_RP (RAM Access Window 2 High, Read Protect).** When set to 1, host read from the upper half of the Shared Access RAM Window 2 is disabled (read protected). 0: Enabled (default). 1: Disabled.
5	R/W	**RW2L_WP (RAM Access Window 2 Low, Write Protect).** When set to 1, host write to the lower half of the Shared Access RAM Window 2 is disabled (write protected). 0: Enabled (default). 1: Disabled[1].

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Type	Description
4	R/W	**RW2L_RP (RAM Access Window 2 Low, Read Protect).** When set to 1, host read from the lower half of the Shared Access RAM Window 2 is disabled (read protected). 0: Enabled (default). 1: Disabled[1].
3	R/W	**RW1H_WP (RAM Access Window 1 High, Write Protect).** The same as bit 7 of this register, but for RAM Access Window 1.
2	R/W	**RW1H_RP (RAM Access Window 1 High, Read Protect).** The same as bit 6 of this register, but for RAM Access Window 1.
1	R/W	**RW1L_WP (RAM Access Window 1 Low, Write Protect).** The same as bit 5 of this register, but for RAM Access Window 1.
0	R/W	**RW1L_RP (RAM Access Window 1 Low, Read Protect).** The same as bit 4 of this register, but for RAM Access Window 1.

1. This excludes the Semaphore registers of Shared Access Window 1 and 2, which are mapped at offset 00h of the RAM window.

RAM Windows Size Register (WIN_SIZE)

Location: FF 0407h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**RWIN2_SIZE**				**RWIN1_SIZE**			
Reset	0	1	0	0	0	1	0	0

Bit	Type	Description
7-4	R/W	**RWIN2_SIZE (RAM Access Window 2, Size Select).** Selects the size of Shared Access RAM Window 2. **Bits 7 6 5 4 — RAM Window Size** 0 0 0 0: 1 byte 0 0 0 1: 2 bytes 0 0 1 0: 4 bytes 0 0 1 1: 8 bytes 0 1 0 0: 16 bytes (default) 0 1 0 1: 32 bytes 0 1 1 0: 64 bytes 0 1 1 1: 128 bytes 1 0 0 0: 256 bytes 1 0 0 1: 512 bytes 1 0 1 0: 1 Kbyte 1 0 1 1: 2 Kbytes 1 1 0 0: 4 Kbytes Others: Reserved

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Type	Description
3-0	R/W	**RWIN1_SIZE (RAM Access Window 1, Size Select).** The same as bits 7-4 but affecting RAM Access Window 1. This field is relevant only while RAM access is in use (i.e., when FLASH_ACC_EN bit is 0). **Bits 3 2 1 0 — RAM Window Size** 0 0 0 0: 1 byte 0 0 0 1: 2 bytes 0 0 1 0: 4 bytes 0 0 1 1: 8 bytes 0 1 0 0: 16 bytes (default) 0 1 0 1: 32 bytes 0 1 1 0: 64 bytes 0 1 1 1: 128 bytes 1 0 0 0: 256 bytes 1 0 0 1: 512 bytes 1 0 1 0: 1 Kbyte 1 0 1 1: 2 Kbytes 1 1 0 0: 4 Kbytes Others: Reserved

Shared Access Window 1-2, Semaphore Register (SHAW1-2_SEM)

SHAW1-2_SEM provides eight semaphore bits between the core and host. Four of the bits may be set by the host and four may be set by the core. These registers are accessible by the host only while the respective RAM window is enabled.

Location: SHAW1_SEM: FF 0408h
SHAW2_SEM: FF 0409h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**CSEM3**	**CSEM2**	**CSEM1**	**CSEM0**	**HSEM3**	**HSEM2**	**HSEM1**	**HSEM0**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-4	R/W	**CSEM3-0 (Core Semaphore Bits 3-0).** These bits can be written by the core and can be read by both the host and the core. They are reset by Core Domain reset.
3-0	RO	**HSEM3-0 (Host Semaphore Bits 3-0).** These bits can be written by the host and can be read by both the host and the core. They are reset by V_{DD} Power-Up reset or Host Hardware reset.

RAM Window 1 Base Register (WIN_BASE1)

Location: FF 040Ah

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**WIN_BASE**															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Type	Description
15-0	R/W	**WIN_BASE (RAM Access Window, Base Address).** Configures the offset of the RAM Access window from the RAM base address. E.g., if the RAM base address is 64 Kbytes and the offset in this register is set to 1 Kbyte, the RAM access window base address (in the core address space) is at 65 Kbytes in the core address space. The base address must be aligned to the window block size. WIN_BASE = (core base address of RAM window - core RAM Base Address) 0000h - 1000h: Valid values for WIN_BASE FFFFh: Window is disabled (default). Other: Reserved.

RAM Window 2 Base Register (WIN_BASE2)

Location: FF 040Ch

Type: R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**WIN_BASE**															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Type	Description
15-0	R/W	**WIN_BASE (RAM Access Window, Base Address).** Configures the base address of Shared Access RAM Window 2. WIN_BASE field, bit mapping and valid values are the same as for WIN_BASE field in WIN_BASE1 register.

Shared Memory Core Override Read Protect Register 0 (SMCORP0)

SMCORP0-3 registers contain core-controlled protection bits for host read access to the shared memory. The host can read from a block of shared memory only when both the Host Read Protection bit in SMHAP0-3 registers and ORPn bit in SMCORP0-3 registers are set to 0. The allocated 4 Mbytes of shared memory is partitioned into 64 Kbyte blocks.

Protection Mapping

- SMCORP0 controls the first memory segment, blocks 0-15 (addresses 0-1 Mbyte).
- SMCORP1 controls the second memory segment, blocks 16-31 (addresses 1-2 Mbytes).
- SMCORP2 controls the third memory segment, blocks 32-47 (addresses 2-3 Mbytes).
- SMCORP3 controls the fourth memory segment, blocks 48-63 (addresses 3-4 Mbytes).

In each read protection register, bit 0 is mapped to the first (lowest) memory block, bit 1 is mapped to the second memory block and so on such that bit 15 (Fh) is mapped to the last block (highest) of each memory segment.

Location: FF 0410h

Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**ORP15-0**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**ORP15-0 (Override Read Protect for Blocks 15-0).** Each bit in this field sets the read protection of a host access to a single shared memory block. This register protects memory blocks 15-0. After being set by writing, a bit of this field can be cleared to 0 only by reset (write 0 is ignored). 0: Reading from the respective shared memory block is enabled if the respective Host Read Protection bit in SMHAP0-3 registers is also set to 0 (default). 1: Reading from the respective shared memory block is disabled.

5.0 Host-to-EC Controller Interface Modules (Continued)

Shared Memory Core Override Read Protect Register 1 (SMCORP1)

Location: FF 0412h

Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ORP31-16															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**ORP31-16 (Override Read Protect for Blocks 31-16).** Same as ORP15-0 field in SMCORP0, but for shared memory blocks 31-16.

Shared Memory Core Override Read Protect Register 2 (SMCORP2)

Location: FF 0414h

Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ORP47-32															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**ORP47-32 (Override Read Protect for Blocks 47-32).** Same as ORP15-0 field in SMCORP0, but for shared memory blocks 47-32.

Shared Memory Core Override Read Protect Register 3 (SMCORP3)

Location: FF 0416h

Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ORP63-48															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**ORP63-48 (Override Read Protect for Blocks 63-48).** Same as ORP15-0 field in SMCORP0, but for shared memory blocks 63-48.

Shared Memory Core Override Write Protect Register 0 (SMCOWP0)

SMCOWP0-3 registers contain core-controlled protection bits for host write access to the shared memory. The host can write to a block of shared memory only when both the Host Write Protection bit in SMHAP0-3 registers and OWPn bit in SMCOWP0-3 registers are set to 0. The allocated 4 Mbytes of shared memory is partitioned into 64 Kbyte blocks.

Protection Mapping

- SMCOWP0 controls the first memory segment, blocks 0-15 (addresses 0-1 Mbyte).
- SMCOWP1 controls the second memory segment, blocks 16-31 (addresses 1-2 Mbytes).
- SMCOWP2 controls the third memory segment, blocks 32-47 (addresses 2-3 Mbytes).
- SMCOWP3 controls the fourth memory segment, blocks 48-63 (addresses 3-4 Mbytes).

In each write protection register, bit 0 is mapped, in the core address space, to the first (lowest) memory block, bit 1 is mapped to the second memory block and so on up to bit 15 (Fh), which is mapped to the last block (highest) of each memory segment.

5.0 Host-to-EC Controller Interface Modules (Continued)

Location: FF 0418h
Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**OWP15-0**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**OWP15-0 (Override Write Protect for Blocks 15-0).** Each bit in this field sets the write protection of a host access to a single shared memory block. This register protects memory blocks 15-0. After being set, the bits can be cleared to 0 only by reset (write 0 is ignored). 0: Writing to shared memory block “n” is allowed if the respective Host Write Protection bit in SMHAP0-3 registers is also set to 0 (default). 1: Writing to shared memory block “n” is disabled.

Shared Memory Core Override Write Protect Register 1 (SMCOWP1)

Location: FF 041Ah
Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**OWP31-16**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**OWP31-16 (Override Write Protect for Blocks 31-16).** Same as OWP15-0 field in SMCOWP0, but for shared memory blocks 31-16.

Shared Memory Core Override Write Protect Register 2 (SMCOWP2)

Location: FF 041Ch
Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**OWP47-32**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**OWP47-32 (Override Write Protect for Blocks 47-32).** Same as OWP15-0 field in SMCOWP0, but for shared memory blocks 47-32.

Shared Memory Core Override Write Protect Register 3 (SMCOWP3)

Location: FF 041Eh
Type: R/W1S

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**OWP63-48**															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Type	Description
15-0	R/W1S	**OWP63-48 (Override Write Protect for Blocks 63-48).** Same as OWP15-0 field in SMCOWP0, but for shared memory blocks 63-48.

5.0 Host-to-EC Controller Interface Modules (Continued)

5.5 MOBILE SYSTEM WAKE-UP CONTROL (MSWC)

The MSWC detects and handles wake-up events from various sources in the host modules. The MSWC generates interrupts to the host via $\overline{SMI}$ or $\overline{PWUREQ}$, and/or alerts the core, which enables the core to control the wake-up sequence. Since the MSWC is powered by V_{CC}, it can operate in low power consumption states. Some MSWC operations depend on the presence of a clock; these functions are not available when the core clock is turned off.

Figure 56 shows the block diagram of the MSWC.

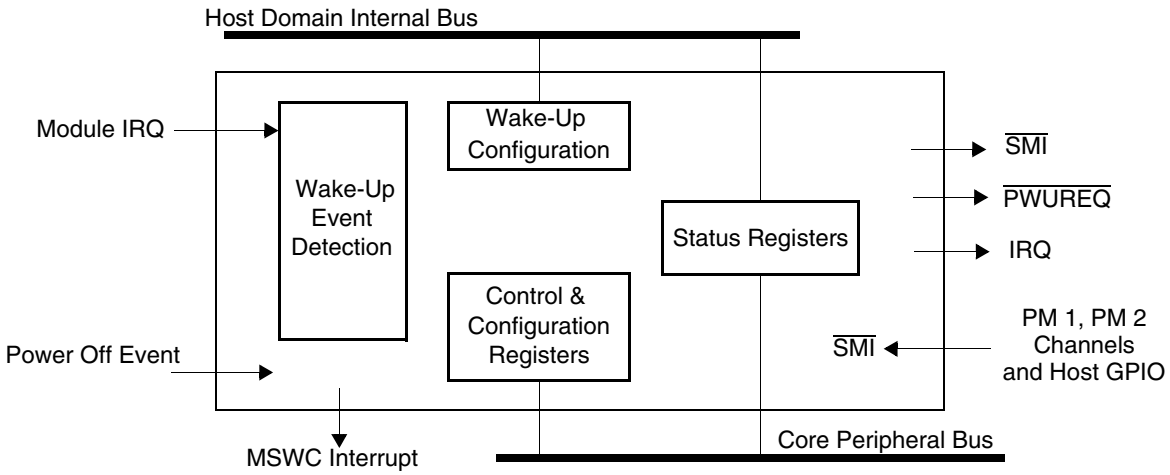


Figure 56. MSWC Block Diagram

5.5.1 Features

The MSWC recognizes the following maskable system events:

- Wake-up on module IRQs from: KBD and Mouse interface, PM Channel 1 and 2.
- Software events
  - — Software-Triggered wake-up event
  - — ACPI power state change indications
  - — Software off command

When any of the above events occurs, the MSWC notifies the host and/or core by asserting one or more of the following:

- Power-Up Request ($\overline{PWUREQ}$) signal
- System Management Interrupt ($\overline{SMI}$) signal
- Interrupt to the host (IRQ)
- MSWC interrupt to the core

5.5.2 Wake-Up Event Detection and Status Bits

The MSWC monitors various system signals for a wake-up event. When an event is detected, a status bit is set to record it. Each event goes to the Wake-Up Mode Control Logic, which determines its effect; see Figure 59 for an illustration of this mechanism. A set of dedicated registers is used to determine the wake-up criteria, including the WOC detection.

The following wake-up input events are detected by the MSWC:

- Software event
- IRQ from KBD and Mouse interface, PM Channel 1 and 2 modules
- ACPI state change
- Legacy off event

When an input event is detected, the corresponding status bit in both host and core status registers is set to 1, regardless of any Routing Enable bit setting. If both the status bit and a Routing Enable bit corresponding to a specific event are set to 1 (no matter in what order), the output pin corresponding to that Routing Enable bit is asserted.

A status bit is cleared by writing 1 to it. Writing 0 to a status bit does not change its value. Clearing the Routing Enable bit of an event prevents the event from issuing the corresponding system notification (output event) but does not affect the status bit. Figure 59 shows the routing mechanism of detected wake-up events to the various means of system notification (i.e., output events).

Both the core and the host have status registers; thus both core and host software can monitor the various event status bits. This enables the handling of events via wake-up logic that is implemented as part of the EC firmware, and passing the wake-up notifications through the Power Management host-interface protocol. The core uses a mask register (WK_SMIENn) to define which of the status bits it should respond to.

5.0 Host-to-EC Controller Interface Modules (Continued)

It is recommended that each of the wake-up sources be handled by one handler routine on the host (i.e., SMI, SCI or IRQ triggered) or the core.

Software Event

A software event may be used to trigger an interrupt to the host and/or core via software control, as shown in Figures 57 and 58.

A software event to the host is active when Software Event Status bit in WK_STS0 register is set. When that status bit is set, WK_EN0 bit 6 enables the generation of an interrupt to the host.

A software event to the core is active when the Software Event Status bit in MSHES0 register is set. When that status bit is set, Bit 6 in MSHEIE0 register enables generation of an interrupt to the core.

The software events are activated (i.e., the status bits in WK_STS0 and MSHES0 registers are set) by writing 1 to the Software Event Status bit in WK_STS0 register when that bit is cleared (the Software Event Status bit in MSHESO is set by a change of the respective bit in WK_STS0, from 0 to 1). The host can activate the software event when V_{DD} is present. The core can activate the software event by accessing the MSWC host registers through the “Core Access to Host Modules” bridge (even when V_{DD} is off).

The software event clearing mechanism is defined by HSECM bit in MSWCTL1 register, as follows:
When HSECM bit is cleared, the host Software Event Status bit in WK_STS0 register is cleared by writing 1 to it when it is set (i.e., writing 1 to Host Software Status bit in WK_STS0 register functions as a toggle operation). The core Software Event Status bit in MSHES0 is cleared by writing 1 to it (write 1 to clear). This mode is useful when the software event interrupts the host and is handled by it (Figure 57).

When HSECM bit is set, host Software Event Status bit in WK_STS0 register and the core Software Event Status bit in MSHES0 are both cleared by writing 1 to the core Software Event Status bit (MSHES0 register). This is useful when the software event is used to interrupt the core and is handled by it (Figure 58).

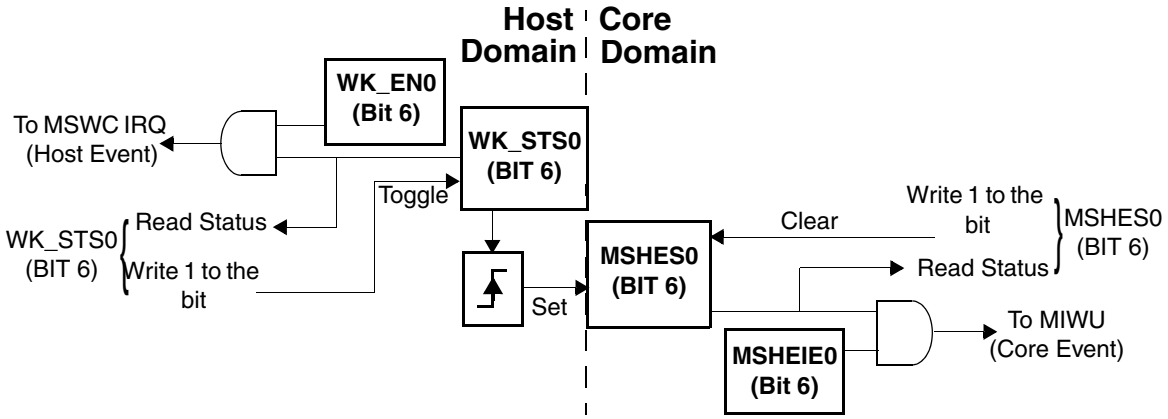


Figure 57. MSWC Software Event Generation for HSECM Bit (MSWCTL1 Register) = 0

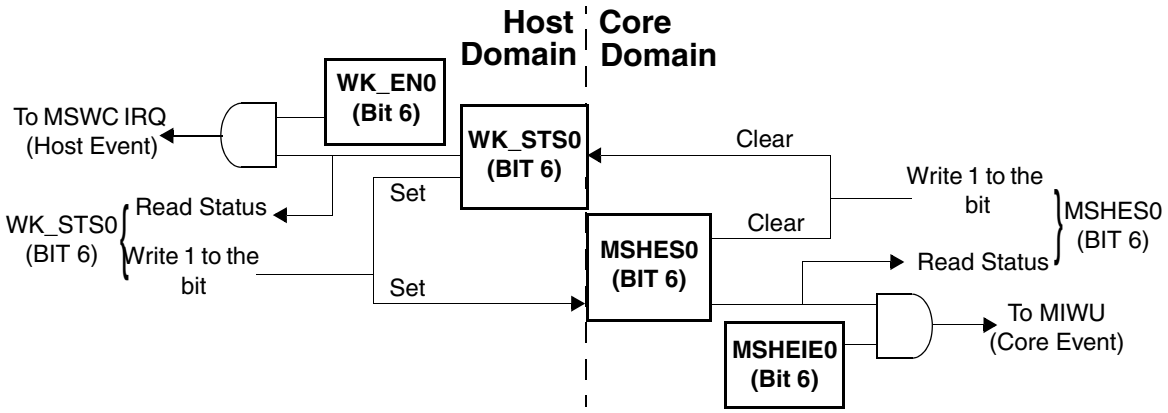


Figure 58. MSWC Software Event Generation for HSECM Bit (MSWCTL1 Register) = 1

Module IRQ Event

A module IRQ wake-up event is defined as the leading edge of the IRQ assertion of any of the following logical devices: KBD and Mouse interface, PM Channel 1 and 2.

To enable the IRQ of a specific logical device to trigger a wake-up event, the associated enable bit must be set to 1. This is bit 4 of the Interrupt Number and Wake-Up on IRQ Enable register, located at index 70h in the configuration space of the logical device; see Table 34 on page 249. When this bit is set, any IRQ assertion of the corresponding logical device activates the module IRQ wake-up event. Therefore, the module IRQ wake-up event is a combination of all IRQ signals of the logical devices for which wake-up on IRQ is enabled.

When an event is detected as active, its associated status bit (bit 7 of WK0_STS register) is set to 1. If the associated enable bit (bit 7 of WK_EN0 register) is also set to 1, the $\overline{\text{PWUREQ}}$ output is asserted and remains asserted until the status bit is cleared.

Since V_{DD} powers IRQ generation of the logical devices, a module IRQ event can be activated only when V_{DD} is on; see Section 6.1 on page 246 for a list of logical devices.

ACPI State Change and Legacy Off Events

The host may operate in either Legacy or ACPI mode. The operation mode is specified by Power Button Mode bit in SuperI/O Configuration D register (SIOCFD). When EICFGPBM bit in MSIEN2 register is set, a change to the Power Button Mode bit generates an interrupt to the core. The core may read the value of the Power Button Mode bit, using CFGPBM bit in MSWCTL2 register, to determine how to interpret the other power state request bits.

Power Supply Off bit in SIOCFD register may be used in Legacy mode to indicate a request to turn power off. A write of 1 to this bit sets CFGPSO bit in MSWCTL2 register. Then, if EICFGPSO bit in MSIEN2 register is set, an interrupt to the core is generated, indicating the event.

A set of System State Change Request bits (S1-S5) is provided in WK_STATE register. The host uses these bits for ACPI-compliant state change requests. A write of 1 to any of these bits indicates a state change request to the core through the respective bit in MSWCTL2 register. When all bits in WK_STATE are written with 0, a request of S0 is indicated and ACPIS0 bit in MSWCTL2 register is set. When any S0-S5 bit in MSWCTL2 is set and the respective mask bit in MSIEN2 register is set, an interrupt to the core is generated when a change to any of the state bits is detected.

All interrupt requests may be cleared by writing 1 to the corresponding status bit or by masking the event (by clearing the corresponding Interrupt Enable bit).

5.5.3 Wake-Up Output Events

The MSWC generates four types of output events:

- IRQ - an interrupt routed as configured in the MSWC PnP configuration registers.
- $\overline{\text{PWUREQ}}$ - an event that is typically connected to an input in the chipset that triggers an SCI event.
- $\overline{\text{SMI}}$ - an event typically connected to an input in the chipset that triggers an SMI event.
- MSWCI - an interrupt to the MIWU in the core domain. This enables the core firmware to handle wake-up events.

Figure 59 shows the enabling mechanism and the event generation mechanism for the various output events. Output events to the host are generated for input events that have their status bit set (WK_STSn.i is 1). Output events to the core, through the MIWU, are generated for input events that have their core status bit set (MSHESn.i is 1).

Each of the three host Wake-Up Event Routing Control registers (WK_ENn, WK_SMIENn and WK_IRQENn) holds a Routing Enable bit for each event. This allows selective routing of these events to $\overline{\text{PWUREQ}}$, $\overline{\text{SMI}}$ and/or the assigned MSWC interrupt request (IRQ) channel, respectively.

After an output event is asserted, it is active until all set status bits are cleared or masked. The current status of the event may be read from the ACPI status registers in the chipset's ACPI controller or by reading Wake-Up Event Status Register 0 (WK_STS0) (see page 237) and Wake-Up Signals Value Register (WK_SIGV); see page 238.

For $\overline{\text{SMI}}$ output events, the MSWC combines the event request coming from the Power Management Channels 1 and 2 and from the Host GPIO module, with MSWC internal $\overline{\text{SMI}}$ events.

The $\overline{\text{SMI}}$ may be output from the WPC8763L using the dedicated $\overline{\text{SMI}}$ signal or by routing $\overline{\text{SMI}}$ to an interrupt request channel via the device's configuration registers.

The Wake-Up Event Routing Control register, MSHEIEn, which is controlled by the core, holds an enable bit for each of the events. This allows selective routing of these events to the core wake-up interrupt (MSWCI) to the MIWU. The core event is controlled using a separate set of status signals to prevent race conditions when clearing events.

The MSWCI interrupt is a level high interrupt that gathers requests from MSHESn, MSWCTL2 and MSWCTL3 registers. Once an output event is asserted, it keeps its active state until all set status bits are cleared or masked. This interrupt signal is connected to the MSWC wake-up input of the MIWU, which enables handling state change requests even in Idle mode. The MSWC output for this input is connected to the core through the MIWU, enabling a power state change on interrupt.

5.0 Host-to-EC Controller Interface Modules (Continued)

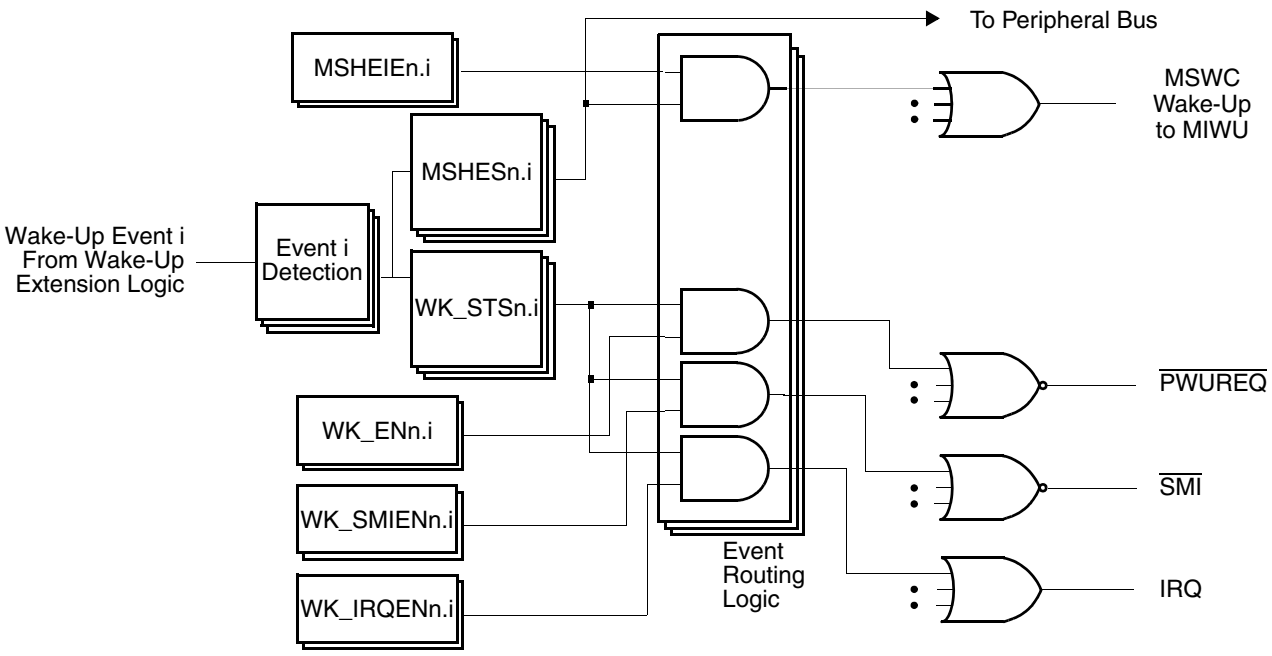


Figure 59. Wake-Up Event Routing Mechanism

5.5.4 Other MSWC Controlled Elements

In addition to its Power Management functions, the MSWC controls the handling of the following system control elements:

- Host Configuration Address Selection
- Host Keyboard Reset Fast Reset Output ($\overline{\text{KBRST}}$)
- GA20 Pin Functionality
- Host Power-On Indication

Host Configuration Address Selection

The standard strap configuration enables the selection of one of two SuperI/O configuration register addresses. When the WPC8763L is enabled in Programmable Configuration Address mode (see Table 29 on page 246), the core can set the address of the SuperI/O configuration index/data registers.

HCBAL and HCBAH are byte-wide read/write registers. HCBAL holds the least significant byte of a host motherboard PnP initial configuration address; HCBAH holds the most significant byte. The contents of HCBAH and HCBAL are set to their default values by V_{CC} Power-Up reset and $\overline{\text{VCC\_POR}}$ Input reset.

To update the base address of the SuperI/O configuration index/data registers:

1. Clear VHCFGA bit in MSWCTL1 register by writing 1 to it.
2. Write the lower byte of the address to HCBAL (LSB must be written 0).
3. Write the higher byte of the address to HCBAH.
4. Set HCFGLK bit to prevent an accidental change of the address written to HCBAL and HCBAH.

The base address is preserved by V_{CC}, and VHCFGA remains set as long as a valid address is maintained. If there is no valid configuration base address, the LPC interface does not respond to configuration requests.

Important Note: If Programmable Configuration Address mode is selected, the host may access the configuration registers only **after** the HCBAL and HCBAH registers are initialized by the core.

5.0 Host-to-EC Controller Interface Modules (Continued)

Host Keyboard Fast Reset

The Host Keyboard Reset output ($\overline{KBRST}$) is an output of the WPC8763L that serves as one of the sources for Host Soft reset commands (i.e., $\overline{INIT}$ input in the x86 processors). Figure 60 shows the $\overline{KBRST}$ generation mechanism. The host is reset when the $\overline{KBRST}$ output is low. A reset command is issued by the WPC8763L by software or hardware, as follows:

- Software: The core firmware can issue a reset command to the host by writing 1 to HRSTOB in MSWCTL1 register. The reset to the host ends by writing 0 to HRSTOB.
- Hardware: When $\overline{LPCPD}$ signal is low, $\overline{KBRST}$ is forced low if LPFTO bit in MSWCTL3 register is set to 1.
- Hardware: The host is reset during V_{CC} Power-Up reset if HRAPU bit in MSWCTL3 register is set and an LPC transaction is started. This prevents accesses to the WPC8763L from being ignored due to the duration of the V_{CC} Power-Up reset.

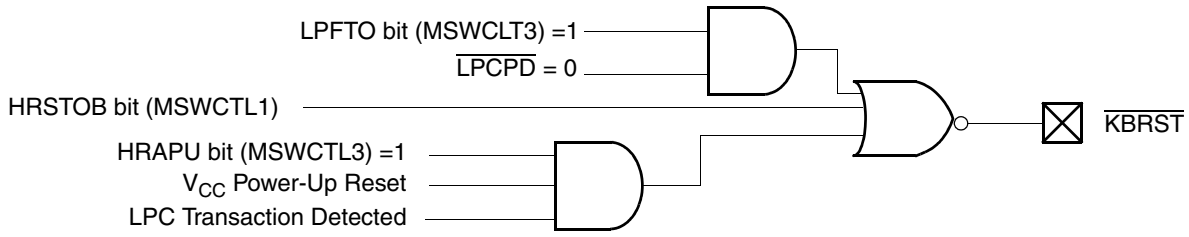


Figure 60. $\overline{KBRST}$ Generation Mechanism

In addition, $\overline{KBRST}$ is held low when the V_{DD} power supply is off.

GA20 Pin Functionality

The GA20 (Gate Address A20) function is part of the PC architecture. In the WPC8763L, the GA20 function is controlled by GA20B bit in MSWCTL1 register. After reset, the default state of GA20 signal is high level. The firmware running on the core can change the GA20 signal state by modifying the GA20B bit.

When $\overline{LPCPD}$ signal is low, if LPFTO bit in MSWCTL3 register is set to 1, the GA20 signal is forced to low level (this indicates that the LPC bus is powered down).

In addition, GA20 is held low when the V_{DD} power supply is off.

5.5.5 MSWC Host Registers

The MSWC registers are organized in four banks, all of which are battery-backed. The offsets are related to a base address that is determined by the MSWC Base Address register in the device configuration registers. The lower 19 offsets (00h-12h) are common to all four banks; the upper offsets (13h-1Fh) are divided as follows:

- Bank 0 is reserved.
- Bank 1 is reserved.
- Bank 2 holds the Event Routing Configuration registers.
- Bank 3 is reserved.

The active bank is selected through Configuration Bank Select field (bits 1-0) in Wake-Up Configuration register (WK_CFG).

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

MSWC Host Register Map

The following tables list the MSWC host registers. For the MSWC core register map, see Section 5.5.6 on page 241.

Table 26. Banks 0, 1, 2 and 3 - The Common Control and Status Register Map

Offset	Mnemonic	Register Name	Type
00h	WK_STS0	Wake-Up Event Status 0	R/W1C
02h	WK_EN0	Wake-Up Enable 0	R/W
04h	WK_CFG	Wake-Up Configuration	R/W
06h	WK_SIGV	Wake-Up Signal Value	RO
07h	WK_STATE	Wake-Up ACPI State	WO
08h-12h	Reserved		

5.0 Host-to-EC Controller Interface Modules (Continued)

Table 27. Bank 2 - Event Routing Configuration Register Map

Offset	Mnemonic	Register Name	Type
13h	WK_SMIEN0	Wake-Up SMI Enable 0	R/W
14h	Reserved		
15h	WK_IRQEN0	Wake-Up Interrupt Request Enable 0	R/W
16h-1Fh	Reserved		

Wake-Up Event Status Register 0 (WK_STS0)

WK_STS0 is set to 00h on V_{CC} Power-Up reset. Writing 1 to a bit clears it to 0. Writing 0 has no effect. Bit 6 behaves in a special way, as described in the table below.

Location: Offset 00h

Type: R/W1C

Bit	7	6	5	4	3	2	1	0
Name	**Module IRQ Event Status**	**Software Event Status**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Module IRQ Event Status.** 0: Event not active (default). 1: Event active.
6	**Software Event Status.** This bit may work in two modes, as defined by HSECM bit in MSWCTL1 register; see "MSWC Control Status Register 1 (MSWCTL1)" on page 241. When HSECM is 0, writing 1 to Software Event Status bit inverts its value. When HSECM is 1, writing 1 to Software Event Status bit sets it; the bit is cleared by a write of 1 to bit 6 in MSHES0 register. 0: Event not active (default). 1: Event active.
5-0	**Reserved.**

Wake-Up Events Enable Register (WK_EN0)

WK_EN0 is set to 00h on V_{CC} Power-Up reset. When enabled, and if wake-up events are detected, the $\overline{\text{PWUREQ}}$ signal is activated.

Location: Offset 02h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Module IRQ Event Enable**	**Software Event Enable**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Module IRQ Event Enable.** 0: Disabled (default). 1: Enabled.

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Description
6	**Software Event Enable.** 0: Disabled (default). 1: Enabled.
5-0	**Reserved.**

Wake-Up Configuration Register (WK_CFG)

WK_CFG is set to the default value on V_{CC} Power-Up reset.

Location: Offset 04h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**Configuration Bank Select**	
Reset	0	0	1	0	0	0	0	0
Required	0	0						

Bit	Type	Description
7-2		**Reserved.**
1-0	R/W	**Configuration Bank Select.** **Bits 1 0 — Bank — Register** 0 0: 0 Reserved (default). 0 1: 1 Reserved. 1 0: 2 Event Routing. 1 1: 3 Reserved.

Wake-Up Signals Value Register (WK_SIGV)

WK_SIGV returns the value of $\overline{SMI}$ and $\overline{PWUREQ}$ signal outputs and inputs to the MSWC module. This register serves to identify the source of the wake-up request when multiple sources are enabled.

Location: Offset 06h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Host GPIO SMI Value**	**Reserved**		**PWUREQ Output Value**	**PM2 SMI Value**	**PM1 SMI Value**	**MSWC SMI Value**	**SMI Output Value**

Bit	Description
7	**Host GPIO SMI Value.** 0: $\overline{SMI}$ output of the Host GPIO module is low (asserted). 1: $\overline{SMI}$ output of the Host GPIO module is high (deasserted).
6-5	**Reserved.**
4	**PWUREQ Wake-Up Output Value.** 0: $\overline{PWUREQ}$ output is low (asserted). 1: $\overline{PWUREQ}$ output is high (deasserted).
3	**PM2 SMI Value.** 0: $\overline{SMI}$ output of the Power Management Channel 2 is low (asserted). 1: $\overline{SMI}$ output of the Power Management Channel 2 is high (deasserted).

2	**PM1 SMI Value.** 0: $\overline{\text{SMI}}$ output of the Power Management Channel 1 is low (asserted). 1: $\overline{\text{SMI}}$ output of the Power Management Channel 1 is high (deasserted).
1	**MSWC SMI Value.** 0: $\overline{\text{SMI}}$ output of the MSWC module is low (asserted). 1: $\overline{\text{SMI}}$ output of the MSWC module is high (deasserted).
0	**SMI Output Value.** 0: $\overline{\text{SMI}}$ output is low (asserted). 1: $\overline{\text{SMI}}$ output is high (deasserted).

Wake-Up ACPI State Register (WK_STATE)

WK_STATE is write-only. It always returns 00h when read.

Location: Offset 07h

Type: WO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**		**S5**	**S4**	**S3**	**S2**	**S1**	**Reserved**

Bit	Description
7-6	**Reserved.**
5	**S5 (Request to Change to S5 State).** A write of 1 indicates to the core that the host requests to change to S5 state. The S state and transition are interpreted as specified in the ACPI standard for S state change requests. This bit always reads 0. 0: Not an S5 state request. 1: S5 state setting request.
4	**S4 (Request to Change to S4 State).** A write of 1 indicates to the core that the host requests to change to S4 state. The S state and transition are interpreted as specified in the ACPI standard for S state change requests. This bit always reads 0. 0: Not an S4 state request. 1: S4 state setting request.
3	**S3 (Request to Change to S3 State).** A write of 1 indicates to the core that the host requests to change to S3 state. The S state and transition are interpreted as specified in the ACPI standard for S state change requests. This bit always reads 0. 0: Not an S3 state request. 1: S3 state setting request.
2	**S2 (Request to Change to S2 State).** A write of 1 indicates to the core that the host requests to change to S2 state. The S state and transition are interpreted as specified in the ACPI standard for S state change requests This bit always reads 0. 0: Not an S2 state request. 1: S2 state setting request.
1	**S1 (Request to Change to S1 State).** A write of 1 indicates to the core that the host requests to change to S1 state. The S state and transition are interpreted as specified in the ACPI standard for S state change requests. This bit always reads 0. 0: Not an S1 state request. 1: S1 state setting request.
0	**Reserved.**

5.0 Host-to-EC Controller Interface Modules (Continued)

Wake-Up Event Routing to $\overline{SMI}$ Enable Register 0 (WK_SMIEN0)

WK_SMIEN0 is set to 00h on V_{CC} Power-Up reset. It controls the routing of detected wake-up events to the $\overline{SMI}$ signal. Enabled wake-up events activate the $\overline{SMI}$ signal regardless of the value of WK_EN0 register.

Location: Bank 2, Offset 13h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**Software Event to $\overline{SMI}$ Enable**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Reserved.**
6	**Software Event to SMI Enable.** 0: Disabled (default). 1: Enabled.
5-0	**Reserved.**

Wake-Up Event Routing to IRQ Enable Register 0 (WK_IRQEN0)

WK_IRQEN0 is set to 00h on V_{CC} Power-Up reset. It controls the routing of detected wake-up events to the assigned MSWC interrupt request (IRQ) channel. Enabled wake-up events activate the assigned IRQ channel regardless of the value of WK_EN0 register.

Location: Bank 2, Offset 15h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**Software Event to IRQ Enable**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Reserved.**
6	**Software Event to IRQ Enable.** 0: Disabled (default). 1: Enabled.
5-0	**Reserved.**

5.0 Host-to-EC Controller Interface Modules (Continued)

5.5.6 MSWC Core Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Table 28. MSWC Core Register Map

Location	Mnemonic	Register Name	Size	Type
FF F040h	MSWCTL1	MSWC Control Status Register 1	Byte	Varies per bit
FF F042h	MSWCTL2	MSWC Control Status Register 2	Byte	Varies per bit
FF F044h	MSWCTL3	MSWC Control Status Register 3	Byte	Varies per bit
FF F048h	HCBAL	Host Configuration Base Address Low	Byte	R/W or RO
FF F04Ah	HCBAH	Host Configuration Base Address High	Byte	R/W or RO
FF F04Ch	MSIEN2	MSWC Interrupt Enable Register 2	Byte	R/W
FF F04Eh	MSHES0	MSWC Host Event Status Register 0	Byte	R/W1C
FF F050h	MSHEIE0	MSWC Host Event Interrupt Enable Register	Byte	R/W

MSWC Control Status Register 1 (MSWCTL1)

The contents of MSWCTL1 are preserved by V_{CC}. The bits are set to their default values by Core Domain reset unless specifically stated otherwise.

Location: FF F040h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**GA20B**	**Reserved**	**HSECM**	**HCFGLK**	**VHCFGA**	**LPCRSTA**	**HPWRON**	**HRSTOB**
Reset	1	0	0	0	0	X	X	0

Bit	Type	Description
7	R/W	**GA20B (Gate A20 Out Bit).** Enables the WPC8763L to control the GA20 pin. 0: GA20 is set to low level. 1: GA20 is set to high level (default).
6		**Reserved.**
5	R/W	**HSECM (Host Software Event Clear Mode).** Controls the Clear mode of Host Software Event Status bit in WK_STS0. 0: Host Software Event Status bit in WK_STS0 (bit 6) toggles on host writes of 1. MSHES0 bit 6 is set when WK_STS0 bit 6 changes from 0 to 1 (default). 1: Host Software Event Status bit in WK_STS0 (bit 6) and MSHES0 bit 6 are both cleared by writes of 1 to MSHES0 register.
4	R/W1S	**HCFGLK (Host Configuration Address Lock).** When 1 is written, HCFGLK changes to read-only (i.e., it cannot be cleared by the firmware) and locks VHCFGA bit, HRAPU bit in MSWCTL3 register, HCBAH register and HCBAL register, preventing their accidental change. 0: Allows update of the Host configuration registers base address (default). 1: Locks the Host configuration registers base address.
3	R/W1C or RO	**VHCFGA (Valid Host Configuration Address).** Is set by a write to HCBAH register, as detailed in the update sequence in "Host Configuration Address Selection" on page 235. The firmware can clear the bit by writing 1 to it. Thus HCBAH and HCBAL registers are used as the address of the configuration registers when the WPC8763L is set to operate with the internal base address. Writing 0 to this bit is ignored. It can be locked and made read-only by setting HCFGLK (bit 4). Cleared by V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset. 0: Host configuration registers base address is not valid, and access to this registers by the host is not enabled (default). 1: Host configuration registers base address is specified in HCBAH and HCBAL registers.

5.0 Host-to-EC Controller Interface Modules (Continued)

Bit	Type	Description
2	RO	**LPCRSTA (LPC Reset Active).** The $\overline{\text{LRESET}}$ input is active (low). LPCRSTA bit is connected to MIWU input WUI57 (see Table 10 on page 66). 0: $\overline{\text{LRESET}}$ is not active (high). 1: $\overline{\text{LRESET}}$ is active (low).
1	RO	**HPWRON (Host Power On).** The V_{DD} power detection logic indicates that V_{DD} is on. 0: V_{DD} is off (below V_{DDON}). 1: V_{DD} is on (above V_{DDON}).
0	R/W	**HRSTOB (Host Reset Out Bit).** Enables the WPC8763L to generate a Host Soft reset via firmware, using the $\overline{\text{KBRST}}$ pin. The pin is held low (reset is active) as long as this bit 1. 0: $\overline{\text{KBRST}}$ is not forced active (default). 1: Force $\overline{\text{KBRST}}$ active (low).

MSWC Control Status Register 2 (MSWCTL2)

Bits in MSWCTL2 are cleared by Core Domain reset unless specifically stated otherwise.

Location: FF F042h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**CFGPSO**	**CFGPBM**	**ACPIS5**	**ACPIS4**	**ACPIS3**	**ACPIS2**	**ACPIS1**	**ACPIS0**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7	R/W1C	**CFGPSO (SuperI/O Configuration Register D Power Supply Off).** Is set when a 1 is written to the Power Supply Off bit in SIOCFD register. This bit may be used by the host software to specify to the core that the power supply should be turned off in a non-ACPI system; see "SuperI/O Configuration D Register (SIOCFD)" on page 256. A write of 1 clears this bit and the interrupt signal generated when this bit is set. A write of 0 is ignored.
6	RO	**CFGPBM (SuperI/O Configuration Register D Power Button Mode).** Reflects the current status of the Power Button Mode bit in SIOCFD register. This bit may be used by the host software to specify to the core the method used for power-off signaling; see "SuperI/O Configuration D Register (SIOCFD)" on page 256. A write of 1 clears the interrupt signal caused by a change in this bit value. A write of 0 is ignored. This bit is cleared only by V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset.
5-1	R/W1C	**ACPIS5-1 (ACPI Request for S5 through S1).** Can be used by ACPI software to directly request a change of power state. A bit is set by a host software write of 1 to the respective bit in WK_STATE register. A bit is cleared by writing 1 to it. A write of 0 is ignored. When any ACPIS5-1 bit is set, an MSWC wake-up interrupt to the core is asserted (via a MIWU input).
0	R/W1C	**ACPIS0 (ACPI request for S0).** May be used by ACPI software to directly request a change of power state. This bit is set when the host software writes a value of 0 to bits S1 through S5 in WK_STATE register. This bit is cleared by writing 1 to it. A write of 0 is ignored. When ACPIS0 is set, an MSWC wake-up interrupt to the core is asserted (via a MIWU input). 0: No pending request for S0 change (default). 1: A request for S0 change was detected.

5.0 Host-to-EC Controller Interface Modules (Continued)

MSWC Control Status Register 3 (MSWCTL3)

MSWCTL3 is reset only on V_{CC} Power-Up reset.

Location: FF F044h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**LPFTO**	**HRAPU**
Reset	0	0	0	0	1	0	0	1

Bit	Type	Description
7-2		**Reserved.**
1	R/W	**LPFTO (LPC Power Fail Turn Off $\overline{\text{KBRST}}$ and GA20).** Selects the handling of $\overline{\text{KBRST}}$ and GA20 outputs when $\overline{\text{LPCPD}}$ is active. 0: Ignore $\overline{\text{LPCPD}}$ in handling these signals (default). 1: Force $\overline{\text{KBRST}}$ and GA20 low if $\overline{\text{LPCPD}}$ is active.
0	R/W or RO	**HRAPU (Host Reset when WPC8763L Accessed During V_{CC} Power-Up Reset).** Generates a $\overline{\text{KBRST}}$ reset to the host if LPC activity is detected before the V_{CC} Power-Up reset is completed. This restarts any LPC transaction that may have addressed the WPC8763L but could not be handled correctly. When HCFGLK bit is set, writes to this bit are ignored. 0: Do not generate a reset on LPC transactions if the WPC8763L is in V_{CC} Power-Up reset. 1: Assert $\overline{\text{KBRST}}$ output on LPC transactions if the WPC8763L is in V_{CC} Power-Up reset (default).

Host Configuration Base Address Low (HCBAL)

HCBAL holds the lower byte of the Host configuration registers base address. Bit 0 of this register is always forced to 0 to guarantee address alignment. This register is set to its default value by V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset.

Location: FF F048h

Type: R/W (Except for bit 0, which is read-only and has a value of 0) or RO

Bit	7	6	5	4	3	2	1	0
Name	**Host Configuration Registers Base Address Low**							
Reset	0	0	0	0	0	0	0	**0**

Host Configuration Base Address High (HCBAH)

HCBAH holds the higher byte of the Host configuration registers base address. This register is set to its default value by V_{CC} Power-Up reset or $\overline{\text{VCC\_POR}}$ Input reset.

Location: FF F04Ah

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**Host Configuration Registers Base Address High**							
Reset	0	0	0	0	0	0	0	0

5.0 Host-to-EC Controller Interface Modules (Continued)

MSWC Interrupt Enable Register 2 (MSIEN2)

MSIEN2 holds enable bits for interrupt generation to the core through the MIWU (level high) for the respective bits in MSWCTL2 and MSWCTL3 registers. The interrupt may be cleared by clearing the status bit or by masking the interrupt. On Core Domain reset, this register is cleared (00h).

Location: FF F04Ch

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**EICFGPSO**	**EICFGPBM**	**EIACPIS5**	**EIACPIS4**	**EIACPIS3**	**EIACPIS2**	**EIACPIS1**	**Reserved**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**EICFGPSO (Enable Interrupt SuperI/O Configuration Register D Power Supply Off).** Enables interrupt generation to the core when CFGPSO bit in MSWCTL2 register is set. 0: Interrupt disabled (default). 1: Generate a level high interrupt when the CFGPSO bit is set to 1.
6	**EICFGPBM (Enable Interrupt SuperI/O Configuration Register D Power Button Mode).** Enables interrupt generation to the core when CFGPBM bit in MSWCTL2 register changes. 0: Interrupt disabled (default). 1: Generate a level high interrupt on any change to the CFGPBM bit.
5-1	**EIACPIS5-1 (Enable Interrupt ACPI request for S5 through S1).** Enables interrupt generation to the core when a ACPIS5-1 bit in MSWCTL2 register changes. An interrupt enable for ACPIS0 is enabled when any of these bits is set. 0: Interrupt disabled (default). 1: Generate a level high interrupt on any change to the ACPIS5-1 bits.
0	**Reserved.**

MSWC Host Event Status Register 0 (MSHES0)

MSHES0 holds information similar to that in WK_STS0 register. The same event that causes a WK_STS0 bit to be set sets the respective bit in MSHES0 register. Clearing bits is done for each of the status registers separately. MSHES0 is reset to 00h on V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset. Writing 1 to a bit clears it to 0. Writing 0 has no effect. Bit 6 of this register behaves in a special way on set and clear, as described below.

Location: FF F04Eh

Type: R/W1C

Bit	7	6	5	4	3	2	1	0
Name	**Module IRQ Event Status**	**Software Event Status**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Module IRQ Event Status.** 0: Event not active (default). 1: Event active.
6	**Software Event Status.** Indicates a host software event. It may operate in two modes depending on HSECM bit in MSWCTL1 register. When HSECM is cleared, this bit is set when bit 6 of WK_STS0 changes from 0 to 1. When HSECM is set, this bit is set on a write of 1 to WK_STS0 bit 1. This bit is cleared by writing 1 to it. 0: Event not active (default). 1: Event active.
5-0	**Reserved.**

5.0 Host-to-EC Controller Interface Modules (Continued)

MSWC Host Event Interrupt Enable Register (MSHEIE0)

MSHEIE0 is cleared to 00h on Core Domain reset. It enables a core interrupt through the MIWU (level high) for the respective bit in the MSHES0 register. The interrupt may be cleared by clearing the status bit or by masking the interrupt.

Location: FF F050h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Module IRQ Event Enable**	**Software Event Enable**	**Reserved**					
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Module IRQ Event Enable.** 0: Disabled (default). 1: Enabled.
6	**Software Event Enable.** 0: Disabled (default). 1: Enabled.
5-0	**Reserved.**

5.5.7 Usage Hints

$\overline{\text{PWUREQ}}$ Output Connection

The $\overline{\text{PWUREQ}}$ concentrates a set of wake-up and other power management events in the WPC8763L. In typical use, this signal is connected to one of the chipset inputs that drive the SCI event to the host.

$\overline{\text{LRESET}}$ Events

$\overline{\text{LRESET}}$ is used to reset the host domain. Therefore, some of the MSWC functions may need to be set to their default values when a falling edge of $\overline{\text{LRESET}}$ is detected. An interrupt routine triggered by this event may be used for this task. The following functions should be reset:

- GA20.
- $\overline{\text{KBRST}}$.
- Host Registers: WK_EN0, WK_SMIEN0 and WK_IRWEN0;
  In this case, use the “Core Access to Host Modules”; see Section 5.3 on page 211.

6.0 Host Modules and Host Interface

6.1 DEVICE ARCHITECTURE AND CONFIGURATION

The WPC8763L host functions consist of a collection of generic and proprietary functional blocks. This section describes the WPC8763L host structure and provides all logical device-specific information, including special implementation of generic blocks, system interface and device configuration. Some parameters in the implementation of the functional blocks may vary per function and/or device.

The WPC8763L host functions consist of logical devices, the host interface and a central set of configuration registers, all built around a central internal bus.

The LPC Bus interface serves as a bridge between the external LPC interface and the internal bus. It supports the following operations, as defined in Intel's *LPC Interface Specification, Revision 1.1*:

- 8-bit I/O read
- 8-bit I/O write
- 8-bit Memory read
- 8-bit Memory write
- 8-bit and 32-bit Firmware Memory (FWH) read
- 8-bit Firmware Memory (FWH) write

The Configuration and Control register set supports ACPI-compliant PnP configuration. The configuration registers are structured as a subset of the Plug and Play Standard registers, defined in Appendix A of the *Plug and Play ISA Specification, Revision 1.1* by Intel and Microsoft, and are similar to those used in Winbond SuperI/O devices. All system resources assigned to the functional blocks (I/O address space and IRQ lines) are configured in and managed by this register set. In addition, some function-specific parameters are configurable through the configuration registers and distributed to the functional blocks through special control signals.

6.1.1 Configuration Structure and Access

The configuration structure consists of a set of banked registers that are accessed via a pair of specialized registers.

The Index-Data Register Pair

Access to the host functions configuration registers is via an index-data register pair, using two system I/O byte locations. The base address of this register pair is determined during V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset according to the state of the hardware strapping option on the BADDR1-0 pins. Table 29 shows the selected base addresses as a function of BADDR1-0; see Section 2.4.15 on page 30.

Table 29. BADDR1-0 Strapping Options

BADDR1-0	I/O Address	
	Index Register	Data Register
1 1	164Eh	164Fh
1 0	2Eh	2Fh
0 1[1]	(HCBAH,HCBAL)	(HCBAH,HCBAL)+1
0 0	XOR-Tree Test Mode	

1. See "Host Configuration Address Selection" on page 235 for more details about this option.

The index register is an 8-bit read/write register located at the selected base address (Base+0). It is used as a pointer to the configuration register file and holds the index of the configuration register that is currently accessible via the data register. Reading the index register returns the last value written to it (or a default of 00h after reset).

The data register is an 8-bit register located at the selected base address (Base+1). Accessing the data register actually accesses the configuration register that the index register is currently pointing to.

6.0 Host Modules and Host Interface (Continued)

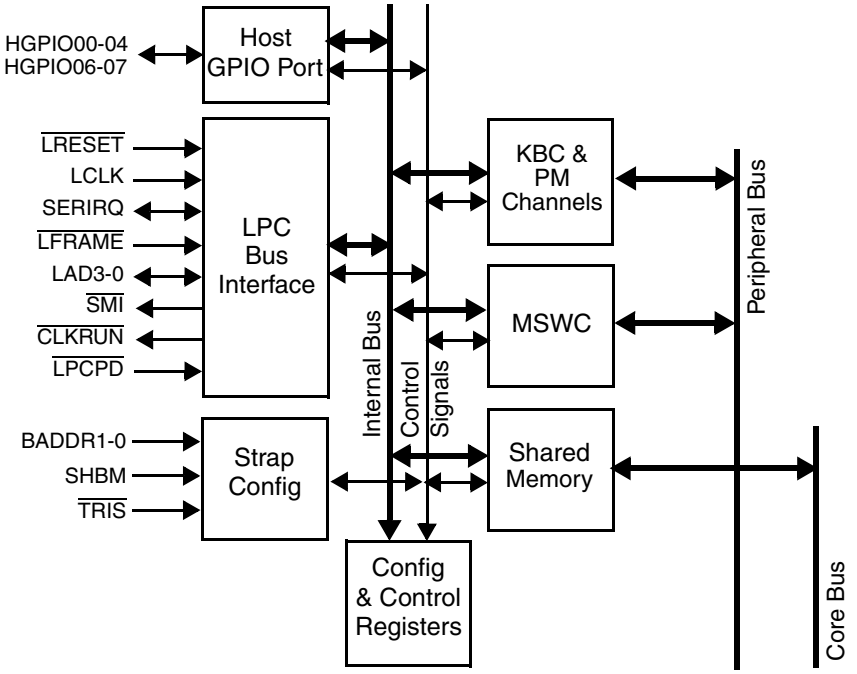


Figure 61. Host Domain Detailed Block Diagram

Banked Logical Device Registers Structure

Each functional block is associated with a Logical Device Number (LDN). The configuration registers are grouped in banks, where each bank holds the standard configuration registers of the corresponding logical device. Table 30 shows the LDN values of the WPC8763L functional blocks. Any value not listed is reserved.

Figure 62 shows the structure of the standard configuration register file. The host function control and configuration registers are not banked and are accessed by the index-data register pair only, as described above. However, the device control and device configuration registers are replicated over the banks of each logical device. Therefore, two-dimensional indexing is used to access a specific register in a specific bank: the LDN register selects the bank (or logical device) and the index register selects the register within the bank. Accessing the data register while the index register holds a value of 30h or higher actually accesses the logical device configuration registers, currently pointed to by the index register, within the logical device currently selected by the LDN register.

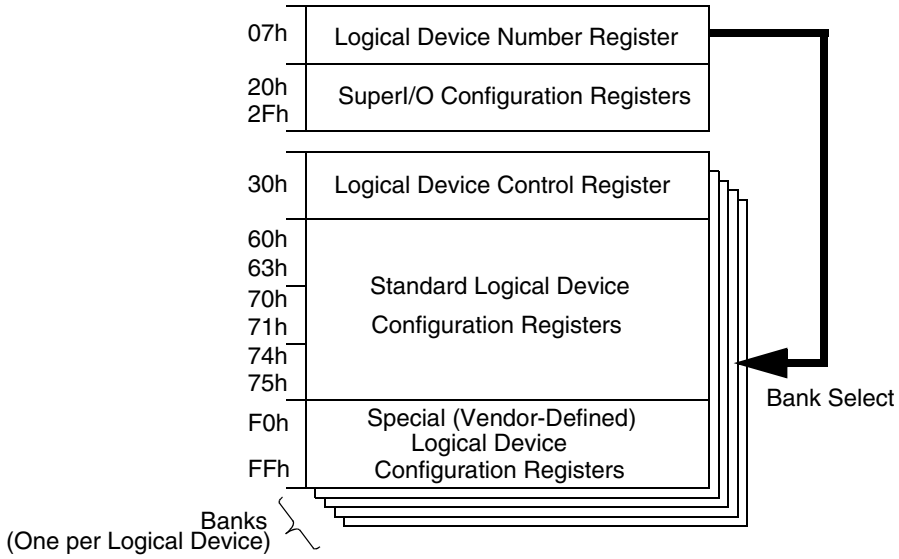


Figure 62. Structure of Standard Configuration Register File

6.0 Host Modules and Host Interface (Continued)

Table 30. Logical Device Number (LDN) Assignments

LDN	Functional Block
04h	Mobile System Wake-Up Control (MSWC)
05h	Keyboard and Mouse Controller (KBC) - Mouse Interface
06h	Keyboard and Mouse Controller (KBC) - Keyboard Interface
07h	Host General-Purpose I/O Ports (HGPIO)
0Fh	Shared Memory (SHM)
11h	Power Management I/F Channel 1 (PM1)
12h	Power Management I/F Channel 2 (PM2)

Write accesses to unimplemented registers (i.e., accessing the data register while the index register points to a non-existing register) are ignored. Reads return 00h for all addresses, except 74h and 75h (DMA configuration registers), which return 04h (indicating that no DMA channel is active). The configuration registers are accessible immediately after Host Domain reset.

Standard Logical Device Configuration Register Definitions

In the registers below, any undefined bit is reserved. Unless otherwise noted, the following definitions also hold true:

- All registers are read/write.
- All reserved bits return 0 on reads except where otherwise noted. To prevent unpredictable results, do not modify these bits. Use read-modify-write to prevent the values of reserved bits from being changed during write.
- Write-only registers must not use read-modify-write during updates.

Table 31. Standard Control Registers

Index	Register Name	Description
07h	Logical Device Number	This register selects the current logical device; see Table 30 for valid numbers. All other values are reserved.
20h - 2Fh	SuperI/O Configuration	SuperI/O configuration registers and ID registers

Table 32. Logical Device Activate Register

Index	Register Name	Description
30h	Activate	Bits 7-1: Reserved Bit 0: Logical device activation control (see “Module Enable/Disable” on page 251) 0: Disabled 1: Enabled

Table 33. I/O Space Configuration Registers

Index	Register Name	Description
60h	I/O Port Base Address Bits (15-8) Descriptor 0	Indicates selected I/O lower limit address bits 15-8 for I/O Descriptor 0.
61h	I/O Port Base Address Bits (7-0) Descriptor 0	Indicates selected I/O lower limit address bits 7-0 for I/O Descriptor 0.
62h	I/O Port Base Address Bits (15-8) Descriptor 1	Indicates selected I/O group 2 lower limit address bits 15-8 for I/O Descriptor 1.
63h	I/O Port Base Address Bits (7-0) Descriptor 1	Indicates selected I/O group 2 lower limit address bits 7-0 for I/O Descriptor 1.

6.0 Host Modules and Host Interface (Continued)

Table 34. Interrupt Configuration Registers

Index	Register Name	Description
70h	Interrupt Number and Wake-Up on IRQ Enable	Indicates selected interrupt number. Bits 7-5: Reserved. Bit 4: Enables wake-up on the IRQ of the logical device. When enabled, IRQ assertion triggers a wake-up event (see "Module IRQ Event" on page 233). 0: Disabled (default) 1: Enabled Bits 3-0: These bits select the interrupt number. A value of 1 selects IRQ1. A value of 15 selects IRQ15. IRQ0 is not a valid interrupt selection and represents no interrupt selection. Note: Avoid selecting the same interrupt number (except 0) for different Logical Devices because it causes the WPC8763L to behave unpredictably.
71h	Interrupt Request Type Select	Indicates the type and polarity of the interrupt request number selected in the previous register. If a logical device supports only one type of interrupt, the corresponding bit is read-only. Bits 7-2: Reserved. Bit 1: Polarity of interrupt request selected in previous register 0: Low polarity 1: High polarity Bit 0: Type of interrupt request selected in previous register 0: Edge 1: Level

Table 35. DMA Configuration Registers

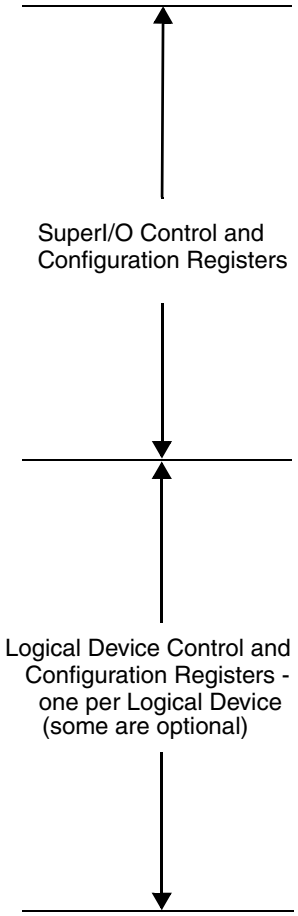
Index	Register Name	Description
74h	DMA Channel Select 0	Indicates selected DMA channel for DMA 0 of the logical device (0 is the first DMA channel if more than one DMA channel is used). Bits 7-3: Reserved. Bits 2-0: These select the DMA channel for DMA 0. The valid choices are 3-0, where: - A value of 0 selects DMA channel 0, 1 selects channel 1, etc. - A value of 4 indicates that no DMA channel is active. - The values 5-7 are reserved. Note: Avoid selecting the same DMA channel (except 4) for different logical devices because it causes the WPC8763L to behave unpredictably.
75h	DMA Channel Select 1	Indicates selected DMA channel for DMA 1 of the logical device (1 is the second DMA channel if more than one DMA channel is used). Bits 7-3: Reserved. Bits 2-0: These select the DMA channel for DMA 1. The valid choices are 3-0, where: - A value of 0 selects DMA channel 0, 1 selects channel 1, etc. - A value of 4 indicates that no DMA channel is active. - The values 5-7 are reserved. Note: Avoid selecting the same DMA channel (except 4) for different logical devices because it causes the WPC8763L to behave unpredictably.

Table 36. Special Logical Device Configuration Registers

Index	Register Name	Description
F0h-FFh	Logical Device Configuration	Special (vendor-defined) configuration options

6.0 Host Modules and Host Interface (Continued)

6.1.2 Standard Configuration Registers



Index	Register Name
07h	Logical Device Number
20h	SuperI/O ID
21h	SuperI/O Configuration 1
25h	SuperI/O Configuration 5
26h	Reserved
27h	SuperI/O Revision ID
28h	Reserved exclusively for Winbond use
29h	Reserved
2Dh	SuperI/O Configuration D
2Eh - 2Fh	Reserved exclusively for Winbond use
30h	Logical Device Control (Activate)
60h	I/O Base Address Descriptor 0 Bits 15-8
61h	I/O Base Address Descriptor 0 Bits 7-0
62h	I/O Base Address Descriptor 1 Bits 15-8
63h	I/O Base Address Descriptor 1 Bits 7-0
70h	Interrupt Number and Wake-Up on IRQ Enable
71h	IRQ Type Select
74h	DMA Channel Select 0
75h	DMA Channel Select 1
F0h - FFh	Device Specific Logical Device Configuration

Figure 63. Configuration Register Map

SuperI/O Control and Configuration Registers

The WPC8763L SuperI/O configuration registers at indexes 20h (SuperI/O ID) and 27h (SuperI/O Revision ID) are used for part identification. The other configuration registers are used for global power management and clock control. For details, see Section 6.1.9 on page 254.

Logical Device Control and Configuration Registers

A subset of these registers is implemented for each logical device; see the functional block descriptions in the sections starting with Section 6.1.10 on page 257.

Control Registers

The only implemented control register for each logical device is the Activate register at index 30h. Bit 0 of the Activate register controls the activation of the associated functional block. Activation enables access to the functional block's runtime registers and assigns system resources to the functional blocks. Note that these resources are unassigned when the block is not activated. Other effects may apply on a function-specific basis (such as clock enable and active pinout signaling). Access to the configuration register of the logical device is enabled even when the logical device is not activated.

Standard Configuration Registers

The standard configuration registers manage the PnP resource allocation to the functional blocks. I/O base address descriptor 0 is a pair of registers at Index 60–61h that hold the first 16-bit base address for the register set of the functional block. An optional 16-bit second base-address (descriptor 1) at index 62–63h is used for logical devices with more than one continuous register set. Interrupt Number and Wake-Up on IRQ Enable (index 70h) and IRQ Type Select (index 71h) allocate an IRQ line to the block and control its type. DMA Channel Select 0 (index 74h) allocates a DMA channel to the block, where applicable. DMA Channel Select 1 (index 75h) allocates a second DMA channel, where applicable.

6.0 Host Modules and Host Interface (Continued)

Vendor-Defined Logical Device Configuration Register

The vendor-defined logical device registers start at index F0h and control function-specific parameters such as operation modes, power saving modes, pin TRI-STATE mode, clock rate selection and non-standard extensions to generic functions.

6.1.3 Default Configuration Setup

The default configuration setup of the WPC8763L host functions is set according to the following reset types (see Section 3.2 on page 48):

- V_{CC} Power-Up reset and $\overline{VCC\_POR}$ Input reset
  - Samples the I/O Base Address straps (BADDR1-0) and the Shared BIOS Memory strap (SHBM).
  - Resets the SuperI/O configuration registers and the MSWC Host registers powered by V_{CC}.
  - Resets the Host GPIO configuration and runtime registers.
  - With the exception of $\overline{VCC\_POR}$ Input reset, resets the Configuration and runtime registers of all the logical devices.
  - Performs all the actions done by a V_{DD} Power-Up reset or Host Hardware reset.
- V_{DD} Power-Up reset and Host Hardware reset
  - Sets the I/O Base Address and Shared BIOS Memory mode according to strap values.
  - Resets all the configuration registers powered by V_{DD}.
  - Resets the Host GPIO Configuration and runtime registers for which VDDLOAD bit in GPCFG2 is set to 1.
  - Resets the Configuration and runtime registers of all the logical devices **except** MSWC Host registers.

After a V_{DD} Power-Up reset or Host Hardware reset, the WPC8763L wakes up with the following default SuperI/O configuration setup:

- The configuration base address is according to the BADDR1-0 strap pins value, as shown in Table 29 on page 246.
- All logical devices are disabled, with the exception of MSWC, Shared Memory and Host GPIO, which remain functional but whose registers cannot be accessed.
- The legacy devices are assigned with their legacy system resource allocation.
- The Winbond proprietary functions are not assigned with any default resources, and the default values of their base addresses are all 00h.

6.1.4 Module Control

Module Enable/Disable

Module control is performed primarily through the Activation bit (bit 0 of index 30h) of each logical device. The operation of each module can be controlled by the host through the LPC bus.

Module enable/disable by the host through the LPC bus is controlled by the following bits:

- Activation bit (bit 0) in index 30h of the Standard configuration registers; see "Standard Logical Device Configuration Register Definitions" on page 248.
- Global Enable bit (GLOBEN) in SIOCF1 register; see "SuperI/O Configuration 1 Register (SIOCF1)" on page 255.

A module is enabled only if all the above-mentioned bits are set to their "enable" value.

When a V_{DD}-powered module (Keyboard and Mouse interface, Power Management Channels, Shared Memory) is disabled, the following occurs:

- The host system resources of the logical device (IRQ and runtime address range) are deassigned.
- Access to the standard- and device-specific Logical Device configuration registers through the LPC bus remains enabled.
- Access to the module's runtime registers through the LPC bus is disabled (transactions are ignored; SYNC cycle is not generated).
- The module is not functional.

When a V_{CC}-powered module (MSWC, Host GPIO) is disabled:

- The host system resources of the logical device (IRQ and runtime address range) are deassigned.
- Access to the standard and device specific Logical Device configuration registers through the LPC bus remains enabled.
- Access to the module's runtime registers through the LPC bus is disabled (transactions are ignored; SYNC cycle is not generated).
- The module remains functional.

6.0 Host Modules and Host Interface (Continued)

6.1.5 Address Decoding

A full 16-bit address decoding is applied when accessing both the configuration I/O space and the registers of the functional blocks. However, the number of configurable bits in the base address registers varies for each logical device.

The lower 0, 1, 2, 3, 4 or 5 address bits are decoded within the functional block to determine the offset of the accessed register within the logical device's I/O range of 1, 2, 4, 8, 16 or 32 bytes, respectively. The rest of the bits are matched with the base address register to decode the entire I/O range allocated to the logical device. Therefore, the lower bits of the base address register are forced to 0 (read-only) and the base address is forced to be 1, 2, 4, 8, 16 or 32 byte-aligned, according to the size of the I/O range.

The base addresses of the Keyboard and Mouse interface are limited to the I/O address range of 0000h to 07FXh only (bits 11-15 are forced to 0). The base addresses of the other devices are configurable within the full 16-bit address range (up to FFFFh).

In some special cases, other address bits are used for internal decoding (such as bit 2 in the KBC). The KBC has two I/O descriptors with some implied dependency between them. For more details, see the description of the base address register for each logical device.

The Shared Memory module serves as a bridge from the LPC to the on-chip RAM and off-chip flash memory. For module control and protection function registers, the 16-bit base address is applied through the configuration address space. To access the registers, the lower three address bits are decoded within the Shared Memory module. The address ranges, in the LPC Memory space and the Firmware Memory memory space, that are bridged to the shared memory, are defined in Section 6.1.13 on page 263. The number of address bits used for this decoding varies according to the specified zones and their sizes.

6.1.6 LPC Interface

Supported LPC Transactions

The WPC8763L LPC interface responds to the following LPC transactions as part of the standard host interface:

- I/O read cycles
- I/O write cycles

In addition, the Shared Memory module uses the following transactions:

- 8-bit LPC Memory (LPC) read and write
- 8-bit Firmware Memory (FWH) read, 8-bit Firmware Memory (FWH) write

LPC transactions conform with Intel's *LPC Interface Specification, Revision 1.1.*

The Firmware Memory read and write protocols are similar to LPC Memory read and write cycles. The specifications of these cycles are listed below. The Address, Data, TAR and SYNC cycles are as specified for LPC Memory read and write cycles. The START and ID fields are similar to the equivalent cycle in LPC Memory read and write transactions but differ in the data placed on the LAD signals (see details in the cycle description).

Note: The WPC8763L supports Firmware Memory transactions from LPC controllers that accept Wait SYNC and Long Wait SYNC cycles.

Firmware Memory Read Cycle

1. START: 1101b (Dh).
2. ID field: FWH ID nibble.
3. Address: Seven address nibbles, MS nibble first (see usage below).
4. MSIZE: Sets the number of bytes to be transferred during each Firmware Memory read transaction (In WPC8763L: one byte only).
5. TAR (two cycles).
6. SYNC.
7. DATA: Two data nibbles, LS nibble first (D3-D0, D7-D4).
8. TAR (two cycles).

Firmware Memory Write Cycle

1. START: 1110b (Eh).
2. ID field: FWH ID nibble (compared with bits 7-4 of FWH_ID_LPC register on page 226).
3. Address: Seven address nibbles MS nibble first (see usage below).
4. MSIZE: Sets the number of bytes to be transferred during each FWH write transaction (In WPC8763L: one byte only).
5. DATA: Two data nibbles, LS nibble first (D3-D0, D7-D4).
6. TAR (two cycles).
7. SYNC.
8. TAR (two cycles).

ID field: This is compared with bits 7-4 of SHM_CFG register on page 264 or with bits 3-0 of FWH_ID_LPC register on page 226. If the two match, the WPC8763L continues handling the transaction; if they do not match, the current Firmware Memory transaction is ignored.

Firmware Memory address translation: The address field in the Firmware Memory transaction contains seven nibbles. They correspond to the first LS seven address nibbles (A27-A0) as follows: the first nibble that appears corresponds to addresses A27-A24, the second to A23-A20, up to the seventh transferred nibble, which corresponds to addresses A3-A0. The MS bits of the 32-bit addresses are '1111' (A31-A28).

MSIZE: The MSIZE nibble sets the number of bytes to be transferred in each transaction. For WPC8763L Firmware Memory transactions, the value of this nibble should be 0h (1-byte transaction).

Core Interrupt

An LPC transaction (I/O or Memory) or Firmware Memory transaction to one of the following generates a positive pulse on the Host Access Wake-Up input of the MIWU:

- Keyboard and Mouse Interface module, runtime registers (LPC transaction only)
- PM channel 1 and PM channel 2, runtime registers (LPC transaction only)
- MSWC module, runtime registers (LPC transaction only)
- Shared Memory module, runtime registers (LPC transaction only)
- On-chip shared RAM or external shared flash

This interrupt can be used to wake up the core to handle any host activity.

$\overline{\text{CLKRUN}}$ Functionality

The WPC8763L supports the $\overline{\text{CLKRUN}}$ I/O signal, the use of which is highly recommended in portable systems. This signal is implemented according to the specification in *PCI Mobile Design Guide, Revision 1.1*, December 18, 1998. The WPC8763L supports operation with stopped clock in ACPI state S0 (the system is active but is not being accessed). The WPC8763L drives the $\overline{\text{CLKRUN}}$ low to force the LPC bus clock into full speed operation when an IRQ is pending internally (i.e., waiting to be sent through the serial IRQ).

$\overline{\text{LPCPD}}$ Functionality

The WPC8763L supports the $\overline{\text{LPCPD}}$ input. This signal is used when the V_{DD} chip supply is not shared by all residents of the LPC bus. The $\overline{\text{LPCPD}}$ signal conforms with Intel's *LPC Interface Specification, Revision 1.1*. Note that if the WPC8763L power supply exists while $\overline{\text{LPCPD}}$ is active, it is not mandatory to reset the WPC8763L when $\overline{\text{LPCPD}}$ is deasserted. When $\overline{\text{LPCPD}}$ is active, a Host Hardware reset can be performed by asserting the $\overline{\text{LRESET}}$ signal.

6.1.7 Interrupt Serializer

The Interrupt Serializer translates internal IRQ sources into serial interrupt request data transmitted over the SERIRQ bus. Figure 64 shows the interrupt serialization mechanism.

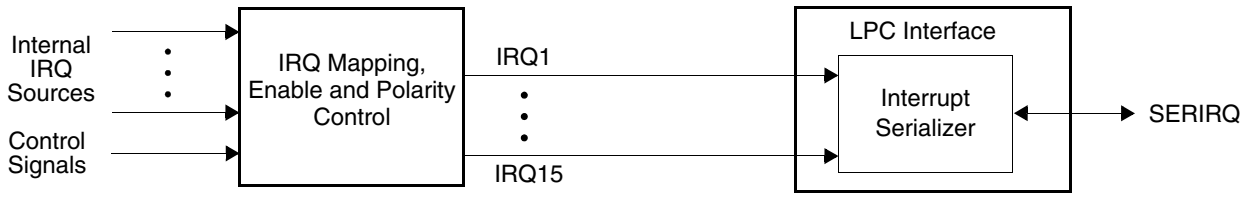


Figure 64. Interrupt Serialization Mechanism

The internal IRQ signals are fed into an IRQ Mapping and Polarity Control block, which maps them to their associated IRQ slots. The IRQs are then fed into the Interrupt Serializer, where they are translated into serial data and transmitted over the SERIRQ bus.

6.1.8 Protection

The WPC8763L device provides features to protect the hardware configuration from changes made by application software running on the host.

The protection is activated when the host software sets a "sticky" lock bit. Each lock bit protects a group of configuration bits located either in the same register or in different registers. When the lock bit is set, the lock bit and all the protected bits become read-only and cannot be modified by the host. All lock bits are reset by V_{DD} Power-Up reset or Host Hardware reset, thus unlocking all the protected bits. Note that the bit locking protection mechanism is optional.

The protected groups of configuration bits are described below.

6.0 Host Modules and Host Interface (Continued)

Host GPIO Pins Configuration Lock

Protects the configuration (but not the data) of all the Host GPIO (HGPIO) pins.

Lock bit: LOCKHGCF in SIOCF1 register (Device Configuration).

Protected bits for each HGPIO pin: LOCKHGCF in SIOCF1 register, and all bits of HGPCFG1 (except LOCKCFP bit), HGPEVR and HGPCFG2 registers (Device Configuration).

Host GPIO Pins Lock

Protects the configuration **and** data of all the Host GPIO (HGPIO) pins.

Lock bit: LOCKCFP in HGPCFG1 register, for each Host GPIO pin (Device Configuration).

Protected bits for each HGPIO pin: LOCKCFP, PUPCTL, OUTTYPE and OUTENA in HGPCFG1 register; all bits of HGPCFG2 register (Device Configuration); the corresponding bit (to the port pin) in HGPDO registers (Host GPIO Ports).

6.1.9 SuperI/O Configuration Registers

This section describes the SuperI/O configuration and ID registers (those registers with first-level indexes in the range of 20h-2Fh; see Figure 62 on page 247). All registers are powered by V_{DD} and are set to their default values by V_{DD} Power-Up or Hardware reset unless specified otherwise in the register description below.

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Table 37. SuperI/O Configuration Registers

Index	Mnemonic	Register Name	Type	Power Well	Reset
20h	SID	SuperI/O ID	RO	V_{DD}	FCh
21h	SIOCF1	SuperI/O Configuration 1	Varies per bit or RO	V_{DD}	11h
22h - 24h	Reserved exclusively for Winbond use				
25h	SIOCF5	SuperI/O Configuration 5	R/W	V_{DD}	00h
26h	Reserved exclusively for Winbond use				
27h	SRID	SuperI/O Revision ID	RO	V_{DD}	See text
28h - 2Ch	Reserved exclusively for Winbond use				
2Dh	SIOCFD	SuperI/O Configuration D	Varies per bit	V_{CC}	00h
2Eh-2Fh	Reserved exclusively for Winbond use				

SuperI/O ID Register (SID)

SID contains the identity number of the device family. The WPC876xL family is identified by the value FCh.

Location: Index 20h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Family ID**							
Reset	FCh							

Bit	Description
7-0	**Family ID.** Identifies a family of devices with similar functionality but with different implemented options.

6.0 Host Modules and Host Interface (Continued)

SuperI/O Configuration 1 Register (SIOCF1)

Location: Index 21h

Type: Varies per bit or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**LOCKHGCF**	**Reserved (must be '01000')**					**GLOBEN**
Reset	0	0	0	1	0	0	0	1

Bit	Type	Description
7		**Reserved.**
6	R/W1S or RO	**LOCKHGCF (Lock Host GPIO Pins Configuration).** When set to 1, locks the configuration registers of all Host GPIO (HGPIO) pins (see Section 6.1.12 on page 259) by disabling writes to all their bits (including the LOCKHGCF bit itself). The locked registers include the HGPCFG1 (except LOCKCFP bit), HGPEVR and HGPCFG2 registers for each Host GPIO pin. Once set, this bit can be cleared by V_{DD} Power-Up reset or Host Hardware reset. 0: Host GPIO configuration registers are R/W (default). 1: Host GPIO configuration registers are RO.
5-1		**Reserved.** Must be '01000'.
0	R/W	**GLOBEN (Global Device Enable).** Allows disabling of all logical devices by changing a single bit (to 0), with the exception of the Shared Memory (SHM) and the Mobile System Wake-Up Control (MSWC) logical devices. When GLOBEN is set to 1, it enables the operation of all the logical devices of the WPC8763L, as long as the logical device is itself enabled. 0: All logical devices in the WPC8763L are disabled except MSWC and Shared Memory. 1: Each logical device is enabled according to its Activate bit (Index 30h) (default).

SuperI/O Configuration 5 Register (SIOCF5)

Location: Index 25h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**$\overline{\text{SMI}}$ to IRQ2 Enable**	**Reserved**			
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-5		**Reserved.**
4	R/W	**$\overline{\text{SMI}}$ to IRQ2 Enable.** Enables routing of $\overline{\text{SMI}}$ to slot number 2 in the SERIRQ protocol in parallel to, or instead of, using the dedicated $\overline{\text{SMI}}$ pin. 0: Disabled (default). 1: Enabled.
3-0		**Reserved.**

6.0 Host Modules and Host Interface (Continued)

SuperI/O Revision ID Register (SRID)

SRID contains the ID number of the specific family member (Chip ID) and the chip revision number (Chip Rev). The Chip Rev is incremented on each revision.

Location: Index 27h

Type: RO

Bit	7	6	5	4	3	2	1	0
Name	**Chip ID**			**Chip Rev**				
Reset	See values in field description			X	X	X	X	X

Bit	Description
7-5	**Chip ID.** Identifies a specific device of a family. For WPC8763L the values of this field are shown below. **Bits 7 6 5 Device** 0 1 1: WPC8763L Other: Reserved
4-0	**Chip Rev.** Identifies the device revision.

SuperI/O Configuration D Register (SIOCFD)

This register is powered by V_{CC} and is set to its default value by V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset.

Location: Index 2Dh

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**Power Supply Off**	**Power Button Mode**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-2		**Reserved.**
1	WO	**Power Supply Off.** This is a write-only bit (it always returns 0 when read). When using Legacy mode (bit 0 is set to 0), setting this bit to 1 indicates to the core that the host requests the core to turn off the V_{DD} power supply. The value of this bit is available to the core through a register in the MSWC; see MSWCTL2 on page 242. 0: No action (default). 1: In Legacy mode, indicates that the host requests the core to turn off the V_{DD} power supply.
0	R/W	**Power Button Mode.** The value of this read-write bit is available to the core through a register in the MSWC; see MSWCTL2 on page 242. 0: Legacy mode (default). 1: ACPI mode.

6.0 Host Modules and Host Interface (Continued)

6.1.10 Mobile System Wake-Up Control (MSWC) Configuration

Logical Device 4 (MSWC) Configuration

Table 38 lists the configuration registers that affect the MSWC; see Sections 6.1.1 and 6.1.2 for a detailed description of these registers. All registers are V_{DD} powered and are set to their default values by V_{DD} Power-Up or Hardware reset.

Table 38. Mobile System Wake-Up Control (MSWC) Configuration Registers

Index	Configuration Register or Action	Type	Power Well	Reset
30h	Activate (see Section 6.1.1 on page 246). When bit 0 is cleared, the registers of this logical device are not accessible[1].	R/W	V_{DD}	00h
60h	Base Address MSB register.	R/W	V_{DD}	00h
61h	Base Address LSB register. Bits 4-0 (for A4-0) are read-only, '00000'.	R/W	V_{DD}	00h
70h	Interrupt Number.	R/W	V_{DD}	00h
71h	Interrupt Type. Bit 1 is read/write. Other bits are read-only.	R/W	V_{DD}	03h
74h	Report no DMA assignment.	RO	V_{DD}	04h
75h	Report no DMA assignment.	RO	V_{DD}	04h

1. The logical device registers are maintained and all wake-up detection mechanisms are functional.

6.0 Host Modules and Host Interface (Continued)

6.1.11 Keyboard and Mouse Interface (KBC) Configuration

Logical Devices 5 and 6 (Mouse and Keyboard) Configuration

Tables 39 and 40 list the configuration registers that affect the Mouse and the Keyboard; see Sections 6.1.1 and 6.1.2 for descriptions of the other configuration registers. The KBC Interface module is activated and access to the runtime registers (pointed to by the base addresses at indexes 60h-63h) is enabled when either the Mouse logical device (5) or the Keyboard logical device (6) is activated (by setting the activation bit at index 30h). Because the IRQ configuration resources are separate for each logical device (Mouse or Keyboard), the specific logical device must be activated to enable its IRQ resources. All registers are V_{DD} powered and are set to their default values by V_{DD} Power-Up or Hardware reset.

Usage Hints: It is recommended to set the type of interrupt request as “level” and its level of interrupt as “high”.

Table 39. Mouse Configuration Registers

Index	Mouse Configuration Register or Action	Type	Power Well	Reset
30h	Activate. See also bit 0 of the SIOCF1. When the Mouse of the KBC is inactive, the IRQ selected by the Mouse Interrupt Number and Wake-Up on IRQ Enable register (index 70h) is not asserted. This register has no effect on host KBC commands handling the Mouse.	R/W	V_{DD}	00h
70h	Mouse Interrupt Number and Wake-Up on IRQ Enable register.	R/W	V_{DD}	0Ch
71h	Mouse Interrupt Type. Bit 1 is read/write; other bits are read-only.	R/W	V_{DD}	03h
74h	Report no DMA assignment.	RO	V_{DD}	04h
75h	Report no DMA assignment.	RO	V_{DD}	04h

Table 40. Keyboard Configuration Registers

Index	Keyboard Configuration Register or Action	Type	Power Well	Reset
30h	Activate; see also bit 0 of the SIOCF1.	R/W	V_{DD}	00h
60h	Base Address MSB register. Bits 7-3 (for A15-11) are read-only, ‘00000’.	R/W	V_{DD}	00
61h	Base Address LSB register. Bits 2-0 are read-only, ‘000’.	R/W	V_{DD}	60
62h	Command Base Address MSB register. Bits 7-3 (for A15-11) are read-only, ‘00000’.	R/W	V_{DD}	00
63h	Command Base Address LSB. Bits 2-0 are read-only, ‘100’.	R/W	V_{DD}	64
70h	KBD Interrupt Number and Wake-Up on IRQ Enable register.	R/W	V_{DD}	01
71h	KBD Interrupt Type. Bit 1 is read/write; other bits are read-only.	R/W	V_{DD}	03
74h	Report no DMA assignment.	RO	V_{DD}	04
75h	Report no DMA assignment.	RO	V_{DD}	04

6.0 Host Modules and Host Interface (Continued)

6.1.12 Host GPIO (HGPIO) Configuration

Logical Device 7 Host GPIO (HGPIO) Configuration

Table 41 lists the configuration registers that affect the Host GPIO. Only the last four registers (F0h-F3h) are described here; see Sections 6.1.1 and 6.1.2 for descriptions of the other configuration registers. The standard configuration registers are powered by V_{DD}; however, the specific configuration registers are powered by V_{CC}. The V_{DD}-powered registers are set to their default values by V_{DD} Power-Up or Hardware reset unless specified otherwise in the register description below. The V_{CC}-powered registers are set to their default values according to VDDLOAD bit in HGPCFG2 register (see page 262), except for LOCKCFP bit in HGPCFG1 register.

Table 41. GPIO Configuration Register

Index	Configuration Register or Action	Type	Power Well	Reset
30h	Activate (see Section 6.1.1 on page 246). When bit 0 is cleared, the registers of this logical device are not accessible.[1]	R/W	V_{DD}	00h
60h	Base Address MSB register.	R/W	V_{DD}	00h
61h	Base Address LSB register. Bits 2-0 (for A2-0) are read-only, '000'.	R/W	V_{DD}	00h
70h	Interrupt Number and Wake-Up on IRQ Enable register.	R/W	V_{DD}	00h
71h	Interrupt Type. Bit 1 is read/write. Other bits are read-only.	R/W	V_{DD}	03h
74h	Report no DMA assignment.	RO	V_{DD}	04h
75h	Report no DMA assignment.	RO	V_{DD}	04h
F0h	Host GPIO Pin Select register (HGPSEL).	R/W	V_{CC}	00h
F1h	Host GPIO Pin Configuration Register 1 (HGPCFG1).	Varies per bit or RO	V_{CC}	44h
F2h	Host GPIO Pin Event Routing register (HGPEVR).	R/W or RO	V_{CC}	00h
F3h	Host GPIO Pin Configuration Register 2 (HGPCFG2).	R/W or RO	V_{CC}	00h

1. The logical device registers are maintained, and event detection mechanisms are functional.

Figure 65 shows the organization of registers HGPCFG1-2 and HGPEVR:

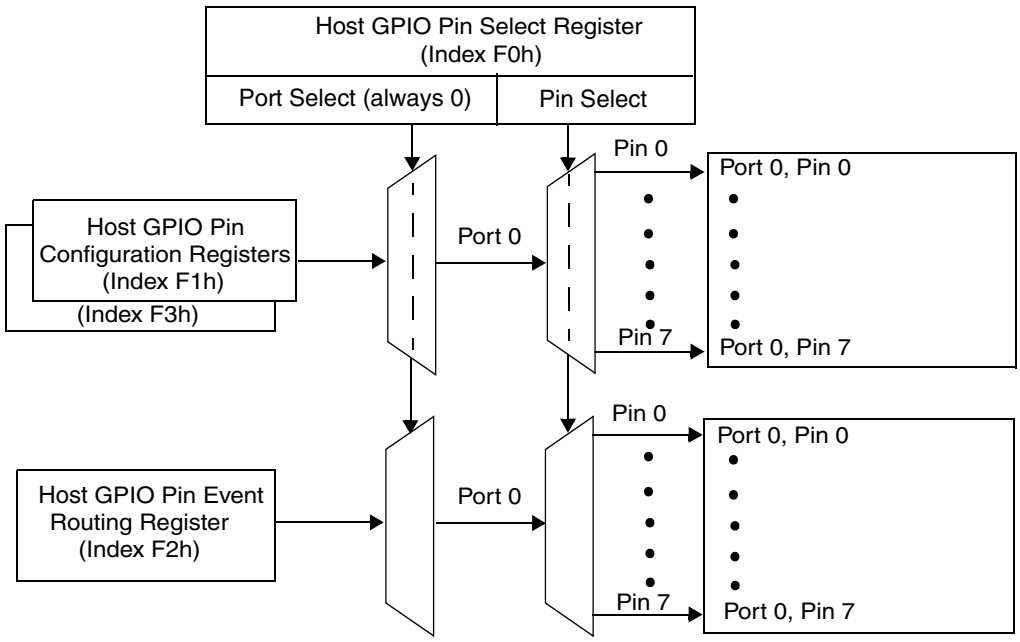


Figure 65. Organization of Host GPIO Pin Registers HGPCFG1-2 and HGPEVR

6.0 Host Modules and Host Interface (Continued)

Host GPIO Pin Select Register (HGPSEL)

HGPSEL selects the Host GPIO pin (port number and bit number) to be configured (i.e., which register is accessed via the Host GPIO pin configuration registers). HGPSEL is reset to 00h.

Location: Index F0h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**PORTSEL**			**Reserved**	**PINSEL**		
Reset	0	0	0	0	0	0	0	0

Bit	Description
7	**Reserved.**
6-4	**PORTSEL (Port Select).** Selects the Host GPIO port to be configured. In the WPC8763L, only one port (port 0) is implemented. **Bits** **6 5 4 Mode** 0 0 0: Port 0 (default). 001-111: Reserved.
3	**Reserved.**
2-0	**PINSEL (Pin Select).** Selects the Host GPIO pin of the selected port to be configured. **Bits** **2 1 0 Mode** 0 0 0: Pin 0 (default). 001-111: Binary value of pin numbers 1-7, respectively.

Host GPIO Pin Configuration Register 1 (HGPCFG1)

HGPCFG1 reflects, for both read and write, the register currently selected by the Host GPIO Pin Select register. All Host GPIO pin configuration registers have a common bit structure, as shown below.

Location: Index F1h

Type: Varies per bit or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**EVDBNC**	**EVPOL**	**EVTYPE**	**LOCKCFP**	**PUPCTL**	**OUTTYPE**	**OUTENA**
Reset	0	1	0	0	0	1	0	0

Bit	Type	Description
7		**Reserved.**
6	R/W or RO	**EVDBNC (Event Debounce Enable).** Enables the debounce circuit in the event input path of the selected Host GPIO pin. The event is detected after a predetermined debouncing period; see Section 6.2.3 on page 271. 0: Disabled. 1: Enabled (default).
5	R/W or RO	**EVPOL (Event Polarity).** Defines the polarity of input signal that issues an event from the selected Host GPIO pin; see Section 6.2.3 on page 271. 0: Falling edge or low level input (default). 1: Rising edge or high level input.
4	R/W or RO	**EVTYPE (Event Type).** Defines the type of input signal that issues an event from the selected Host GPIO pin; see Section 6.2.3 on page 271. 0: Edge input (default). 1: Level input.

Bit	Type	Description
3	R/W1S	**LOCKCFP (Lock Configuration of Pin).** When set to 1, locks the Host GPIO pin configuration and data (also see Section 6.2.4 on page 273) by disabling writing to itself, to HGPCFG1 register bits PUPCTL, OUTTYPE and OUTENA, to all bits of HGPCFG2 register and to the corresponding bit in HGPDO register. Once set, this bit can be cleared only by V_{DD} Power-Up reset, or Host Hardware reset. 0: R/W bits are enabled for write (default). 1: All bits are RO.
2	R/W or RO	**PUPCTL (Pull-Up Control).** Controls the internal pull-up resistor of the selected Host GPIO pin; see Section 6.2.2 on page 271. 0: Disabled. 1: Enabled (default).
1	R/W or RO	**OUTTYPE (Output Type).** Controls the output buffer type of the selected Host GPIO pin; see Section 6.2.2 on page 271. 0: Open-drain (default). 1: Push-pull.
0	R/W or RO	**OUTENA (Output Enable).** Controls the output buffer of the selected Host GPIO pin; see Section 6.2.2 on page 271. 0: TRI-STATE mode (default). 1: Output buffer enabled.

Host GPIO Event Routing Register (HGPEVR)

HGPEVR reflects, for both read and write, the register currently selected by the Host GPIO Pin Select register. It enables the routing of the Host GPIO event (see Section 6.2.3 on page 271) to an IRQ and/or $\overline{SMI}$.

Location: Index F2h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**						**EV2SMI**	**EV2IRQ**
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-2	**Reserved.**
1	**EV2SMI (Event to SMI Routing).** Controls the routing of the event from the selected Host GPIO pin to $\overline{SMI}$; see Section 6.2.3 on page 271. 0: Disabled (default). 1: Enabled.
0	**EV2IRQ (Event to IRQ Routing).** Controls the routing of the event from the selected Host GPIO pin to IRQ; see Section 6.2.3 on page 271. 0: Disabled (default). 1: Enabled.

6.0 Host Modules and Host Interface (Continued)

Host GPIO Pin Configuration Register 2 (HGPCFG2)

HGPCFG2 reflects, for both read and write, the register currently selected by the Host GPIO Pin Select register. It controls the connection of the Host GPIO pin to a V_{DD}-powered load.

Location: Index F3h

Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**			**VDDLOAD**	**Reserved**			
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-5	**Reserved.**
4	**VDDLOAD (V_{DD}-Powered Load).** Indicates that the selected Host GPIO pin is connected to a device powered by V_{DD}. When this bit is 1, the input and output buffers (including the internal pull-up) of the selected Host GPIO pin are disabled when V_{DD} power to the WPC8763L device falls below a certain value; see Section 8.1.5 on page 279. 0: Host GPIO pin connected to a V_{CC}-powered load (default): The configuration and data of the Host GPIO pin are reset by V_{CC} Power-Up reset or $\overline{VCC\_POR}$ Input reset (see Section 3.2 on page 48). 1: Host GPIO pin connected to a V_{DD}-powered load: The configuration (excepting the VDDLOAD bit) and data of the Host GPIO pin are reset by V_{DD} Power-Up reset or Host Hardware reset (see Section 3.2 on page 48).
3-0	**Reserved.**

6.0 Host Modules and Host Interface (Continued)

6.1.13 Shared Memory (SHM) Configuration

Logical Device 15 (0Fh) Shared Memory (SHM) Configuration

Table 42 lists the configuration registers that affect the Shared Memory functional block. The Shared Memory base address registers point to the Shared Memory registers described in Section 5.4.9 on page 221. The memory space to which the Shared Memory responds is defined by the configuration registers in the following sections; see Sections 6.1.1 and 6.1.2 for descriptions of the other configuration registers. All registers are powered by V_{DD} and are set to their default values by V_{DD} Power-Up or Hardware reset.

For additional information on the device-specific registers (index F0h-FBh), see Section 5.4.2 on page 216.

Table 42. Shared Memory Configuration Registers

Index	Configuration Register or Action	Type	Power Well	Reset
30h	Activate (see Section 6.1.1 on page 246).	R/W	V_{DD}	00h
60h	Base Address MSB register.	R/W	V_{DD}	00h
61h	Base Address LSB register. Bits 2-0 (for A2-A0) are read-only, '000'.	R/W	V_{DD}	00h
70h	No interrupt assignment.	RO	V_{DD}	00h
71h	No interrupt assignment.	RO	V_{DD}	00h
74h	Report no DMA assignment.	RO	V_{DD}	04h
75h	Report no DMA assignment.	RO	V_{DD}	04h
F0h	Shared Memory Configuration register (SHM_CFG).	R/W	V_{DD}	00h or 09h[1]
F1h	Shared Access Windows Configuration register (WIN_CFG).	Varies per bit	V_{DD}	07h
F2h-F3h	Reserved			
F4h	Shared Access Window 1, Base Address 0 register (SHAW1BA_0).	R/W	V_{DD}	00h
F5h	Shared Access Window 1, Base Address 1 register (SHAW1BA_1).	R/W	V_{DD}	00h
F6h	Shared Access Window 1, Base Address 2 register (SHAW1BA_2).	R/W	V_{DD}	00h
F7h	Shared Access Window 1, Base Address 3 register (SHAW1BA_3).	R/W	V_{DD}	00h
F8h	Shared Access Window 2, Base Address 0 register (SHAW2BA_0).	R/W	V_{DD}	00h
F9h	Shared Access Window 2, Base Address 1 register (SHAW2BA_1).	R/W	V_{DD}	00h
FAh	Shared Access Window 2, Base Address 2 register (SHAW2BA_2).	R/W	V_{DD}	00h
FBh	Shared Access Window 2, Base Address 3 register (SHAW2BA_3).	R/W	V_{DD}	00h

1. The reset value depends on the setting of the SHBM strap input.

6.0 Host Modules and Host Interface (Continued)

Shared Memory Configuration Register (SHM_CFG)

SHM_CFG is reset to either 00h or 09h, depending on the value of the SHBM strap input.

Location: Index F0h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**SHW_FWH_ID**				**BIOS_FWH_EN**	**FLASH_ACC_EN**	**BIOS_EXT_EN**	**BIOS_LPC_EN**
Reset	0	0	0	0	Strap	0	0	Strap

Bit	Type	Description
7-4	R/W	**SHW_FWH_ID (Shared Windows FWH ID Value).** When SHWIN_ACC bit in WIN_CFG register is set to 1, these four bits are compared with the identification nibble (see Section 6.1.6 on page 252 for details) of Firmware Memory transactions to Shared Access Window 1 and 2 (for Window 1, this applies to both RAM and flash access modes).
3	R/W	**BIOS_FWH_EN (BIOS Firmware Memory Access Enable).** When this bit is set to 1, the WPC8763L responds to Firmware Memory transactions to the BIOS Firmware Memory address space. The reset value of BIOS_FWH_EN is defined by the SHBM strap input. The value of this bit is updated later by the WPC8763L, based on the detected host BIOS protocol. 0: Access disabled (default when SHBM = 1, i.e., disable shared BIOS memory). 1: Access enabled (default when SHBM = 0, i.e., enable shared BIOS memory).
2	R/W	**FLASH_ACC_EN (Flash Memory Access Enable).** When this bit is set to 1, the Shared Access Window 1 is used for flash access instead of for RAM access. The WPC8763L responds to either LPC Memory transactions or Firmware Memory transactions (selected by SHWIN_ACC bit in WIN_CFG register) to the Shared Flash Access Window 1 as if they were accessing the BIOS address space. The window base address is specified by SHAW1BA_3-2 registers; its size is specified by FWIN1_SIZE field in WIN_CFG register. 0: Shared Access window 1 used to access internal RAM (default). 1: Shared Access window 1 used to access expansion flash memory.
1	R/W	**BIOS_EXT_EN (BIOS Extended Space Enable).** When set to 1, the WPC8763L responds to LPC Memory transactions to the Extended BIOS LPC address space. 0: Access disabled (default). 1: Access enabled.
0	R/W	**BIOS_LPC_EN (BIOS LPC Access Enable).** When set to 1, the WPC8763L responds to LPC Memory transactions to the BIOS LPC address space. The reset value of BIOS_LPC_EN is defined by the SHBM strap input. The value of this bit is updated later by the WPC8763L, based on the detected host BIOS protocol. 0: Access disabled (default when SHBM = 1, i.e., disable shared BIOS memory). 1: Access enabled (default when SHBM = 0, i.e., enable shared BIOS memory).

Shared Access Windows Configuration Register (WIN_CFG)

Location: Index F1h

Type: Varies per bit

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**	**SHWIN_ACC**	**Reserved**		**FWIN1_SIZE**			
Reset	0	0	0	0	0	1	1	1

Bit	Type	Description
7		**Reserved.**

Bit	Type	Description
6	R/W	**SHWIN_ACC (Shared Windows Access Select).** Selects the LPC transaction type supported by Shared Access Window 1 and 2. For Window 1, the selected transaction type applies to both RAM and flash access modes. 0: Shared Access Window 1 and 2 respond to LPC Memory transactions (default). 1: Shared Access Window 1 and 2 respond to Firmware Memory transactions.
5-4		**Reserved.**
3-0	R/W	**FWIN1_SIZE (Flash Access Window 1, Size Select).** Selects the size of the Shared Access Window 1 when the window is used for flash access (i.e., when FLASH_ACC_EN bit in SHM_CFG register is set to 1). Since FWIN1_SIZE defines the area allocated for flash access, it must select a size equal to or greater than the size of the flash expansion memory connected to WPC8763L. FWIN1_SIZE is irrelevant if FLASH_ACC_EN bit in SHM_CFG register is set to 0. **Bits 3 2 1 0 — Flash Window Size** 0 1 1 1: 128 Kbytes (default) 1 0 0 0: 256 Kbytes 1 0 0 1: 512 Kbytes 1 0 1 0: 1 Mbyte 1 0 1 1: 2 Mbytes 1 1 0 0: 4 Mbytes Others: Reserved

Shared Window 1, Base Address 0 Register (SHAW1BA_0)

Registers SHAW1BA_3-0 define the base address of Shared Access Window 1. Their contents depends on the setting of both SHWIN_ACC bit in WIN_CFG register and FLASH_ACC_EN bit in SHM_CFG register, as follows:

- SHWIN_ACC = 0 (LPC Memory access):
  - FLASH_ACC_EN = 0 (RAM access): SHAW1BA_3-0 registers define bits 31-0 of the window base address. The base address must be aligned on the window block size, selected by RWIN1_SIZE field in WIN_SIZE register, and must not overlap with Shared Access Window 2.
  - FLASH_ACC_EN = 1 (Flash access): SHAW1BA_3-2 registers define bits 31-17 of the window base address. SHAW1BA_1-0 registers are ignored. The base address must be aligned on the window block size, selected by FWIN1_SIZE field in WIN_CFG register.
- SHWIN_ACC = 1 (Firmware Memory access - bits 31-28 of the address are replaced by the ID nibble):
  - FLASH_ACC_EN = 0 (RAM access): SHAW1BA_3-0 registers define bits 27-0 of the window base address. Bits 7-4 of SHAW1BA_3 register are ignored. The base address must be aligned on the window block size, selected by RWIN1_SIZE field in WIN_SIZE register, and must not overlap with Shared Access Window 2.
  - FLASH_ACC_EN = 1 (Flash access): SHAW1BA_3-2 registers define bits 27-17 of the window base address. Bits 7-4 of SHAW1BA_3 register are ignored. SHAW1BA_1-0 registers are ignored. The base address must be aligned on the window block size, selected by FWIN1_SIZE field in WIN_CFG register.

When **all** the SHAW1BA_3-0 registers are set to 00h, Shared Access Window 1 is disabled.

Location: Index F4h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_0**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_0 (Window Base Address, Byte 0).** Defines byte 0 (address bits A7-A0) of the base address of Shared Access Window 1. WIMBA_0 is irrelevant if FLASH_ACC_EN bit in SHM_CFG register is set to 1 (flash access).

6.0 Host Modules and Host Interface (Continued)

Shared Window 1, Base Address 1 Register (SHAW1BA_1)

Location: Index F5h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_1**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_1 (Window Base Address, Byte 1).** Defines byte 1 (address bits A15-A8) of the base address of Shared Access Window 1. WIMBA_1 is irrelevant if FLASH_ACC_EN bit in SHM_CFG register is set to 1 (flash access).

Shared Window 1, Base Address 2 Register (SHAW1BA_2)

The contents of SHAW1BA_2 depend on the setting of FLASH_ACC_EN bit in SHM_CFG register.

Location: Index F6h

Type: R/W

For RAM Access:

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_2_RAM**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_2_RAM (Window Base Address, Byte 2 - RAM Access).** Defines byte 2 (address bits A23-A16) of the base address of Shared Access Window 1 when FLASH_ACC_EN bit in SHM_CFG register is set to 0 (RAM access).

For Flash Access:

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_2_FLASH**							**Reserved**
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-1	R/W	**WINBA_2_FLASH (Window Base Address, Byte 2 - Flash Access).** Defines byte 2 (address bits A23-A17) of the base address of Shared Access Window 1 when FLASH_ACC_EN bit in SHM_CFG register is set to 1 (flash access).
0		**Reserved.** Reserved when FLASH_ACC_EN bit in SHM_CFG register is set to 1.

6.0 Host Modules and Host Interface (Continued)

Shared Window 1, Base Address 3 Register (SHAW1BA_3)

Location: Index F7h

Type: R/W

For LPC Memory Access:

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_3_LPC**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_3_LPC (Window Base Address, Byte 3 for LPC Memory Access).** Defines byte 3 (address bits A31-A24) of the base address of Shared Access Window 1 when SHWIN_ACC bit in WIN_CFG register is set to 0 (LPC Memory access).

For Firmware Memory Access:

Bit	7	6	5	4	3	2	1	0
Name	**Reserved**				**WINBA_3_FWH**			
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-4		**Reserved for Firmware Memory Access.** Reserved when SHWIN_ACC bit in WIN_CFG register is set to 1 (Firmware Memory access).
3-0	R/W	**WINBA_3_FWH (Window Base Address, Byte 3 for Firmware Memory Access).** Defines address bits A27-A24 of the base address of Shared Access Window 1 when SHWIN_ACC bit in WIN_CFG register is set to 1 (Firmware Memory access).

Shared Window 2, Base Address 0 Register (SHAW2BA_0)

Registers SHAW2BA_3-0 define the base address of Shared Access Window 2. Their contents depends on the setting of SHWIN_ACC bit in WIN_CFG register, as follows:

- SHWIN_ACC = 0 (LPC Memory access): SHAW2BA_3-0 registers define bits 31-0 of the window base address. The base address must be aligned on the window block size, selected by RWIN2_SIZE field in WIN_SIZE register, and must not overlap with Shared Access Window 1.
- SHWIN_ACC = 1 (Firmware Memory access - bits 31-28 of the address are replaced by the ID nibble): SHAW2BA_3-0 registers define bits 27-0 of the window base address. Bits 7-4 of SHAW2BA_3 register are ignored. The base address must be aligned on the window block size, selected by RWIN2_SIZE field in WIN_SIZE register, and must not overlap with Shared Access Window 1.

When **all** the SHAW2BA_3-0 registers are set to 00h, the Shared Access Window 2 is disabled.

Location: Index F8h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_0**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_0 (Window Base Address, Byte 0).** Defines byte 0 (address bits A7-A0) of the base address of Shared Access Window 2.

6.0 Host Modules and Host Interface (Continued)

Shared Window 2, Base Address 1 Register (SHAW2BA_1)

Location: Index F9h

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_1**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_1 (Window Base Address, Byte 1).** Defines byte 1 (address bits A15-A8) of the base address of Shared Access Window 2.

Shared Window 2, Base Address 2 Register (SHAW2BA_2)

Location: Index FAh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_2**							
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_2 (Window Base Address, Byte 2).** Defines byte 2 (address bits A23-A16) of the base address of Shared Access Window 2.

Shared Window 2, Base Address 3 Register (SHAW2BA_3)

Location: Index FBh

Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	**WINBA_3_LPC**							
Name	**Reserved**				**WINBA_3_FWH**			
Reset	0	0	0	0	0	0	0	0

Bit	Type	Description
7-0	R/W	**WINBA_3_LPC (Window Base Address, Byte 3 for LPC Memory Access).** Defines byte 3 (address bits A31-A24) of the base address of Shared Access Window 2 when SHWIN_ACC bit in WIN_CFG register is set to 0 (LPC Memory access).
3-0	R/W	**WINBA_3_FWH (Window Base Address, Byte 3 for Firmware Memory Access).** Defines address bits A27-A24 of the base address of Shared Access Window 2 when SHWIN_ACC bit in WIN_CFG register is set to 1 (Firmware Memory access).
7-4		**Reserved for Firmware Memory Access.** Reserved when SHWIN_ACC bit in WIN_CFG register is set to 1 (Firmware Memory access).

6.0 Host Modules and Host Interface (Continued)

6.1.14 Power Management Interface Channel 1 (PM1) Configuration

Logical Device 17 (11h) Power Management Channel 1

Table 43 lists the configuration registers that affect Power Management I/F Channel 1; see Sections 6.1.1 and 6.1.2 for descriptions of the other configuration registers. The standard configuration registers are powered by V_{DD} and are set to their default values by V_{DD} Power-Up or Hardware reset.

Table 43. Power Management Configuration Registers

Index	Power Management Channel 1 Configuration Register or Action	Type	Power Well	Reset
30h	Activate (see Section 6.1.1 on page 246); see also bit 0 of the SIOCF1.	R/W	V_{DD}	00h
60h	Data Register Base Address MSB register.	R/W	V_{DD}	00h
61h	Data Register Base Address LSB register.	R/W	V_{DD}	62h
62h	Command/Status Base Address MSB register.	R/W	V_{DD}	00h
63h	Command/Status Base Address LSB.	R/W	V_{DD}	66h
70h	PM Interrupt Number and Wake-Up on IRQ Enable register.	R/W	V_{DD}	01h
71h	PM Interrupt Type. Bit 1 is read/write; other bits are read-only.	R/W	V_{DD}	03h
74h	Report no DMA assignment.	RO	V_{DD}	04h
75h	Report no DMA assignment.	RO	V_{DD}	04h

6.1.15 Power Management Interface Channel 2 (PM2) Configuration

Logical Device 18 (12h) Power Management Channel 2

Table 44 lists the configuration registers that affect Power Management I/F Channel 2; see Sections 6.1.1 and 6.1.2 for descriptions of the other configuration registers. The standard configuration registers are powered by V_{DD} and are set to their default values by V_{DD} Power-Up or Hardware reset.

Table 44. Power Management Configuration Registers

Index	Power Management 2 Configuration Register or Action	Type	Power Well	Reset
30h	Activate (see Section 6.1.1 on page 246); see also bit 0 of the SIOCF1.	R/W	V_{DD}	00h
60h	Data Register Base Address MSB register.	R/W	V_{DD}	00h
61h	Data Register Base Address LSB register.	R/W	V_{DD}	68h
62h	Command/Status Base Address MSB register.	R/W	V_{DD}	00h
63h	Command/Status Base Address LSB.	R/W	V_{DD}	6Ch
70h	PM Interrupt Number and Wake-Up on IRQ Enable register.	R/W	V_{DD}	01h
71h	PM Interrupt Type. Bit 1 is read/write; other bits are read-only.	R/W	V_{DD}	03h
74h	Report no DMA assignment.	RO	V_{DD}	04h
75h	Report no DMA assignment.	RO	V_{DD}	04h

6.0 Host Modules and Host Interface (Continued)

6.2 HOST GENERAL-PURPOSE INPUT/OUTPUT (HGPIO) PORT

This section describes one 8-bit port. A device may include a combination of several ports with different implementations. In the WPC8763L, only one port (port number 0) is implemented; see Section 6.1.12 on page 259.

6.2.1 Overview

The HGPIO port is an 8-bit port, connected to eight pins. It features:

- Software capability to control and read pin levels.
- Flexible system notification by several types, based on the pin level or level transition.
- Ability to capture and route events and their associated status.
- Back-drive protected pins.

HGPIO port operation is associated with two sets of registers:

- Pin configuration registers, mapped in the Device Configuration space. These registers are used to configure the logical behavior of each pin. Each Host GPIO pin has the following registers: Host GPIO Pin Configuration Registers 1 and 2 (GPCFG1, GPCFG2) and the Host GPIO Pin Event Routing register (GPEVR).
- 8-bit runtime registers: Host GPIO Data Out (HGPDO), Host GPIO Data In (HGPDI), Host GPIO Event Enable (HGPEVEN) and Host GPIO Event Status (HGPEVST). These registers are mapped in the Host GPIO device I/O space (which is determined by the base address registers in the Host GPIO Device Configuration). They are used to control and/or read the pin values and to handle system notification. Each runtime register corresponds to the 8-pin port, such that bit 'n' in each register is associated with a GPIOXn pin, where 'X' is the port number (in the WPC8763L, X=0).

Each Host GPIO pin is associated with configuration bits and the corresponding bit slice of the runtime registers, as shown in Figure 66.

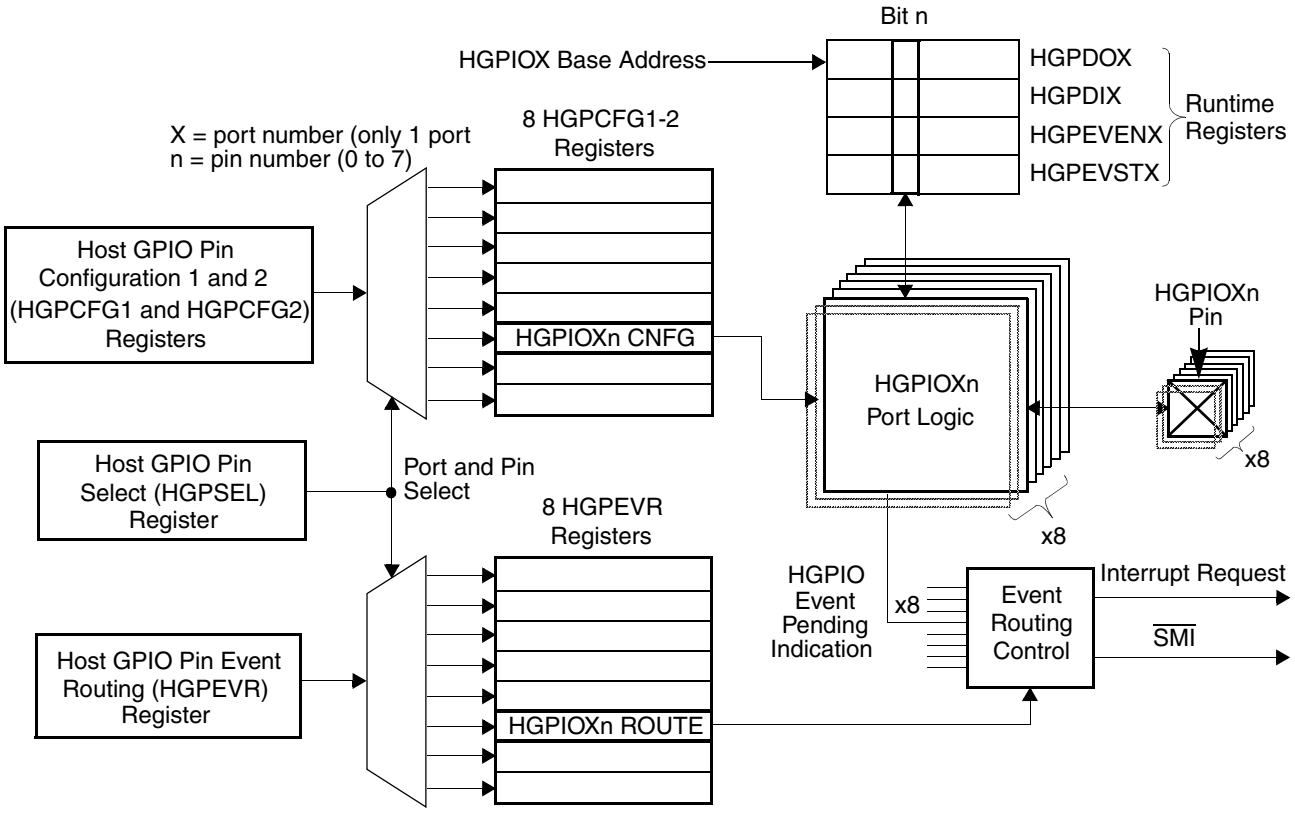


Figure 66. Host GPIO Port Architecture

The functionality of the Host GPIO port is divided into:

- Basic functionality: Includes configuration of, writing to and reading from Host GPIO pins (see Section 6.2.2).
- Enhanced functionality: Includes system notification (see Section 6.2.3).

6.0 Host Modules and Host Interface (Continued)

6.2.2 Basic Functionality

The basic functionality of each GPIO pin is based on four configuration bits and a bit slice of runtime registers HGPDO and HGPDI.

Configuration Options

The HGPCFG1 register controls the following basic configuration options:

- Port Direction: Controlled by Output Enable (bit 0).
- Output Type: Push-pull vs. open-drain; it is controlled by Output Buffer Type (bit 1) by enabling/disabling the upper transistor of the output buffer.
- Static Pull-Up: May be added to any type of port (input, open-drain or push-pull). It is controlled by Pull-Up Control (bit 2).
- Pin Lock: A Host GPIO pin may be locked to prevent any changes in the output value and/or the output configuration. The lock is controlled by bit 3, which disables writes to HGPDO registers bits, to bits 7, 3–0 of HGPCFG1 register (including the Lock bit itself) and to bit 4 of HGPCFG2 register.

HGPCFG2 register controls the Load Protection "V_{DD}-Powered Load" protection configuration option. If the specific GPIO pin is connected to a V_{DD}-powered device and V_{DD} power to the device is not present (No_Vdd), the following are disabled: the Output Buffer (if enabled), the Static Pull-Up (if enabled) and the Input Buffer. This function is controlled by the V_{DD}-powered Load bit (bit 4).

Operation

If the HGPIO output is enabled, the value that is written to the HGPDO register is driven to the pin. Reading from HGPDO register returns its contents regardless of the actual pin value or the port configuration.

HGPDI register is a read-only register. Reading from it returns the actual pin value regardless of its source (the port itself or an external device). Writing to HGPDI register is ignored.

Activation of the Host GPIO module is controlled by device-specific configuration bits. When this module is inactive, access through the LPC bus to the runtime registers (HGPDI, HGPDO) is disabled; however, there is no change in the HGPDO values and therefore, there is no effect on the outputs of the pins.

The configuration and data registers of each Host GPIO pin are reset according to the setting of VDDLOAD bit in GPCFG2 register; see Host GPIO Pin Configuration Register 2 (HGPCFG2) on page 262.

6.2.3 Event Handling and System Notification

The Host GPIO port supports system notification based on event detection. This functionality is based on configuration bits and a bit slice of runtime registers HGPEVEN and HGPEVST. System notification is described on page 272.

Event Configuration

Each pin in the Host GPIO port is a potential input event source. The event detection can trigger a system notification on pre-determined behavior of the source pin. HGPCFG1 register determines the event detection trigger type for system notification.

Event Debounce Enable

The input signal can be debounced for at least 16 ms before entering the detector. To ensure that the signal is stable, the signal state is transferred to the event detector only after a debouncing period during which the signal has no transitions. The debouncer adds a delay time, equal to the debouncing period, to both the assertion and deassertion of the event pending indicator (IRQ, $\overline{SMI}$). The debounce is controlled by EVDBNC (bit 6 of HGPCFG1 register).

Event Type and Polarity

Two trigger types of event detection are supported: edge and level. An edge-triggered event may be detected on a source pin transition either from high to low or low to high. A level-triggered event may be detected when the source pin is either at high or low level. The trigger type is determined by EVTYPE (bit 4 of HGPCFG1 register). The direction of the transition (for edge) or the polarity of the active level (for level) is determined by EVPOL (bit 5 of HGPCFG1 register).

"Active edge" refers to a **change** in a Host GPIO pin level that matches the EVPOL bit (1 for rising edge and 0 for falling edge). "Active level" refers to the Host GPIO pin **level** that matches the Event Polarity bit (1 for high level and 0 for low level). The corresponding bit in HGPEVST register is set by hardware when an active edge or an active level is detected, regardless of the HGPEVEN register setting. Writing 1 to the status bit clears it to 0. Writing 0 is ignored.

A Host GPIO pin is in event pending state if an active event occurred (the corresponding bit in HGPEVST register is set) and the corresponding bit in HGPEVEN register is set.

6.0 Host Modules and Host Interface (Continued)

System Notification

System notification on Host GPIO-triggered events is achieved by asserting an Interrupt Request (via the Interrupt Serializer in the LPC Bus interface) or a System Management Interrupt ($\overline{SMI}$).

The system notification for each Host GPIO pin is controlled by the corresponding bit in HGPEVEN register together with the bits of HGPEVR register. System notification by a Host GPIO pin is enabled if the corresponding bit of HGPEVEN register is set to 1. The bits of the HGPEVR register select the type of system notification (IRQ, $\overline{SMI}$) that the detected Host GPIO event is routed to.

The system notification to the target is asserted if at least one Host GPIO pin is in event pending state.

The selection of the target (for system notification) is determined by HGPEVR register. The specific IRQ number is determined by the IRQ selection procedure of the device configuration. The assertion of IRQ (as a type of system notification) is disabled either when the Host GPIO functional block is deactivated or when V_{DD} power is off.

The assertion of $\overline{SMI}$ is independent of the activation of the Host GPIO functional block.

System notification via IRQ or $\overline{SMI}$ can be initiated by software by writing to the Data Out bit (in HGPDO register) of a Host GPIO pin. This is possible only if the output of the corresponding Host GPIO pin is enabled, pin multiplexing is selected for the Host GPIO function (see Section 2.3 on page 23) and the Host GPIO event is routed to IRQ or $\overline{SMI}$. System notification is asserted according to the actual level at the Host GPIO pin driven by the Host GPIO output and/or by external circuitry.

A pending edge-triggered event may be cleared by clearing the corresponding HGPEVST bit. However, a level-triggered event cannot be released by software (except for disabling the source) as long as the pin is at active level. When a level-triggered event is used, it is also recommended to disable the input debouncer.

On deactivation of the Host GPIO functional block and while V_{DD} power is off, access through the LPC bus to the runtime registers (HGPEVST and HGPEVEN) is disabled. All types of system notification that include the target IRQ number are detached from the Host GPIO and deasserted.

When V_{DD} power is off, the status bits of the V_{DD}-powered Host GPIO pins (VDDLOAD = 1) are cleared; however, the status bits of the V_{CC}-powered Host GPIO pins (VDDLOAD = 0) are not affected.

Before enabling system notification, it is recommended to set the desired event configuration and then verify that the status registers are cleared.

6.0 Host Modules and Host Interface (Continued)

6.2.4 Host GPIO Port Registers

For a summary of the abbreviations used for Register Type, see "Register Abbreviations and Access" on page 20.

Host GPIO Pin Configuration Registers Structure

For each Host GPIO Port, there is a group of eight identical sets of configuration registers. Each set is associated with one Host GPIO pin. The entire group is mapped to the PnP configuration space. The mapping mechanism is based on HGPSEL register (see page 260), which functions as an index register for the pin, and the selected HGPCFG1, HGPEVR and HGPCFG2 registers, which reflect the configuration of the currently selected pin (see Table 45). All registers are V_{CC}-powered.

Table 45. Host GPIO Pin Configuration Registers

Index	Configuration Register or Action	Type	Power Well	Reset
F0h	Host GPIO Pin Select register (HGPSEL)	R/W	V_{CC}	00h
F1h	Host GPIO Pin Configuration Register 1 (HGPCFG1)	Varies per bit	V_{CC}	44h
F2h	Host GPIO Pin Event Routing register (HGPEVR)	R/W or RO	V_{CC}	00h
F3h	Host GPIO Pin Configuration Register 2 (HGPCFG2)	R/W or RO	V_{CC}	00h

Host GPIO Port Runtime Register Map

The WPC8763L contains one Host GPIO port. The Host GPIO port registers are accessible to the host via the host I/O space. The registers' base address is set in the Host GPIO LDN Configuration registers (see Section 6.1.12 on page 259). The following table lists the Host GPIO port, its registers and the registers' offset addresses.

These registers are V_{CC} powered. They are set to their default values according to the value of VDDLOAD bit in HGPCFG2 register (see Host GPIO Pin Configuration Register 2 (HGPCFG2) on page 262).

Offset	Mnemonic	Register Name	Type	Reset
00h[1]	HGPDO	Host GPIO Data Out	R/W or RO	FFh
01h[1]	HGPDI	Host GPIO Data In	RO	–[2]
02h[1]	HGPEVEN	Host GPIO Event Enable	R/W	00h
03h[1]	HGPEVST	Host GPIO Event Status	R/W1C	00h

1. The location of this register is defined in Section 6.1.12 on page 259.
2. The data read from this register after reset is undefined.

Host GPIO Data Out Register (HGPDO)

Power Well: V_{CC}
Offset: 00h
Type: R/W or RO

Bit	7	6	5	4	3	2	1	0
Name	DATAOUT							
Reset	1	1	1	1	1	1	1	1

Bit	Description
7-0	**DATAOUT (Data Out).** Bits 7-0 correspond to pins 7-0 of the specific port. The value of each bit determines the value driven on the corresponding Host GPIO pin when its output buffer is enabled. Writing to the bit latches the written data unless the bit is locked by the HGPCFG1 register Lock bit. Reading the bit returns its value regardless of the pin value and configuration. 0: Corresponding pin driven to low. 1: Corresponding pin driven or released (according to buffer type selection) to high (default).

6.0 Host Modules and Host Interface (Continued)

Host GPIO Data In Register (HGPDI)

Power Well: V_{CC}
Offset: 01h
Type: RO

Bit	7	6	5	4	3	2	1	0
Name	DATAIN							
Reset	X	X	X	X	X	X	X	X

Bit	Description
7-0	**DATAIN (Data In).** Bits 7-0 correspond to pins 7-0 of the specific port. Reading each bit returns the value of the corresponding Host GPIO pin. Pin configuration and the HGPDO register value might influence the pin value. Writes are ignored. 0: Corresponding pin level low. 1: Corresponding pin level high.

Host GPIO Event Enable Register (HGPEVEN)

Power Well: V_{CC}
Offset: 02h
Type: R/W

Bit	7	6	5	4	3	2	1	0
Name	EVTENA							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**EVTENA (Event Enable).** Bits 7-0 correspond to pins 7-0 of the specific port. Each bit enables system notification by the corresponding Host GPIO pin. The bit has no effect on the corresponding status bit in HGPEVST register. 0: Event pending by corresponding Host GPIO pin masked. 1: Event pending by corresponding Host GPIO pin enabled.

GPIO Event Status Register (GPEVST)

Power Well: V_{CC}
Offset: 03h
Type: R/W1C

Bit	7	6	5	4	3	2	1	0
Name	EVTSTAT							
Reset	0	0	0	0	0	0	0	0

Bit	Description
7-0	**EVTSTAT (Event Status).** Bits 7-0 correspond to pins 7-0 of the specific port. The setting of each bit is independent of Event Enable bit in HGPEVEN register. An active event sets the status bit, which may be cleared only by software writing 1 to the bit. 0: No active edge or level detected since last cleared. 1: Active edge or level detected.

7.0 Booter Program

The WPC8763L Booter program resides in the on-chip ROM.

The Booter has these main functions:

- After reset, it performs all boot procedures.
- Then it checks the firmware (EC BIOS) in the flash and if it detects a problem, the Booter enters Recovery mode and allows debugging via JTAG (after the WPC8763L is set to OBD environment).
- Then it passes control to the firmware (EC BIOS).

The Booter initializes only the critical WPC8763L registers; the firmware must then finish the initialization process. After the WPC8763L is initialized, the firmware must enable host access to the device via LPC.

7.1 BOOTER DATA

The Booter configures the WPC8763L using data from the “header”, located in flash memory.

The header is allocated 256 bytes at the bottom of the flash, starting at offset 00h from the flash base address. This is equivalent to the core domain address range of 02 0000h-02 00FFh. Therefore, the firmware can start at any location starting at or above 02 0100h.

Table 46 shows the header contents.

Table 46. Header Contents

Offset[1]	Size	Data Item	Field/Bit					
		Name	Bits	Name	Description	Values[2]	Default[2]	Updating[3]
00h	Word	Signature	15-0	Signature	Flash Signature	8761h		No (fixed)
02h	Word	Reserved	15-0	Reserved			0000h	
04h	Byte	HCBAL_DAT	7-0	HCBA7-0	Host Configuration Base Address Low	Page 243	00h	Mandatory **only** if BADDR1-0 = 01
05h	Byte	HCBAH_DAT	7-0	HCBA15-7	Host Configuration Base Address High	Page 243	00h	
06h	Word	Reserved	15-0	Reserved			0000h	
08h	Byte	HFCGML_DAT	7-0	HFCGM7-0	HFCG, M Value, Low	Page 184	89h	Optional
09h	Byte	HFCGMH_DAT	7-0	HFCGM15-8	HFCG, M Value, High	Page 184	09h	
0Ah	Byte	HFCGN_DAT	5-0	HFCGN5-0	HFCG, N Value, Low	Page 185	02h	
0Bh	Byte	HFCGP_DAT	7-4	FPRED	Flash Clock Divider	Page 185	2h	
			3-0	CPRED	Core Clock Divider		1h	
0Ch	Word	Reserved	15-0	Reserved			0000h	
0Eh	Byte	FL_SIZE_DAT	5-0	FL_SIZE_P1	Flash Size	Page 59	21h	Mandatory
0Fh	Byte	BURST_DAT	5-4	W_BURST	Flash Write Burst	Page 59	0h	
			1-0	R_BURST	Flash Read Burst		0h	
10h	Byte	SPI_FL_DAT	7	FL_DEV	Number of Flash Dev.	Page 63	0h	Mandatory
			6	F_READ	Flash Fast Read		0h	
			5-0	DEV_SIZE	Flash Device Size		20h	
11h	Byte	MISC_CTL	7	BAD_FCKSM_IGN	Bad Firmware Checksum Ignore	Page 276	00h	Optional
12h	Word	Reserved	15-0	Reserved			0000h	
14h	DWord	FIRMW_START	31-0	FIRMW_START	Firmware Start Address	Note 4	0002 0100h	Optional
18h	DWord	FIRMW_SIZE	31-0	FIRMW_SIZE	Firmware Size (in words)	Note 5	0000 0000h	Mandatory

7.0 Booter Program (Continued)

Table 46. Header Contents (Continued)

Offset[1]	Size	Data Item	Field/Bit					
		Name	Bits	Name	Description	Values[2]	Default[2]	Updating[3]
1Ch-3Eh	Word	Reserved	15-0	Reserved			0000h	
40h	Word	HEADER_CKSM	15-0	HEADER_CKSM	Checksum of Header	Note [6]		Automatic
42h	Word	Reserved	15-0	Reserved			0000h	
44h	DWord	FIRMW_CKSM	31-0	FIRMW_CKSM	Checksum of Firmware	Note [7]		Automatic

1. Offset from the flash base. In the WPC8763L, the core address of the flash base is 02 0000h.
2. For more information on default values and on how to change them, see the *WPC876xL Software User Guide*.
3. Level of the requirement to update the default value to a value specific to the actual system.
4. Any value from 0002 0100h, to the top of the actual flash, aligned to DWord size.
5. The number of words equal to the actual firmware size.
6. 8-bit checksum of the header (flash offsets 00h-3Eh), calculated modulo 16-bit.
7. 8-bit checksum of the firmware (Firmware Size number of Words, starting from Firmware Start Address), calculated modulo 32-bit.

7.1.1 MISC_CTL Data Item

The contents of the Miscellaneous Control (MISC_CTL) data item (byte at offset 11h) are as follows:

Bit	Description
7	**BAD_FCKSM_IGN (Bad Firmware Checksum Ignore).** When this bit is set to 1, the Booter transfers control to the firmware, regardless of the status of the firmware checksum (see "Cold Boot," on page 277). Note that this bit does **not** affect Booter behavior if an incorrect signature or an incorrect header checksum is detected. 0: Enter Recovery mode, upon detection of an incorrect firmware checksum. (default) 1: Transfer control to the firmware, ignoring the status of the firmware checksum.
6-0	**Reserved.**

7.2 BOOTER OPERATION

7.2.1 Boot Types

The Booter supports two boot types: Cold and Warm. The Booter decides which boot type to perform by detecting the type of the last reset operation that affected the WPC8763L.

Reset Type Detection. After getting out of reset, the Booter checks the internal status bits that are affected by the different reset types. Based on the state of these bits, the Booter detects the type of the last reset operation as follows (see also Section 3.2 on page 48):

- V_{CC} Power-Up reset: If none of the other reset types was detected.
- $\overline{\text{VCC\_POR}}$ Input reset: If VCC_POR_STS bit in RSTCTL register is set to 1.
- Watchdog Cold reset: If WDRSR_MODE bit in TWCFG register is set to 0 **and** WDRSR_STS bit in T0CSR register is set to 1.
- Debugger Cold reset: If DBGRST_MODE bit in RSTCTL register is set to 0 **and** DBGRST_STS bit in RSTCTL register is set to 1.
- Watchdog Warm reset: If WDRSR_MODE bit in TWCFG register is set to 1 **and** WDRSR_STS bit in T0CSR register is also set to 1.
- Debugger Warm reset: If DBGRST_MODE bit in RSTCTL register is set to 1 **and** DBGRST_STS bit in RSTCTL register is also set to 1.

7.0 Booter Program (Continued)

Cold Boot. The Booter performs a Cold boot if one of the following resets is detected:

- V_{CC} Power-Up reset
- $\overline{VCC\_POR}$ Input reset
- Watchdog Cold reset
- Debugger Cold reset

During Cold boot, the Booter performs the following (see also Table 46 on page 275):

1. Checks the Signature in the flash for the 8761h value. If the Signature is incorrect, the Booter enters Recovery mode.
2. Calculates the checksum of the header and compares the result with HEADER_CKSM value. If the checksum is incorrect, the Booter enters Recovery mode.
3. If BADDR1-0 straps are set to '01', the Booter copies HCBAL_DAT and HCBAH_DAT to HCBAL and HCBAH registers, respectively.
4. Initializes the core clock domains:
   a. Copies HFCGML_DAT and HFCGMH_DAT to HFCGML and HFCGMH registers, respectively.
   b. Copies HFCGN_DAT to HFCGN register.
   c. Copies HFCGP_DAT to HFCGP register.
5. Initializes the Flash interface:
   a. Copies FL_SIZE_DAT to FIU_CFG register.
   b. Copies BURST_DAT to BURST_CFG register.
   c. he Booter copies SPI_FL_DAT to SPI_FL_CFG register.
6. Initializes the Shared Memory:
   a. Copies bits 5-0 of FL_SIZE_DAT to bits 5-0 of FLASH_SIZE register.
7. Calculates the checksum of the firmware and compares the result with FIRMW_CKSM value.
   — If the checksum is correct, writes 1h to core register R2.
   — If the checksum is incorrect, writes 0h to core register R2; then, if BAD_FCKSM_IGN bit in MISC_CTL is 0, the Booter enters Recovery mode.
8. Transfers control to the firmware by jumping to the flash address in FIRMW_START.

Warm Boot. The Booter performs a Warm boot if a Watchdog Warm reset or Debugger Warm reset was detected.

During Warm boot, the Booter performs only items 1 and 8 of the Cold boot (i.e., it checks Signature and transfers control to the firmware).

7.2.2 Firmware Requirements

The Booter does not perform full initialization of the WPC8763L registers. Completion of the initialization is left to the firmware, for flexibility. (The firmware must clear the reset type indication bits and then enable host access to the device, via LPC.)

Reset Indication Clear. Clearing the reset indication bits depends on the type of the last reset operation, as follows:

- V_{CC} Power-Up reset: No action necessary.
- $\overline{VCC\_POR}$ Input reset: No action necessary; the VCC_POR_STS bit is cleared to 0 by the next V_{CC} Power-Up reset.
- Watchdog Cold reset: Write 1 to WDRST_STS bit (this clears the bit to 0).
- Debugger Cold reset: Write 1 to DBGRST_STS bit (this clears the bit to 0).
- Watchdog Warm reset: Write 1 to WDRSR_STS bit (this clears the bit to 0).
- Debugger Warm reset: Write 1 to DBGRST_STS bit (this clears the bit to 0).

Host Access Enable. After initialization is completed, write 1 to HOSTWAIT bit in SMC_CTL register. This clears the bit to 0 and releases the LPC interface from the Long Wait SYNC (Ready SYNC is generated), thus enabling host access to the WPC8763L modules.

8.0 Device Specifications

8.1 GENERAL DC ELECTRICAL CHARACTERISTICS

8.1.1 Recommended Operating Conditions

Symbol	Parameter	Min	Typ	Max	Unit
V_{DD}	Host Domain Supply Voltage	3.0	3.3	3.6	V
V_{CC}	Core Domain Supply Voltage	3.0	3.3	3.6	V
AV_{CC}	Analog Supply Voltage	3.135	3.3	3.465	V
V_{OFF}	V_{DD}, V_{CC} and AV_{CC} Power Off Voltage[1]	–0.3	0	+0.5	V
T_A	Operating Temperature	0		+70	°C

1. Not fully tested; characterized only.

8.1.2 Absolute Maximum Ratings

Storage Temperature: –65°C to +150°C

Temperature Under Bias: 0°C to +70°C

Absolute maximum ratings are values beyond which damage to the device may occur. Unless otherwise specified, all voltages are relative to ground.

Symbol	Parameter	Conditions	Min	Max	Unit
V_{SUP}	Supply Voltage[1]		–0.5	+4.0	V
V_{REF}	Reference Voltage		–0.5	+4.0	V
V_I	Input Voltage	Buffer types: IN_{AD}, IN_{OSC}, IN_{PCI}; also IN_{TS}, IN_T multiplexed with them	–0.5	V_{SUP}[2] + 0.5	V
		All other buffer types	–0.5	5.5	V
V_O	Output Voltage	Buffer types: O_{DA}, O_{OSC}, O_{PCI}; also IN_{PCI}, $O_{p/n}$ multiplexed with them	–0.5	V_{SUP}[2] + 0.5	V
		All other buffer types	–0.5	5.5	V
I_{SINK}	Total device Sink Current[3]	Total of all output pins			mA
	ESD Tolerance	C_{ZAP} = 100 pF, R_{ZAP} = 1.5 KΩ[4]	2000		V
T_{STG}	Storage Temperature		–65	+150	°C
P_D	Power Dissipation			1	W
T_L	Lead Temperature Soldering	Soldering time < 20s		+260	°C

1. V_{SUP} is V_{DD}, V_{CC} or AV_{CC}.
2. V_{SUP} is V_{DD}, V_{CC} or AV_{CC} according to the power well of the input or output, respectively.
3. Not fully tested; characterized only.
4. Value based on test complying with RAI-5-048-RA human body model ESD testing.

8.0 Device Specifications (Continued)

8.1.3 Capacitance

Symbol	Parameter	Conditions	Min[2]	Typ[1]	Max[2]	Unit
C_{IO}	I/O Pin Capacitance	All other pins		10	15	pF
C_{INC}	LPC Clock Input Capacitance	LCLK (IN_{PCI} buffer type)	5	8	12	pF
C_{PCI}	LPC Pin Capacitance	All the pins with IN_{PCI}/O_{PCI} buffer type		8	10	pF

1. T_A = 25°C; f = 1 MHz.
2. Not fully tested; characterized only.

8.1.4 Power Supply Current Consumption under Recommended Operating Conditions

Symbol	Parameter	Power Mode	Conditions[1]	Typ[2]	Max[2]	Unit
I_{DD}	V_{DD} Average Supply Current	Active	V_{IL} = 0.5V, V_{IH} = 2.4V	2		mA
I_{DDLP}	V_{DD} Quiescent Supply Current in Low Power Mode[3]	Modules Disabled	V_{IL} = GND, V_{IH} = V_{DD}	0.8	1	mA
I_{CC}	V_{CC} and AV_{CC} Active Supply Current	Active	t_{CLK} = 250 ns	10		mA
			t_{CLK} = 40 ns	25	40	mA
I_{CCW}	V_{CC} and AV_{CC} Active Executing WAIT, Supply Current		t_{CLK} = 250 ns	7		mA
			t_{CLK} = 40 ns	15		mA
I_{CCI}	V_{CC} and AV_{CC} Deep Idle Mode Supply Current	Deep Idle	V_{IL} = GND, V_{IH} = V_{CC}, V_{DD} = 0V	0.4	0.8	mA

1. Unless stated otherwise, all parameters are specified for 0°C ≤ T_A ≤ 70°C, V_{DD}, AV_{CC} and V_{CC} = 3.0V - 3.6V and no resistive load.
2. Not fully tested; characterized only.
3. All the host domain modules disabled; no LPC bus activity.

8.1.5 Voltage Thresholds

Symbol	Parameter[1]	Min[2]	Typ	Max	Unit
V_{DDON}	V_{DD} Detected as Power-On	2.3		2.9	V
V_{CCON}	V_{CC} Detected as Power-On	2.3		2.9	V

1. All parameters specified for 0°C ≤ T_A ≤ 70°C.
2. Not fully tested; characterized only.

8.2 DC CHARACTERISTICS OF PINS BY I/O BUFFER TYPES

The tables in this section summarize the DC characteristics of all device pins described in Chapter 2.4 on page 25. The characteristics describe the general I/O buffer types defined in Table 2 on page 23. For exceptions, see Section 8.2.11 on page 283. For the DC characteristics of the analog pins (IN_{AD} and IN_{DA}), see Section 8.4 on page 285. The DC characteristics of the LPC interface meet the *PCI Local Bus Specification (Rev 2.2 December 18, 1998)* for 3.3V DC signaling.

8.2.1 Input, CMOS Compatible with Schmitt Trigger

Symbol: IN_{CS}

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IH}	Input High Voltage		0.75 V_{SUP}[1]	5.5	V
V_{IL}	Input Low Voltage		–0.5	1.1	V

8.0 Device Specifications (Continued)

Symbol	Parameter	Conditions	Min	Max	Unit
V_H	Input Hysteresis		500[2]		mV
I_{ILK}[3]	Input Leakage Current	V_{SUP} = 3.0V - 3.6V and 0 < V_{IN} < V_{SUP}		±2	µA
		V_{SUP} = 3.0V - 3.6V and V_{SUP} < V_{IN} < 5.5V[4]		10[5]	µA

1. V_{SUP} is V_{DD} or V_{CC} according to the power well of the input.
2. Not tested; based on design simulation.
3. For additional conditions, see Section 8.2.11 on page 283.
4. Only if **all** the buffers of the specific pin are back-drive protected and 5V tolerant.
5. Not fully tested; characterized only.

8.2.2 Input, SMBus Compatible

Symbol: IN_{SM}

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IH}	Input High Voltage		1.75	5.5	V
V_{IL}	Input Low Voltage		–0.5	0.8	V
I_{ILK}[1]	Input Leakage Current	V_{CC} = 3.0V - 3.6V and 0 < V_{IN} < V_{CC}		±2	µA
		V_{CC} = 3.0V - 3.6V and V_{CC} < V_{IN} < 5.5V[2]		10[3]	µA

1. For additional conditions, see Section 8.2.11 on page 283.
2. Only if **all** the buffers of the specific pin are back-drive protected and 5V tolerant.
3. Not fully tested; characterized only.

8.2.3 Input, TTL Compatible

Symbol: IN_T

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IH}	Input High Voltage		2.0	5.5	V
V_{IL}	Input Low Voltage		–0.5	0.8	V
I_{ILK}[1]	Input Leakage Current	V_{SUP}[2] = 3.0V - 3.6V and 0 < V_{IN} < V_{SUP}		±2	µA
		V_{SUP} = 3.0V - 3.6V and V_{SUP} < V_{IN} < 5.5V[3]		10[4]	µA

1. For additional conditions, see Section 8.2.11 on page 283.
2. V_{SUP} is V_{DD} or V_{CC} according to the power well of the input.
3. Only if **all** the buffers of the specific pin are back-drive protected and 5V tolerant.
4. Not fully tested; characterized only.

8.2.4 Input, TTL Compatible, with Schmitt Trigger

Symbol: IN_{TS}

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IH}	Input High Voltage	5V tolerant pins[1]	2.0	5.5	V
		Pins without 5V tolerance[1]	2.0	V_{SUP}[2]+0.5	V
V_{IL}	Input Low Voltage		–0.5	0.8	V
V_H	Input Hysteresis		250[3]		mV
I_{ILK}[4]	Input Leakage Current	V_{SUP} = 3.0V - 3.6V and 0 < V_{IN} < V_{SUP}		±2	µA
		V_{SUP} = 3.0V - 3.6V and V_{SUP} < V_{IN} < 5.5V[5]		10[6]	µA

8.0 Device Specifications (Continued)

1. See Section 8.1.2 on page 278.
2. V_{SUP} is V_{DD} or V_{CC} according to the power well of the input.
3. Not tested; based on design simulation.
4. For additional conditions, see Section 8.2.11 on page 283.
5. Only if **all** the buffers of the specific pin are back-drive protected and 5V tolerant.
6. Not fully tested; characterized only.

8.2.5 Input, PCI 3.3V Compatible

Symbol: IN_{PCI}

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IH}	Input High Voltage		0.5 V_{DD}	V_{DD}+0.5	V
V_{IL}	Input Low Voltage		–0.5	0.3 V_{DD}	V
I_{ILK}[1]	Input Leakage Current	V_{DD} = 3.0V - 3.6V and 0 < V_{IN} < V_{DD}		±2	µA

1. For additional conditions, see Section 8.2.11 on page 283.

8.2.6 Input, Analog Reference

Symbol: IN_{REF}

Symbol	Parameter	Conditions	Min	Max	Unit
V_{IN}	Input Voltage Range[1]		TBD	3.3V[2]	V
I_{IN}	Input Current	V_{INMIN} < V_{IN} < V_{INMAX}		±2	µA

1. Not fully tested; characterized only.
2. V_{IN} must be lower than or equal to AV_{CC}.

8.0 Device Specifications (Continued)

8.2.7 Output, TTL/CMOS Compatible, Push-Pull Buffer

Symbol: $O_{p/n}$

Output, TTL/CMOS Compatible, rail-to-rail push-pull buffer that is capable of sourcing *p* mA and sinking *n* mA.

Symbol	Parameter	Conditions	Min	Max	Unit
V_{OH}	Output High Voltage	$I_{OH} = -p$ mA	2.4		V
		$I_{OH} = -50$ µA	V_{SUP}[1] – 0.2		V
V_{OL}	Output Low Voltage	$I_{OL} = 20$ mA[2]		0.7	V
		$I_{OL} = n$ mA		0.4	V
		$I_{OL} = 50$ µA		0.2	V
I_{OLK}[3]	Output Leakage Current	$V_{SUP} = 3.0V$ - $3.6V$ and $0 < V_{IN} < V_{SUP}$		±2	µA
		$V_{SUP} = 3.0V$ - $3.6V$ and $V_{SUP} < V_{IN} < 5.5V$[4]		10[5]	µA

1. V_{SUP} is V_{DD} or V_{CC} according to the power well of the input.
2. Applicable only to $O_{2/20}$ buffer type.
3. For additional conditions, see Section 8.2.11 on page 283.
4. Only if **all** the buffers of the specific pin are back-drive protected and 5V tolerant.
5. Not fully tested; characterized only.

8.2.8 Output, TTL/CMOS Compatible, Open-Drain Buffer

Symbol: OD_n

Output, TTL/CMOS-compatible open-drain output buffer capable of sinking *n* mA. Output from these signals is open-drain and is never forced high.

Symbol	Parameter	Conditions	Min	Max	Unit
V_{OL}	Output Low Voltage	$I_{OL} = n$ mA		0.4	V
		$I_{OL} = 50$ µA		0.2	V
I_{OLK}[1]	Output Leakage Current	V_{SUP}[2] $= 3.0V$ - $3.6V$ and $0 < V_{IN} < V_{SUP}$		±2	µA
		$V_{SUP} = 3.0V$ - $3.6V$ and $V_{SUP} < V_{IN} < 5.5V$[3]		10[4]	µA

1. For additional conditions, see Section 8.2.11 on page 283.
2. V_{SUP} is V_{DD} or V_{CC} according to the power well of the input.
3. Only if **all** the buffers of the specific pin are back-drive protected and 5V tolerant.
4. Not fully tested; characterized only.

8.2.9 Output, PCI 3.3V Compatible

Symbol: O_{PCI}

Symbol	Parameter	Conditions	Min	Max	Unit
V_{OH}	Output High Voltage	$I_{out} = -500$ µA	0.9 V_{DD}		V
V_{OL}	Output Low Voltage	$I_{out} = 1500$ µA		0.1 V_{DD}	V
I_{OLK}[1]	Output Leakage Current	$V_{DD} = 3.0V$ - $3.6V$ and $0 < V_{IN} < V_{DD}$		±2	µA

1. For additional conditions, see Section 8.2.11 on page 283.

8.0 Device Specifications (Continued)

8.2.10 Leakage Current

Symbol	Parameter	Conditions	Typ	Max	Unit
I_{LKTOT}	Total leakage of all device pins	V_{SUP}[1] = 3.0V - 3.6V and 0 < V_{PIN} < V_{SUP}	-	20[2]	µA
I_{LKTOT5}	Total leakage of all 5V-tolerant pins	V_{SUP}[1] = 3.0V - 3.6V and V_{SUP} < V_{PIN} < 5.5V	-	20[2]	µA
I_{BD}	Leakage of back-drive protected input and output pins	V_{SUP}[1] = 0V and V_{PIN} < 5.5V	-	10[2]	µA

1. V_{SUP} is V_{DD} or V_{CC}, according to the power well each pin.
2. Not fully tested; characterized only.

8.2.11 Notes and Exceptions

1. Only pins noted with "✔" in the "5VT" column in sections Section 2.4.1 to Section 2.4.16. are back-drive protected and 5V tolerant. None of the other pins are back-drive protected or 5V tolerant.
2. I_{ILK} and I_{OLK} are measured in the following cases (where applicable):
   - Internal pull-up or pull-down resistor is disabled
   - Push-pull output buffer is disabled (TRI-STATE mode)
   - Open-drain output buffer is at high level
3. I_{ILK} and I_{OLK} are not cumulative per pin. This means that for pins having multiple buffer types (such as different types of input buffers or input/output buffers), the leakage current is the maximum caused by the relevant buffer types, at the given supply voltage and pin voltage.
4. Some pins have an internal static pull-up resistor (when enabled) and therefore may have leakage current from V_{SUP} (when V_{IN} = 0). See Section 2.5 on page 32 for a list of the relevant pins.
5. Some pins have an internal static pull-down resistor (when enabled) and therefore may have leakage current to GND (when V_{IN} = V_{SUP}). See Section 2.5 on page 32 for a list of the relevant pins.
6. The following strap pins have an internal static pull-up resistor enabled during Power-Up reset and therefore may have leakage current from V_{CC} (when V_{IN} = 0): BADDR1-0, SHBM, $\overline{\text{TRIS}}$, $\overline{\text{JEN0}}$, $\overline{\text{JENK}}$.
7. I_{OH} is valid for a GPIO pin only when it is not configured as open-drain.

8.2.12 Terminology

Back-Drive Protection. Back-drive protected pins sustain any voltage within the specified voltage limits when the device power supply is off.

5-Volt Tolerance. 5V tolerant pins sustain 5V even if the applied voltage is above the device power supply voltage. A pin is 5V tolerant in the following conditions (where applicable):

- Internal pull-up or pull-down resistor is disabled
- Push-pull output buffer is disabled (TRI-STATE mode)

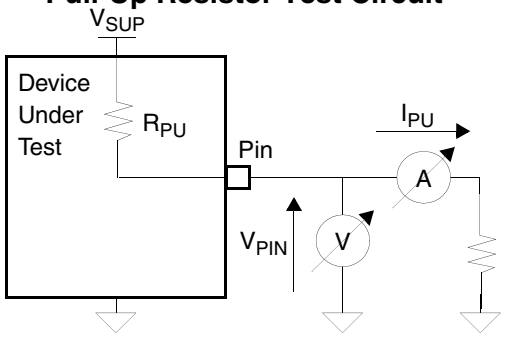
Note: If a pin has multiple buffers, the lowest "maximum voltage" among the buffers is the "maximum voltage" allowed to be applied to the pin.

8.0 Device Specifications (Continued)

8.3 INTERNAL RESISTORS

DC Test Conditions

Pull-Up Resistor Test Circuit



Pull-Down Resistor Test Circuit

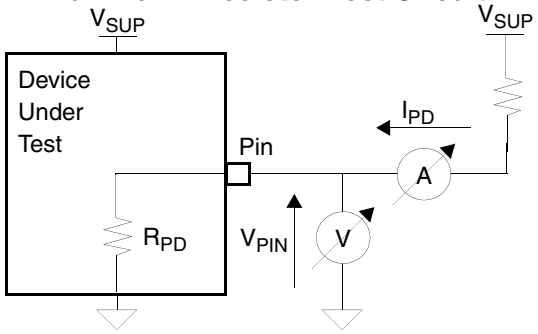
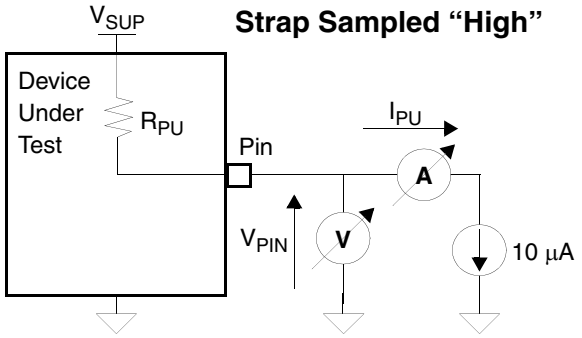


Figure 67. Internal Resistor Test Conditions, T_A = 0°C to 70°C, V_{SUP} = 3.3V

Internal Pull-Up Strap

Strap Sampled "High"



Strap Sampled "Low"

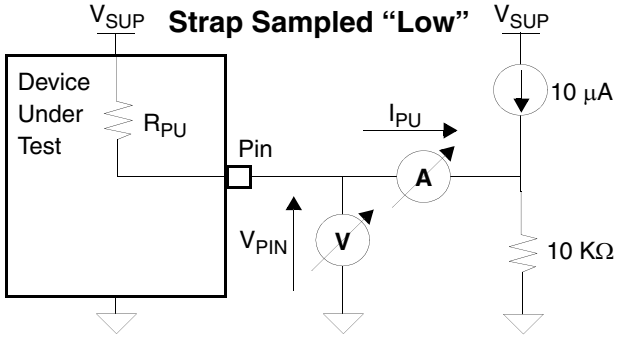


Figure 68. Internal Resistor Design Requirements, T_A = 0°C to 70°C, V_{SUP} = 3.3 V

Notes:
1. V_{SUP} is V_{DD} or V_{CC} according to the pin power well.
2. The equivalent resistance of the pull-up resistor is calculated by $R_{PU} = (V_{SUP} - V_{PIN}) / I_{PU}$.
3. The equivalent resistance of the pull-down resistor is calculated by $R_{PD} = V_{PIN} / I_{PD}$.

8.3.1 Pull-Up Resistors

Symbol: PU_{nn}

Symbol	Parameter[1,2]	V_{PIN} = 0V		V_{PIN} = 0.17 V_{SUP}	V_{PIN} = 0.8 V_{SUP}	Unit
		Min	Max	Min	Max	
R_{PU80}	PU_{80} equivalent resistance	57	107	48	43	KΩ
R_{PU30}	PU_{30} equivalent resistance	22	43.5	18.5	15.5	KΩ
$R_{PU1.25}$	$PU_{1.25}$ equivalent resistance	0.9	1.7	0.8	0.8	KΩ

1. TA = 0°C to 70°C, V_{SUP} = 3.3V.
2. Not fully tested; characterized only.

8.3.2 Pull-Down Resistors

Symbol: PD_{nn}

Symbol	Parameter[1,2]	V_{PIN} = V_{SUP}		V_{PIN} = 0.8 V_{SUP}	V_{PIN} = 0.17 V_{SUP}	Unit
		Min	Max	Max	Min	
R_{PD80}	PD_{80} equivalent resistance	45	117	99	21.5	KΩ
R_{PD30}	PD_{30} equivalent resistance	21	39.5	32	9	KΩ

1. TA = 0°C to 70°C, V_{SUP} = 3.3V.
2. Not fully tested; characterized only.

8.0 Device Specifications (Continued)

8.4 ANALOG CHARACTERISTICS

8.4.1 ADC Characteristics

Parameter	Symbol	Conditions[1]	Min	Typ	Max	Unit
Resolution	RES			8		Bit
Offset External Inputs[2]	OFS_E	V_{REF} = 3V, 0.05 V ≤ V_{IN} ≤ 2.8 V			±2.5[3]	LSB
Full Scale Error External Inputs[4]	$FSER_E$	V_{REF} = 3V, 0.05 V ≤ V_{IN} ≤ 2.8 V			±2.5[3]	LSB
Integral Non-linearity External Inputs[5]	INL_E	V_{REF} = 3V, 0.05 V ≤ V_{IN} ≤ 2.8 V			±2.3[3]	LSB
Accuracy External Inputs[6]	ACU_E	V_{REF} = 3V, 0.05 V ≤ V_{IN} ≤ 2.8 V			±3	LSB
Full Scale Voltage External Inputs	V_{FSE}			V_{REF}		V
Input Voltage Range External Inputs[7]	V_{INE}		0		V_{REF}	V
External Inputs Leakage Current	I_{ALE}	External Inputs; 0 ≤ V_{IN} ≤ AV_{CC}			±1	µA
External Inputs Resistance	R_{AINE}	External Inputs; 0 ≤ V_{IN} ≤ V_{REF}	4			MΩ
Input Capacitance	C_{AIN}	External Inputs		50	60	pF
ADC Clock Frequency	F_{ACLK}		1.60	2	2.25	MHz
Channel Conversion Duration	t_{CC}	At F_{ACLK} = 2 MHz			125	µs
ADC Activation Delay[8]	t_{END}	At F_{ACLK} = 2 MHz			1	ms

1. All parameters specified for 0°C ≤ T_A ≤ 70°C and AV_{CC}= 3.135V - 3.456V unless otherwise specified.
2. The difference between 0V and the actual voltage value for code 00h (extrapolated from the range specified in “Conditions”).
3. Not fully tested; characterized only.
4. The difference between $^{255}/_{256}$ * V_{REF} and the actual voltage for code FFh (extrapolated from the range specified in “Conditions”).
5. The maximum difference between the ideal (straight) conversion line and the actual conversion curve, not including the offset, gain and quantization (±0.5 LSB) errors.
6. Total unadjusted error (includes the offset, full-scale error, integral non-linearity and quantization (±0.5 LSB) error).
7. Input Voltage recommended for measurement. Voltage levels outside this range (but within the limits specified in Section 8.1.2 on page 278) are allowed. However, in this case measurement accuracy is not guaranteed.
8. Time from the moment when either START is set to 1 **or** a timer pulse is generated when ADCTTE is set to 1, until the beginning of the conversion.

8.0 Device Specifications (Continued)

8.4.2 DAC Characteristics

Parameter	Symbol	Conditions[1]	Min	Typ	Max	Unit
Resolution	RES			8		Bit
Offset[2]	OFS	0.05 V ≤ V_{OUT} ≤ AV_{CC} −0.1V, AV_{CC} = 3.3V		±1		LSB
Gain Error[3]	GER			±1.5		LSB
Integral Non-linearity[4]	INL			±1.5		LSB
Differential Non-linearity[5,6]	DNL				±2.5	LSB
Accuracy[7]	ACU				±3	LSB
Output Voltage Range	V_{OUT}	Note[8]	0		AV_{CC}	V
Output Resistance[9]	R_S	0 ≤ V_{OUT} ≤ AV_{CC}	3	4	5	KΩ
Output Capacitance[9]	C_{AO}			10	15	pF
DAC Settling Time[10]	T_{SET}	C_L = 50 pF			1.25[9]	µs
DAC Enable Delay[11]	T_{END}	C_L = 50 pF			10	µs

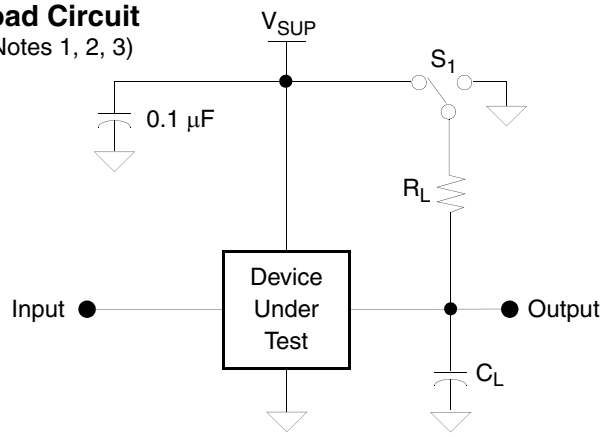
1. All parameters specified for 0°C ≤ T_A ≤ 70°C. AV_{CC} = 3.135V - 3.465V unless otherwise specified.
2. The difference between 0V and the actual voltage value for code 00h (extrapolated from the range specified in “Conditions”).
3. The difference between $^{255}/_{256}$ * AV_{CC} and the actual voltage for code FFh (extrapolated from the range specified in “Conditions”).
4. The maximum difference between the ideal (straight) conversion line and the actual conversion curve, not including the offset, gain error and quantization (±0.5 LSB) error.
5. The maximum difference between an ideal step size (1 LSB) and any actual step size.
6. Monotonic.
7. Total unadjusted error (includes the offset, gain error, integral non-linearity and quantization (±0.5 LSB) error).
8. Output Voltage allowed for normal operation. Linear range is as defined for the different parameters.
9. Not fully tested; characterized only.
10. Time from the converter loading with data, to output voltage settling within an error of ±0.5 LSB, for a full scale step (i.e., from 0.05V to AV_{CC} −0.1V).
11. Time from the moment when DACEN*n*=1 in DACCTRL register until the settling of the output voltage, in the same conditions as specified in Note 10.

8.0 Device Specifications (Continued)

8.5 AC ELECTRICAL CHARACTERISTICS

8.5.1 AC Test Conditions

Load Circuit
(Notes 1, 2, 3)



AC Testing Input, Output Waveform

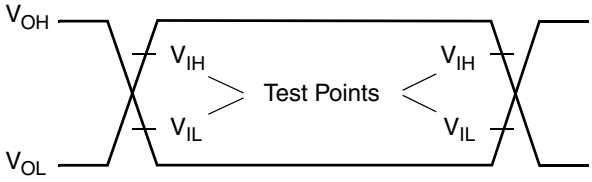


Figure 69. AC Test Conditions, T_A = 0°C to 70°C, V_{SUP} = 3.0V - 3.6V

Notes:

1. V_{SUP} is V_{DD}, V_{CC} or AV_{CC} according to the power well of the pin.
2. C_L = 50 pF for all output pins except the following pin groups (values include both jig and oscilloscope capacitance)
   – C_L = 15 pF for CLKOUT
   – C_L = 400 pF for SMBus and SensorPath pin in x1 mode; C_L = 100 pF for SensorPath pin in x4 mode
3. S_1 = Open – for push-pull output pins
   S_1 = V_{SUP} – for high impedance to active-low and active-low to high-impedance transition measurements
   S_1 = GND – for high impedance to active-high and active-high to high-impedance transition measurements
   R_L = 1.0 KΩ– for all the pins
4. The following abbreviations are used in Section 8.5: RE = Rising Edge; FE = Falling Edge

Definitions

The timing specifications in this section are relative to V_{IL} or V_{IH} (according to the specific buffer type) on the rising or falling edges of all the signals, as shown in the following figures (unless specifically stated otherwise).

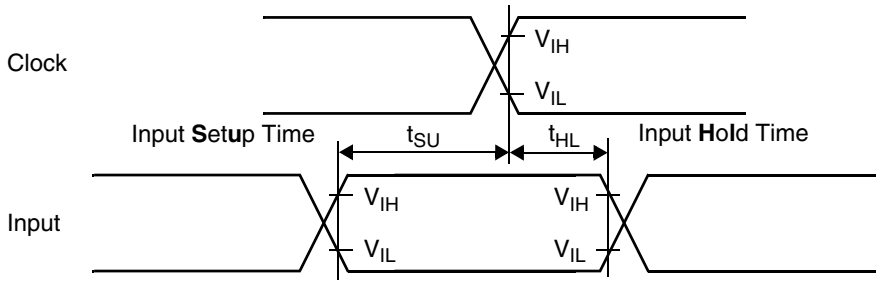


Figure 70. Input Setup and Hold Time

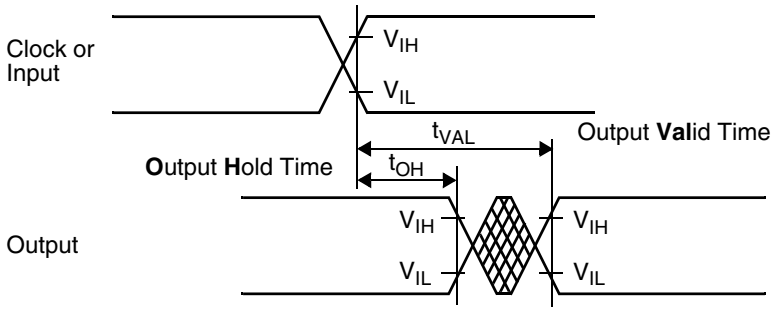


Figure 71. Clock-to-Output and Propagation Delay

8.0 Device Specifications (Continued)

8.5.2 Reset Timing

Core Domain Reset

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
$t_{RSTDLY1}$	72	Reset delay 1			t_{32KVAL}[1] + 41.4	ms
$t_{RSTDLY2}$	73	Reset delay 2			25.7	ms
t_{PORW}	72	$\overline{VCC\_POR}$ output width	After $V_{CC} > V_{CCON}$	93	t_{32KVAL}[1] + 103	ms
t_{PORR}	72	$\overline{VCC\_POR}$ input rise time[2]	From V_{IL} to V_{IH}		10	µs
t_{WPOR}	73	$\overline{VCC\_POR}$ input width	Driven by external source	61	Note [3]	µs

1. For t_{32KVAL} value, see "Low-Frequency Clock Timing" on page 290.
2. Depends on the signal capacitance (C_L) and the pull-up value (R_L).
3. $t_{RSTDLY2}$ starts after the end of t_{WPOR}.

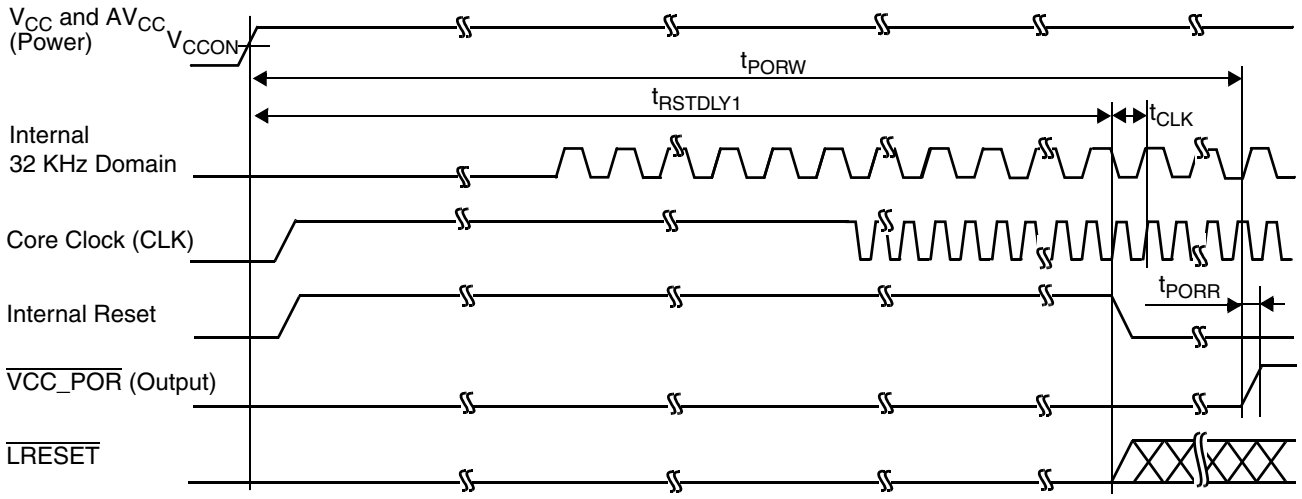


Figure 72. V_{CC} Power-Up Reset

8.0 Device Specifications (Continued)

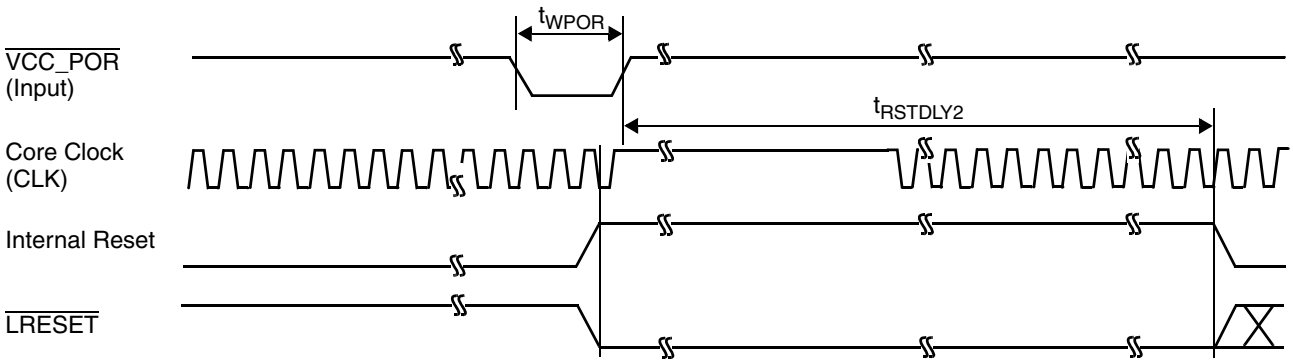


Figure 73. VCC_POR Input Reset

Host Domain Reset

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
t_{HIRST}	74	Internal Host Power-Up reset time	V_{DD} power-up to end of internal reset		t_{LRST}	–
t_{LRST}	74	LRESET active time	PS_PWR_OK[1] to end of LRESET	15		ms
t_{WLRST}	75	LRESET pulse width	From LRESET FE to LRESET RE	100		ns

1. PS_PWR_OK is a system signal not connected to WPC8763L.

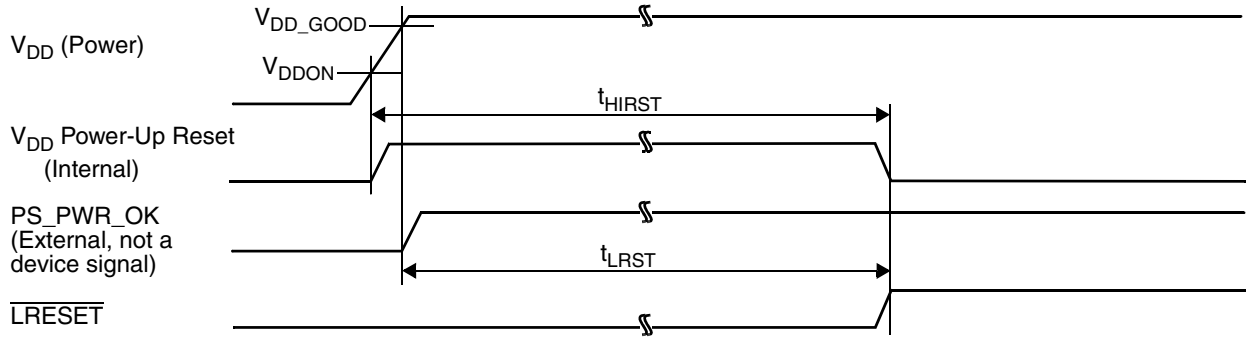


Figure 74. V_{DD} Power-Up Reset

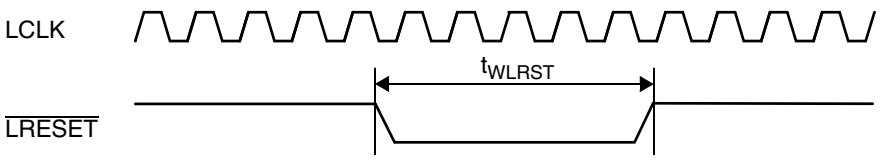


Figure 75. Host Domain Hardware Reset

8.0 Device Specifications (Continued)

8.5.3 Clock Timing

Low-Frequency Clock Timing

Symbol	Figure	Description	Reference Conditions	Min	Typ	Max	Units
Clock Input Timing							
$t_{32KCLKIN}$	–	Required clock period for 32KCLKIN[1]	From RE to RE of 32KCLKIN	30.5145 (t_{32TYP} – 100ppm)	30.517578 (t_{32TYP})	30.5206 (t_{32TYP} + 100ppm)	µs
Clock Output Timing							
t_{32KOSC}	76	Clock period of 32K oscillator[2]	From RE to RE of CLKOUT		30.517578 (t_{32TYP})		µs
t_{32KVAL}	76	32K valid time[2]	After AV_{CC} > AV_{CCON}			1	s

1. Recommended for timing accuracy.
2. Determined by the values of the external crystal circuit components.

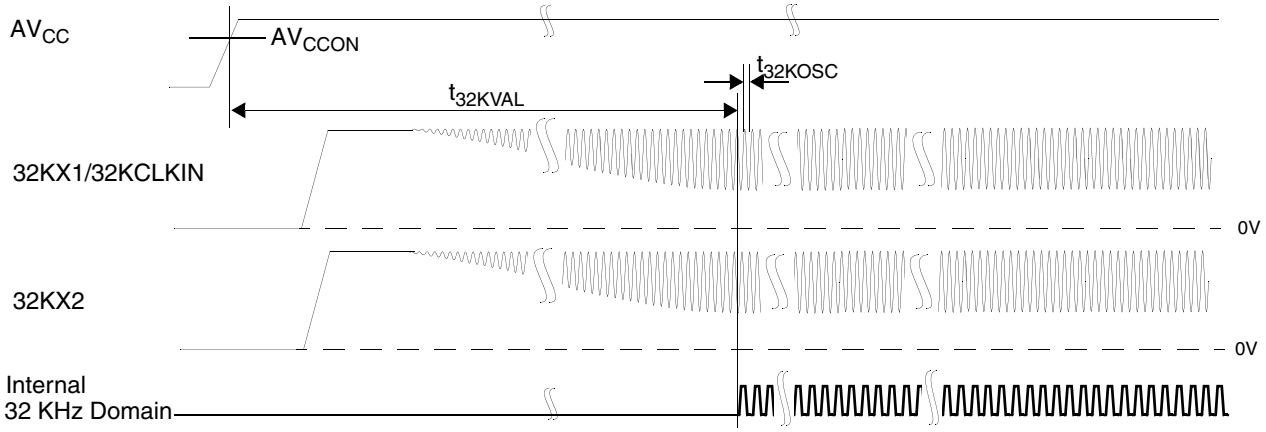


Figure 76. Low-Frequency Clock Waveforms

8.0 Device Specifications (Continued)

Core, High-Frequency Clock Timing

Symbol	Figure	Description	Reference Conditions	Min	Typ	Max	Units
t_{CLK}	77	Core clock (CLK) period	From RE to RE of CLKOUT	40		250[1]	ns
F_{CLK}	–	Core clock frequency	Set CLKOUT frequency	F_{TYP} – 1%	F_{TYP}	F_{TYP} + 1%	
t_{WUDI}	78	Wake-up from Deep Idle delay	After wake-up event			3	ms
t_{WUI}	78	Wake-up from Idle delay	After wake-up event			35	µs

1. Not tested; based on design simulation.

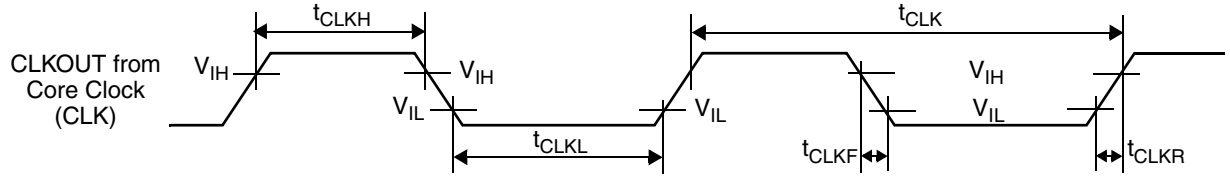


Figure 77. Core, High-Frequency Clock Waveforms

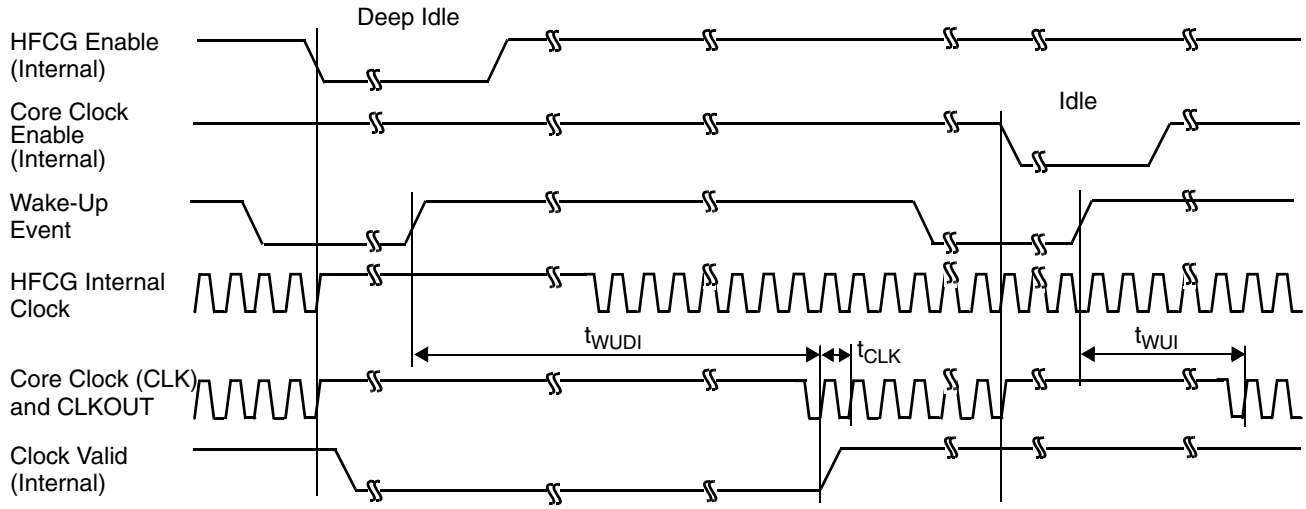


Figure 78. Core, High-Frequency Clock Wake-Up Timing

8.0 Device Specifications (Continued)

8.5.4 FIU Timing

Symbol	Figure	Description	Reference Conditions		Min	Typ	Max	Units
Clock Timing								
t_{SFCYC}	79	F_SCK cycle Time	From RE to RE	C_L = 30 pF	40		100[1]	ns
				C_L = 15 pF	20		100[1]	ns
t_{SFHIGH}, t_{SFLOW}	79	F_SCK high/low Time	At t_{SFCYC} = 40 ns	C_L = 30 pF	18			ns
			At t_{SFCYC} = 20 ns	C_L = 15 pF	9			ns
	–	F_SCK Slew Rate	From 0.2 * V_{CC} to 0.8 * V_{CC}		2[1]			V/ns
Signal Timing								
t_{SFVAL}	80	F_SDO Valid Delay	After FE of F_SCK	C_L = 30 pF			8	ns
				C_L = 15 pF			3	ns
t_{SFOHL}	80	F_SDO Hold Time	After FE of F_SCK	C_L = 15 pF	–3[2]			ns
t_{SFSU}	80	F_SDI Setup Time	Before RE of F_SCK		–2			ns
t_{SFHL}	80	F_SDI Hold Time	After RE of F_SCK		8			ns
t_{SFCHL}	81	$\overline{F\_CS0}$ Hold Time	After FE of F_SCK	C_L = 15 pF	1 * t_{SFCYC}			ns
t_{SFCSU}	81	$\overline{F\_CS0}$ Setup Time	Before RE of F_SCK	C_L = 30 pF	1.5 * t_{SFCYC} – 20 ns[2]			ns
t_{SFCSW}	81	$\overline{F\_CS0}$ Inactive Width	From RE to FE of $\overline{F\_CS0}$-1		5 * t_{SFCYC}			ns

1. Not tested; based on design simulation.
2. Not fully tested; characterized only.

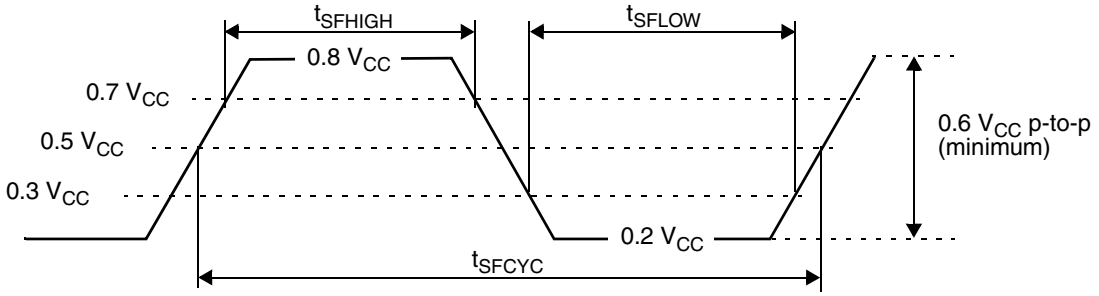


Figure 79. F_SCK Timing

8.0 Device Specifications (Continued)

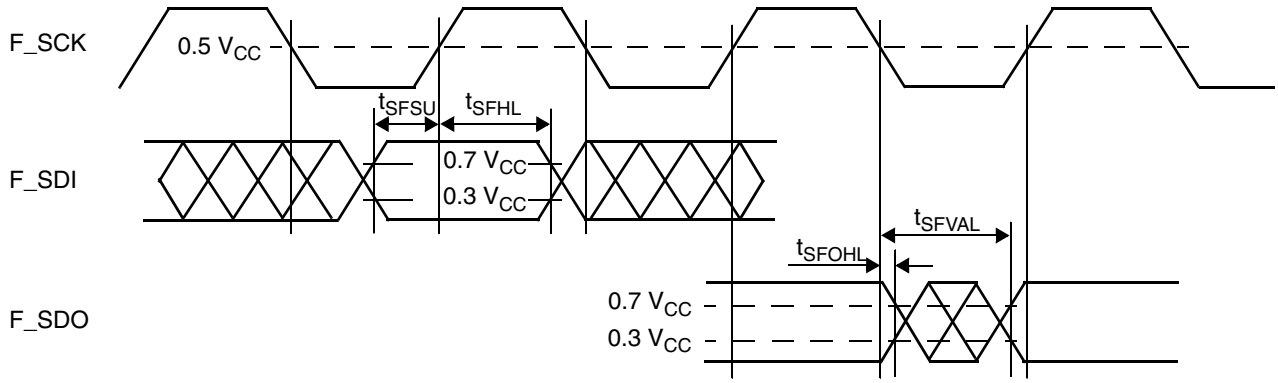


Figure 80. SPI Signal Timing

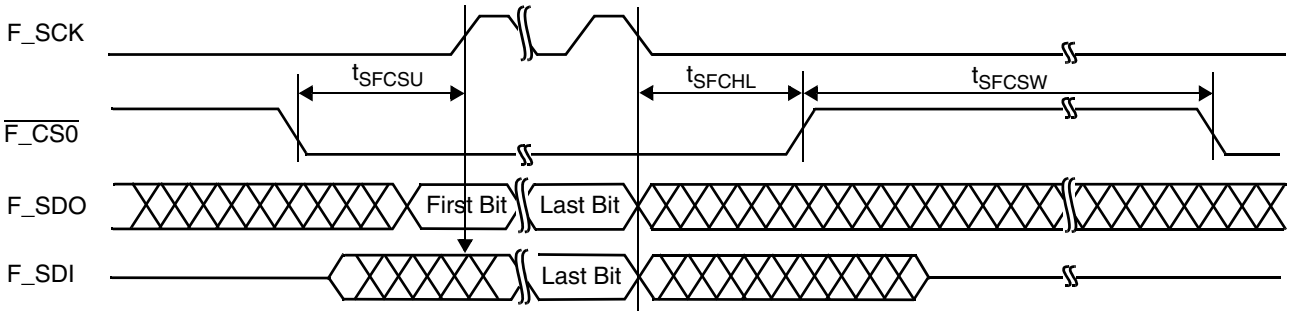


Figure 81. F_CS0 Timing

8.5.5 MIWU Input Signals Detection Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
t_{ASW}	82	Input pulse width (that guarantees detection)	GPIO01, GPIO03-07, GPIO10-14, GPIO16, GPI90-97, SCL1-2, SDA1-2, PSCLK1-3, PSDAT1-3, KBSIN0-7	30		ns

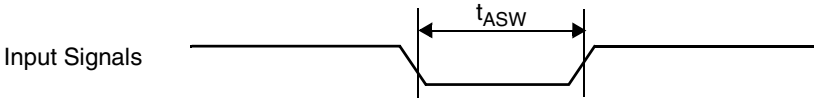


Figure 82. MIWU Input Signals Detection Timing

8.5.6 GPIO Ports and Keyboard Scan Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
t_{ECHOD}	83	Echo GPIO Delay	Combinatorial, from GPIO01, GPIO03 to GPIO21, GPIO23		100	ns

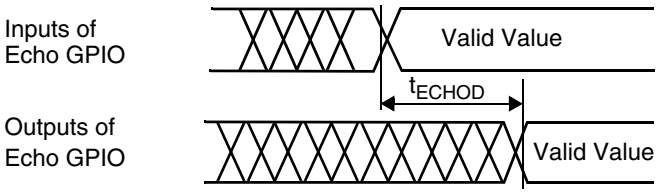


Figure 83. GPIO Ports Signal Timing

8.0 Device Specifications (Continued)

8.5.7 PS/2 Interface Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
PS/2 Input Timing						
t_{PSCLKL}	84	PSCLKn low time[1]	At V_{IL} (Both Edges)	$(n^2+8) * t_{CLK}$[3]		ns
t_{PSCLKH}	84	PSCLKn high time[1]	At V_{IH} (Both Edges)	$(n^2+8) * t_{CLK}$		ns
t_{PSDISU}	84	PSDATn input setup time	Before FE PSCLKn	0		ns
t_{PSDIH}	84	PSDATn input hold time	After RE PSCLKn	0		ns
PS/2 Output Timing						
$t_{PSDOVAL}$	85	PSDATn output valid time	After FE PSCLKn		$(n^2 + 7) * t_{CLK}$	ns
t_{PSDOH}	85	PSDATn output hold time	After FE PSCLKn	0		ns

1. Not tested; based on design simulation.
2. 'n' is the number of clock cycles, as programed in the IDB field; see "PS/2 Control Register (PSCON)" on page 97.
3. t_{CLK} is the cycle time of the core clock (CLK).

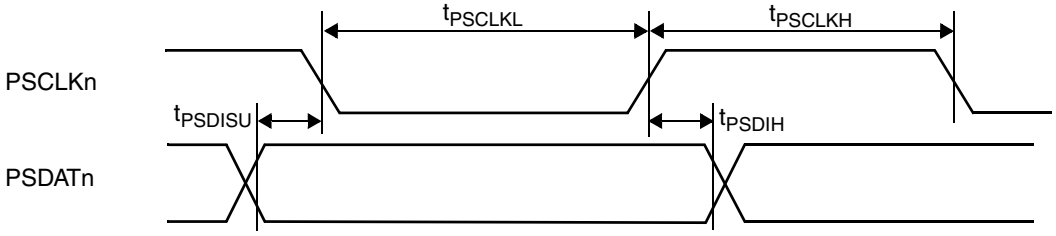


Figure 84. PS/2 Receive Timing

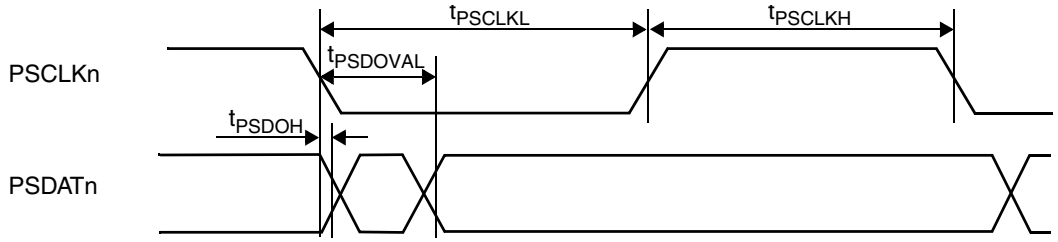


Figure 85. PS/2 Transmit Timing

8.0 Device Specifications (Continued)

8.5.8 MFT16 Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
Input Timing						
t_{TABH}	86	TA1-2, TB1 input high time	At V_{IH} (Both Edges)	t_{CLK}[1] + 20 ns		–
t_{TABL}	86	TA1-2, TB1 input low time	At V_{IL} (Both Edges)	t_{CLK} + 20 ns		–
Output Timing						
t_{OUTCYC}	87	TA1-2 output cycle time	At 50% (Both Edges)	n[2] * t_{CLK}[1] - 10 ns	n * t_{CLK} + 10 ns	–

1. t_{CLK} is the cycle time of the core clock (CLK).
2. 'n' is the number of clock cycles, as programed in Section 4.8 on page 101.

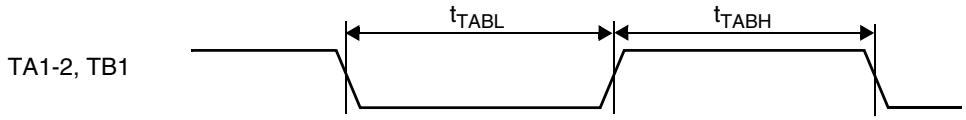


Figure 86. Multi-Function Timer (MFT16) Input Timing

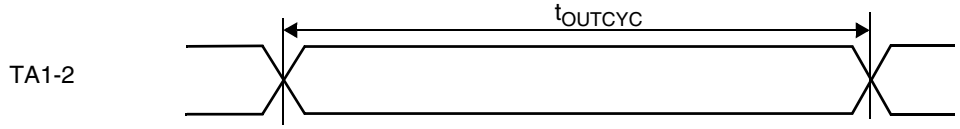


Figure 87. Multi-Function Timer (MFT16) Output Timing

8.5.9 PWM Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
t_{OUTCYC}	88	A_PWM1-0, B_PWM0 output cycle time	At 50% (Both Edges)	n[1] * t_{CLK}[2] - 10 ns	n * t_{CLK} + 10 ns	–

1. 'n' is the number of clock cycles, as programed in Section 4.9 on page 118.
2. t_{CLK} is the cycle time of the core clock (CLK).

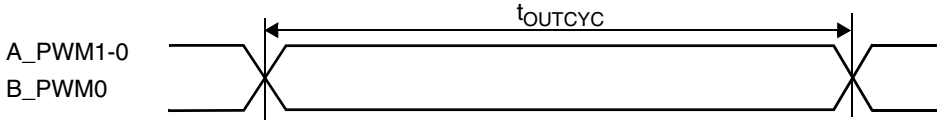


Figure 88. PWM Signal Timing

8.5.10 CR_UART Timing

Symbol	Figure	Parameter	Conditions	Min	Max	Units
t_{BT}	89	Single Bit Time[1]	Transmitter	t_{BTN}[2] – 25	t_{BTN} + 25	ns
			Receiver	t_{BTN}[2] – 2%	t_{BTN} + 2%	ns

1. Not tested; based on design simulation.
2. t_{BTN} is the nominal bit time: t_{BTN} = 1 / BR; BR is determined by the setting of the Baud Rate Generator (see "Baud Rate Generator" on page 125).

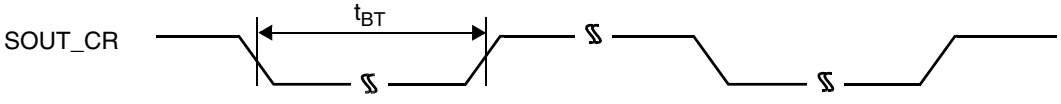


Figure 89. CR_UART Timing

8.0 Device Specifications (Continued)

8.5.11 SMBus Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
			Input Timing			
f_{SCLFI}	90	SCL frequency	At 1.2V SCL RE to RE		100	KHz
t_{SCLLI}	90	SCL low time	At 0.8V (Both Edges)	$K^1 * t_{CLK}^2$		-
t_{SCLHI}	90	SCL high time	At 1.75V (Both Edges)	$K * t_{CLK}$		-
t_{SMBRI}	90	SCL, SDA rise time	From 0.8V to 1.75V[3]		1[4]	µs
t_{SMBFI}	90	SCL, SDA fall time	From 1.75V to 0.8V[3]		300[4]	ns
t_{SDASI}	91	SDA setup time	Before SCL RE	$1 * t_{CLK}$		-
t_{SDAHI}	91	SDA hold time	After SCL FE	0		ns
t_{CSTRSI}	93	SCL setup time	Before Restart condition	$8 * t_{CLK}$		-
t_{CSTRHI}	92, 93	SCL hold time	After Start/Restart condition	$8 * t_{CLK}$		-
t_{CSTOSI}	92	SCL setup time	Before Stop condition	$8 * t_{CLK}$		-
t_{BUFI}	92	Bus free time	Between Stop and Start conditions	$8 * t_{CLK}$		-
			Output Timing			
f_{SCLFO}	90	SCL frequency	At 1.2V SCL RE to RE		100	KHz
t_{SCLLO}	90	SCL low time	At 0.8V (Both Edges)	$K^1 * t_{CLK} - t_{SMBFO}$		-
t_{SCLHO}	90	SCL high time	At 1.75V (Both Edges)	$K * t_{CLK} - t_{SMBRO}$		-
t_{SMBRO}	90	SCL, SDA rise time[4]	From 0.8V to 1.75V[3]		1[5]	µs
t_{SMBFO}	90	SCL, SDA fall time[4]	From 1.75V to 0.8V[3]		250	ns
t_{SDAHO}	91	SDA hold time[6]	After SCL FE	$7 * t_{CLK}$		-
t_{SDALVO}	91	SDA low valid time	After SCL FE		$7 * t_{CLK}$	-
t_{SDAHVO}	91	SDA high valid time	After SCL FE		$7 * t_{CLK} + t_{SMBRO} - t_{SMBFO}$	-
t_{CSTRSO}	93	SCL setup time	Before Restart condition	$(K - 1) * t_{CLK} - t_{SMBRO} + t_{SMBFO}$		-
t_{CSTRHO}	92, 93	SCL hold time	After Start/Restart condition	$(K + 1) * t_{CLK}$		-
t_{CSTOSO}	92	SCL setup time	Before Stop condition	$(K - 1) * t_{CLK}$		-
t_{BUFO}	92	Bus free time	Between Stop and Start conditions	$(K + 1) * t_{CLK} - t_{SMBRO}$		-

1. K = 2 * SCLFRQ (see "SMB Control Register 2 (SMBnCTL2)" on page 165). The minimum value for "K" is 16.
2. t_{CLK} is the cycle time of the core clock (CLK).
3. Test conditions: R_L = 1 KΩ to V_{CC} = 3.3V, C_L = 400 pF to GND.
4. Not tested; based on design simulation.
5. Depends on the signal capacitance (C_L) and the pull-up value (R_L). Must be less than 1 µs.
6. In I^2C and SMBus standards, the data hold time is defined from SCL at V_{IL}. Denoting this parameter as t_{DH}, then $t_{DH} = t_{SDAHO} - t_{SMBFO}$. If CL is high, the hold time (t_{DH}) at high core clock (CLK) frequencies is lower than the SMBus minimum requirement of 300 ns, but always meets the I^2C minimum requirement of 0 ns.

Note: In Figure 90 through Figure 93, an "O" is added to parameter names in the timing tables for output signals and an "I" for input signals, as displayed in the preceding table.

8.0 Device Specifications (Continued)

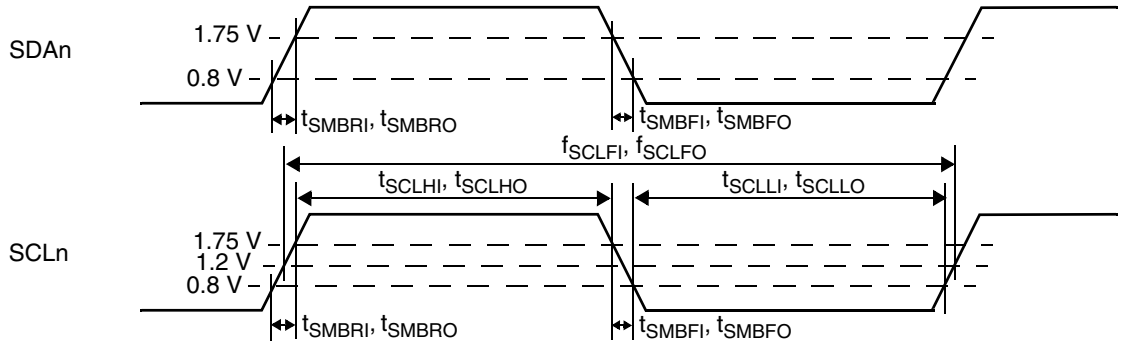


Figure 90. SMBus Signals (SDA and SCL) Timing

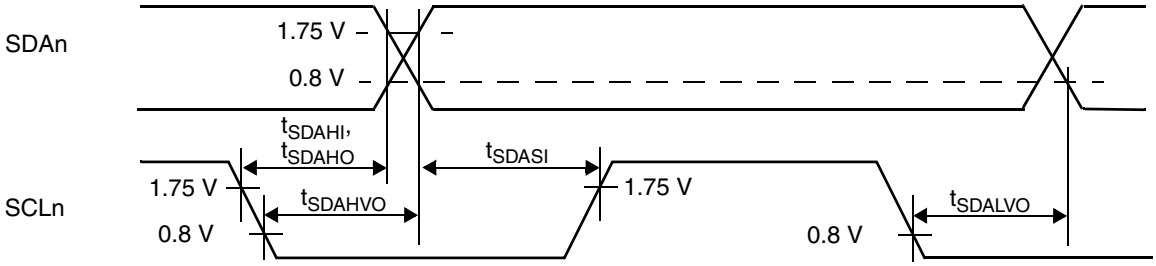


Figure 91. SMBus Data Bit Timing

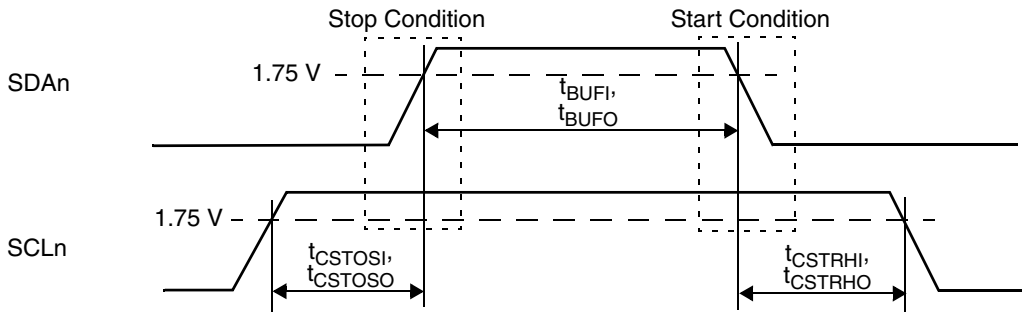


Figure 92. SMBus Start and Stop Condition Timing

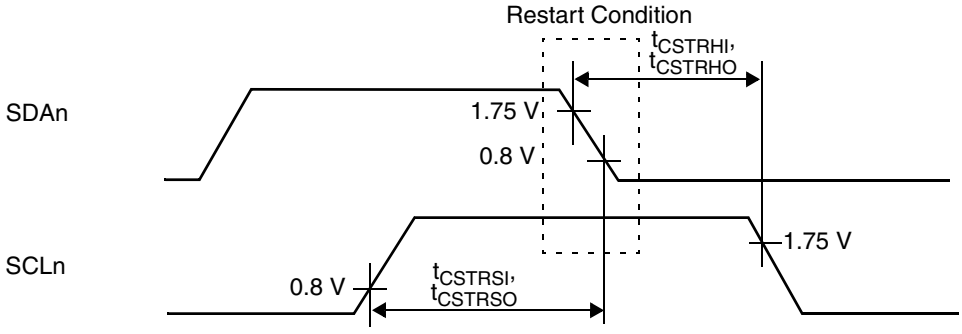


Figure 93. SMBus Restart Condition TIming

8.0 Device Specifications (Continued)

8.5.12 SensorPath Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
t_{RISE}	94	SWD Rise Time[1]	x1 Bus Speed, C_L = 400 pF		1000	ns
			x4 Bus Speed, C_L = 100 pF		250	ns
t_{FALL}	94	SWD Fall Time[1]	x1 Bus Speed, C_L = 400 pF		300	ns
			x4 Bus Speed, C_L = 100 pF		75	ns

1. Not fully tested; characterized only.

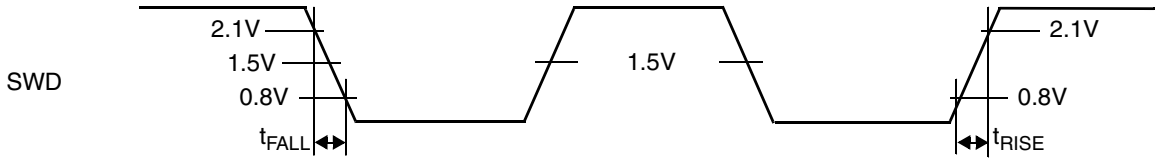


Figure 94. SensorPath Timing

8.0 Device Specifications (Continued)

8.5.13 JTAG (Debugger Interface) Timing

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
Input Signals						
t_{MSU}	95	TMS Setup Time	Before RE TCK	10		ns
t_{MHL}	95	TMS Hold Time	After RE TCK	5		ns
t_{DSU}	95	TDI Setup Time	Before RE TCK	10		ns
t_{DHL}	95	TDI Hold Time	After RE TCK	5		ns
t_{RSU}	95	$\overline{TRST}$ Setup Time	Before FE TCK	15		ns
t_{WRST}	95	$\overline{TRST}$ Pulse Width	At 1.5V (Both Edges)	15[1]		ns
t_{HW}	95	TCK High Pulse Width	At 1.5V (Both Edges)	20		ns
t_{LW}	95	TCK Low Pulse Width	At 1.5V (Both Edges)	20		ns
f_{MAX}	-	Maximum TCK Clock Frequency			20	MHz
t_{PUCC}	72	Wait Time, V_{CC} Power-Up to TCK[2,3]		$t_{RSTDLY1}$		-
t_{PUCO}	73	Wait Time, $\overline{VCC\_POR}$ Input Reset, to TCK[2,3]		$t_{RSTDLY2}$		-
Output Signals						
t_{JVAL}	95	Propagation Delay TCK to TDO	After FE TCK	–5	10	ns

1. Not fully tested; characterized only.
2. See Section 8.5.2 on page 288.
3. Not tested; based on design simulation.

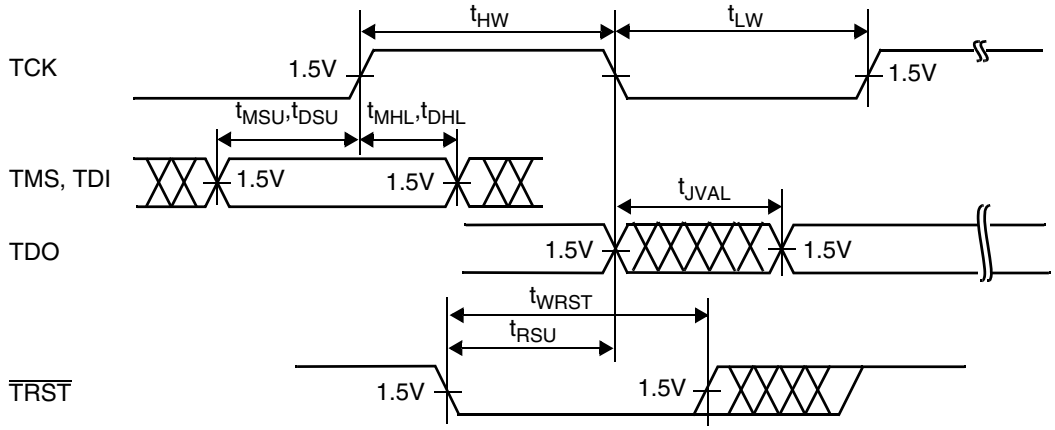


Figure 95. JTAG Signal Timing

8.5.14 LPC Interface Timing

The AC characteristics of the LPC interface meet the PCI Local Bus Specification *(Rev 2.2 December 18, 1998)* for 3.3V DC signaling.

LCLK and $\overline{LRESET}$

Symbol	Figure	Parameter	Min	Max	Units
t_{CYC}	96	LCLK Cycle Time[1]	30	33.33[2]	ns
t_{HIGH}	96	LCLK High Time	11		ns
t_{LOW}	96	LCLK Low Time	11		ns

8.0 Device Specifications (Continued)

Symbol	Figure	Parameter	Min	Max	Units
–	–	LCLK Slew Rate[2,3]	1	4	V/ns
–	–	$\overline{\text{LRESET}}$ Slew Rate[2,4]	50		mV/ns

1. LCLK may have any clock frequency between 30 MHz and 33 MHz. The clock frequency may be changed at any time during the operation of the system as long as the clock edges remain “clean” (monotonic) and the minimum cycle high and low times are not violated. The clock may be stopped only in a low state.
2. Not tested; based on design simulation.
3. Rise and fall times are specified in terms of the edge rate measured in V/ns. This slew rate must be met across the minimum peak-to-peak portion of the clock wavering (0.2 * V_{DD} to 0.6 * V_{DD}) as shown below.
4. The minimum $\overline{\text{LRESET}}$ slew rate applies only to the rising (deassertion) edge of the reset signal and ensures that system noise cannot make an otherwise monotonic signal appear to bounce in the switching range.

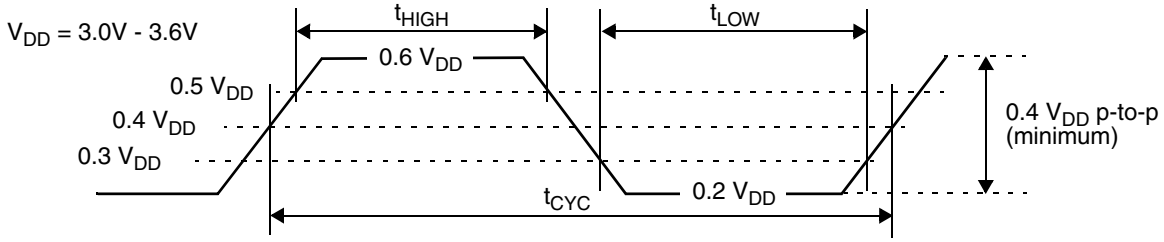


Figure 96. LCLK Waveform

LPC Signals

Symbol	Figure	Description	Reference Conditions	Min	Max	Units
t_{VAL}	97	Output Valid Delay	After RE of LCLK	2[1]	11	ns
t_{ON}	97	Float to Active Delay	After RE of LCLK	2[1]		ns
t_{OFF}	97	Active to Float Delay	After RE of LCLK		28[1]	ns
t_{SU}	98	Input Setup Time	Before RE of LCLK	7		ns
t_{HL}	98	Input Hold Time	After RE of LCLK	0		ns

1. Not fully tested; characterized only.

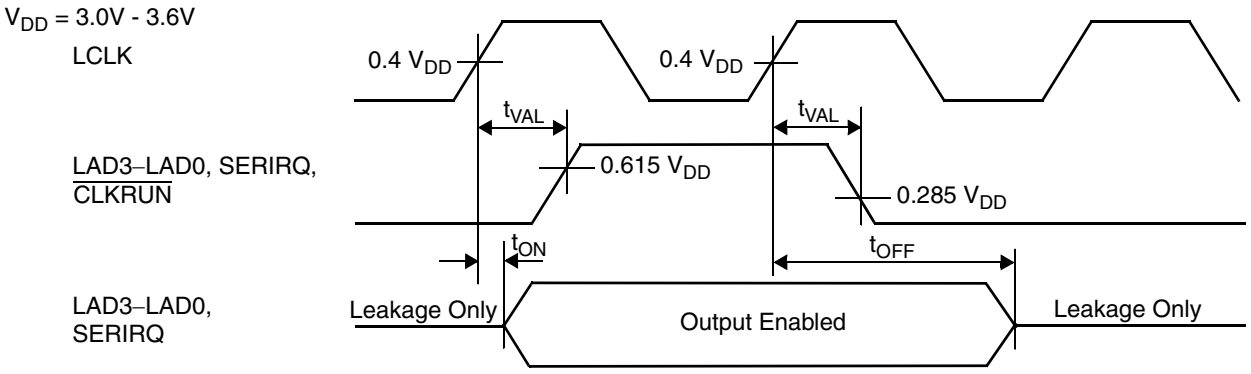


Figure 97. LPC Output Signal Timing

8.0 Device Specifications (Continued)

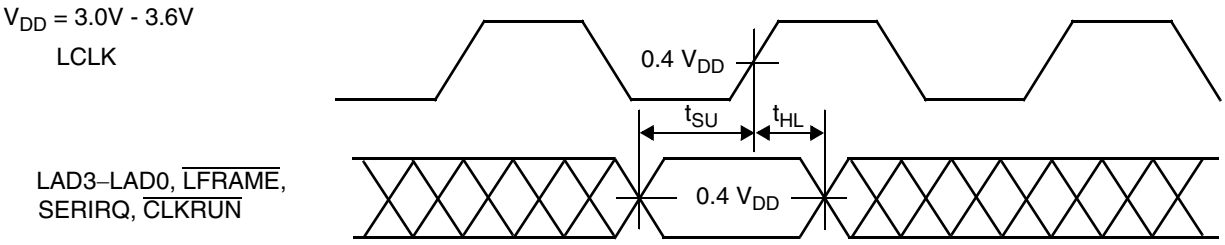


Figure 98. LPC Input Signal Timing

8.6 PACKAGE THERMAL INFORMATION

Thermal resistance (degrees C/W) $Theta_{JA}$ values for the WPC8763L package are as follows:

Table 47. Theta (Θ) J Values

Package Type	$Theta_{JA}$@0 lfpm	$Theta_{JA}$@225 lfpm	$Theta_{JA}$@500 lfpm	$Theta_{JA}$@900 lfpm
128-Pin LQFP	47	41	38.7	37.2

Notes: Airflow for $Theta_{JA}$ values is measured in linear feet per minute (lfpm).
All values apply to a device soldered to a 4-layer PCB with two signal layers and two power planes.

Appendix A. Register List

A.1 CORE DOMAIN REGISTERS

A.1.1 System Configuration

(See Section 2.6.1 on page 35.)

Address	Mnemonic	Size	Type	Reset Value	Comments
FF F000h	DEVCNT	Byte	R/W	00h	
FF F001h	STRPST	Byte	RO	0**SSS**0 0**SSS**b	'S' = Strap dependent bit
FF F002h	RSTCTL	Byte	Varies per bit	00h	
FF F010h	DEVALT0	Byte	Varies per bit	00**S**0 0000b	'S' = Strap dependent bit
FF F011h	DEVALT1	Byte	R/W	00h	
FF F012h	DEVALT2	Byte	R/W	00h	
FF F013h	DEVALT3	Byte	R/W	00h	
FF F014h	DEVALT4	Byte	R/W	00h	
FF F015h	DEVALT5	Byte	R/W	00h	
FF F0**35**h	DEVALT6	Byte	R/W	03h	
FF F016h	Reserved				
FF F017h	DEVALT7	Byte	R/W	3Ch	
FF F018h	DEVALT8	Byte	R/W	00h	
FF F019h	DEVALT9	Byte	R/W	00h	
FF F020h	Reserved				
FF F021h	GES1	Byte	R/W	00h	
FF F028h	DEVPU0	Byte	R/W	00h	
FF F029h	DEVPU1	Byte	R/W	00h	

A.1.2 Flash Interface Unit (FIU)

(See Section 4.1.6 on page 58.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF 0000h	FIU_CFG	Byte	R/W or RO	21h	
FF 0001h	BURST_CFG	Byte	R/W	00h	
FF 0002h	RESP_CFG	Byte	R/W	00h	
FF 0003h	CFBB_PROT	Byte	R/W1S	00h	
FF 0004h	FWIN1_LOW	Word	R/W or RO	0000h	
FF 0006h	FWIN1_HIGH	Word	R/W or RO	0000h	
FF 0008h	FWIN2_LOW	Word	R/W or RO	0000h	
FF 000Ah	FWIN2_HIGH	Word	R/W or RO	0000h	
FF 000Ch	FWIN3_LOW	Word	R/W or RO	0200h	
FF 000Eh	FWIN3_HIGH	Word	R/W or RO	0200h	
FF 0010h	PROT_LOCK	Byte	R/W1S	00h	
FF 0011h	PROT_CLEAR	Byte	WO	–	
FF 0012h-FF 0013h	Reserved				

Appendix A. (Continued)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF 0014h	SPI_FL_CFG	Byte	R/W or RO	20h	
FF 0015h	Reserved				
FF 0016h	UMA_CODE	Byte	R/W	00h	
FF 0017h	UMA_AB0	Byte	R/W	00h	
FF 0018h	UMA_AB1	Byte	R/W	00h	
FF 0019h	UMA_AB2	Byte	R/W	00h	
FF 001Ah	UMA_DB0	Byte	R/W	00h	
FF 001Bh	UMA_DB1	Byte	R/W	00h	
FF 001Ch	UMA_DB2	Byte	R/W	00h	
FF 001Dh	UMA_DB3	Byte	R/W	00h	
FF 001Eh	UMA_CTS	Byte	R/W	00h	
FF 001Fh	UMA_ECTS	Byte	R/W	03h	

A.1.3 Multi-Input Wake-Up (MIWU)

(See Section 4.2.3 on page 69.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F1C0h	WKEDG1	Byte	R/W	00h	
FF F1C2h	WKEDG2	Byte	R/W	00h	
FF F1C4h	WKEDG3	Byte	R/W	00h	
FF F1C6h	WKEDG4	Byte	R/W	00h	
FF F1C8h	WKEDG5	Byte	R/W	00h	
FF F1CAh	WKPND1	Byte	R/W1S	**XX**h	'X' = Undefined
FF F1CCh	WKPCL1	Byte	WO	–	
FF F1CEh	WKPND2	Byte	R/W1S	**XX**h	
FF F1D0h	WKPCL2	Byte	WO	–	
FF F1D2h	WKPND3	Byte	R/W1S	**XX**h	
FF F1D4h	WKPCL3	Byte	WO	–	
FF F1D6h	WKPND4	Byte	R/W1S	**XX**h	
FF F1D8h	WKPCL4	Byte	WO	–	
FF F1DAh	WKPND5	Byte	R/W1S	**XX**h	
FF F1DCh	WKPCL5	Byte	WO	–	
FF F1DEh	WKEN1	Byte	R/W	00h	
FF F1E0h	WKEN2	Byte	R/W	00h	
FF F1E2h	WKEN3	Byte	R/W	00h	
FF F1E4h	WKEN4	Byte	R/W	00h	
FF F1E6h	WKEN5	Byte	R/W	00h	

Appendix A. (Continued)

A.1.4 Interrupt Control Unit (ICU)

(See Section 4.3.4 on page 77.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF FE00h	IVCT	Byte	RO	10h	
FF FE02h	NMISTAT	Byte	ROC	00h	
FF FE04h	EXNMI	Byte	Varies per bit	**XXXX** 00**X**0b	'X' = Undefined bit
FF FE0Ah	ISTAT0	Word	RO	0000h	
FF FE0Ch	ISTAT1	Word	RO	0000h	
FF FE0Eh	IENAM0	Word	R/W	0000h	
FF FE10h	IENAM1	Word	R/W	0000h	
FF FE12h	IECLR0	Word	WO	–	
FF FE14h	IECLR1	Word	WO	–	

A.1.5 General-Purpose I/O (GPIO)

(See Section 4.4.5 on page 82.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F200h	P0DOUT	Byte	R/W	0000 0**X**0**X**b	'X' = Undefined bit
FF F201h	P0DIN	Byte	RO	**XX**h	'X' = Undefined
FF F202h	P0DIR	Byte	R/W	0000 0**X**0**X**b	
FF F203h	P0PULL	Byte	R/W	0000 0**X**0**X**b	
FF F204h	P0PUD	Byte	R/W	0000 0**X**0**X**b	
FF F205h	P0ENVDD	Byte	R/W	0000 0**X**0**X**b	
FF F210h	P1DOUT	Byte	R/W	**X**0**X**0 0000b	
FF F211h	P1DIN	Byte	RO	**XX**h	
FF F212h	P1DIR	Byte	R/W	**X**0**X**0 0000b	
FF F213h	P1PULL	Byte	R/W	**X**1**X**0 0001b	
FF F214h	P1PUD	Byte	R/W	**X**1**X**0 0000b	
FF F215h	P1ENVDD	Byte	R/W	**X**1**X**0 00**X**1b	
FF F220h	P2DOUT	Byte	R/W	0000 0**X**00b	
FF F221h	P2DIN	Byte	RO	**XX**h	
FF F222h	P2DIR	Byte	R/W	0000 0**X**00b	
FF F223h	P2PULL	Byte	R/W	0001 0**X**00b	
FF F224h	P2PUD	Byte	R/W	0000 0**X**00b	
FF F225h	P2ENVDD	Byte	R/W	000**X** 0**X**00b	
FF F230h	P3DOUT	Byte	R/W	**X**0**X**0 0000b	
FF F231h	P3DIN	Byte	RO	**XX**h	
FF F232h	P3DIR	Byte	R/W	**X**0**X**0 0000b	
FF F233h	P3PULL	Byte	R/W	**X**0**X**0 0000b	
FF F234h	P3PUD	Byte	R/W	**X**0**X**0 0000b	

Appendix A. (Continued)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F235h	P3ENVDD	Byte	R/W	**X**1**X**1 0000b	
FF F240h	P4DOUT	Byte	R/W	0000 00**X**0b	
FF F241h	P4DIN	Byte	RO	**XX**h	
FF F242h	P4DIR	Byte	R/W	0000 00**X**0b	
FF F243h	P4PULL	Byte	R/W	0000 00**X**0b	
FF F244h	P4PUD	Byte	R/W	0000 00**X**0b	
FF F245h	P4ENVDD	Byte	R/W	1111 11**X**1b	
FF F250h	P5DOUT	Byte	R/W	000**X** 0000b	
FF F251h	P5DIN	Byte	RO	**XX**h	
FF F252h	P5DIR	Byte	R/W	000**X** 0000b	
FF F253h	P5PULL	Byte	R/W	000**X** 0000b	
FF F254h	P5PUD	Byte	R/W	000**X** 0000b	
FF F255h	P5ENVDD	Byte	R/W	000**X** 1111b	
FF F260h	P6DOUT	Byte	R/W	**X**0**X**0 0000b	
FF F261h	P6DIN	Byte	RO	**XX**h	
FF F262h	P6DIR	Byte	R/W	**X**0**X**0 0000b	
FF F263h	P6PULL	Byte	R/W	**X**0**X**0 0000b	
FF F264h	P6PUD	Byte	R/W	**X**0**X**0 0000b	
FF F265h	P6ENVDD	Byte	R/W	**X**0**X**0 0000b	
FF F270h	P7DOUT	Byte	R/W	000**X X**000b	
FF F271h	P7DIN	Byte	RO	**XX**h	
FF F272h	P7DIR	Byte	R/W	000**X X**000b	
FF F273h	P7PULL	Byte	R/W	100**X X**010b	
FF F274h	P7PUD	Byte	R/W	000**X X**010b	
FF F275h	P7ENVDD	Byte	R/W	000**X X**111b	
FF F280h	P8DOUT	Byte	R/W	0**XX**0 000**X**b	
FF F281h	P8DIN	Byte	RO	**XX**h	
FF F282h	P8DIR	Byte	R/W	0**XX**0 000**X**b	
FF F283h	P8PULL	Byte	R/W	0**XX**0 000**X**b	
FF F284h	P8PUD	Byte	R/W	0**XX**0 000**X**b	
FF F285h	P8ENVDD	Byte	R/W	1**XX**1 110**X**b	
FF F290h	Reserved				
FF F291h	P9DIN	Byte	RO	**XX**h	
FF F292h-FF F294h	Reserved				
FF F295h	P9ENVDD	Byte	R/W	00h	

Appendix A. (Continued)

A.1.6 Keyboard Scan

(See Section 4.5.2 on page 86.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F6C4h	KBSIN	Byte	RO	**XX**h	'X' = Undefined
FF F6C5h	KBSINPU	Byte	R/W	00h	
FF F6C6h	KBSOUT0	Word	R/W	FFFFh	
FF F6C8h	KBSOUT1	Word	R/W	0003h	

A.1.7 System Glue

(See Section 4.6.3 on page 88.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F6C0h	IOEE	Byte	R/W	00h	
FF F6C1h	Reserved				
FF F6C2h	SMB_SBD	Byte	R/W1C	00h	
FF F6C3h	SMB_EEN	Byte	R/W	00h	

A.1.8 PS/2

(See Section 4.7.5 on page 96.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F300h	PSDAT	Byte	R/W	**XX**h	'X' = Undefined
FF F302h	PSTAT	Byte	RO	**X**000 0000b	'X' = Undefined bit
FF F304h	PSCON	Byte	R/W	00h	
FF F306h	PSOSIG	Byte	R/W	47h	
FF F308h	PSISIG	Byte	RO	**XX**h	
FF F30Ah	PSIEN	Byte	R/W	00h	

A.1.9 Multi-Function 16-Bit Timer (MFT)

(See Section 4.8.8 on page 111.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F340h	T1CNT1	Word	R/W	**XXXX**h	'X' = Undefined
FF F342h	T1CRA	Word	R/W	**XXXX**h	
FF F344h	T1CRB	Word	R/W	**XXXX**h	
FF F346h	T1CNT2	Word	R/W	**XXXX**h	
MFF F380h	T1PRSC	Byte	R/W	00h	
FF F34Ah	T1CKC	Byte	R/W	00h	
FF F34Ch	T1MCTRL	Byte	R/W	00h	
FF F34Eh	T1ICTRL	Byte	R/1WS	00h	
FF F350h	T1ICLR	Byte	WO	–	
FF F352h	T1IEN	Byte	R/W	00h	
FF F354h	T1CPA	Word	R/W	0000h	

Appendix A. (Continued)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F356h	T1CPB	Word	R/W	0000h	
FF F358h	T1CPCFG	Byte	R/W	00h	
FF F380h	T2CNT1	Word	R/W	**XXXX**h	
FF F382h	T2CRA	Word	R/W	**XXXX**h	
FF F384h	T2CRB	Word	R/W	**XXXX**h	
FF F386h	T2CNT2	Word	R/W	**XXXX**h	
FF F388h	T2PRSC	Byte	R/W	00h	
FF F38Ah	T2CKC	Byte	R/W	00h	
FF F38Ch	T2MCTRL	Byte	R/W	00h	
FF F38Eh	T2ICTRL	Byte	R/1WS	00h	
FF F390h	T2ICLR	Byte	WO	–	
FF F392h	T2IEN	Byte	R/W	00h	
FF F394h	T2CPA	Word	R/W	0000h	
FF F396h	T2CPB	Word	R/W	0000h	
FF F398h	T2CPCFG	Byte	R/W	00h	

A.1.10 Pulse Width Modulator (PWM)

(See Section 4.9.5 on page 119.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F440h	PRSCA	Word	R/W	0000h	
FF F442h	CTRA	Word	R/W	FFFFh	
FF F444h	PWMCTLA	Byte	R/W	00h	
FF F446h	DCR0A	Word	R/W	0000h	
FF F448h	DCR1A	Word	R/W	0000h	
FF F480h	PRSCB	Word	R/W	0000h	
FF F482h	CTRB	Word	R/W	FFFFh	
FF F484h	PWMCTLB	Byte	R/W	00h	
FF F486h	DCR0B	Word	R/W	0000h	

A.1.11 Timer and Watchdog (TWD)

(See Section 4.11.3 on page 135.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F4C0h	TWCFG	Byte	Varies per bit or RUP	00h	
FF F4C2h	TWCP	Byte	R/W or RUP	00h	
FF F4C4h	TWDT0	Word	R/W or RUP	FFFFh	

Appendix A. (Continued)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F4C6h	T0CSR	Byte	Varies per bit	00h	
FF F4C8h	WDCNT	Byte	WO or Touch	0Fh	
FF F4CAh	WDSDM	Byte	WO	–	

A.1.12 Core Universal Asynchronous Receiver-Transmitter (CR_UART)

(See Section 4.10.4 on page 127.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F680h	UTBUF	Byte	R/W	00h	
FF F682h	URBUF	Byte	RO	**XX**h	'X' = Undefined
FF F684h	UICTRL	Byte	Varies per bit	01h	
FF F686h	USTAT	Byte	RO	00h	
FF F688h	UFRS	Byte	R/W	00h	
FF F68Ah	UMDSL	Byte	R/W	00h	
FF F68Ch	UBAUD	Byte	R/W	00h	
FF F68Eh	UPSR	Byte	R/W	00h	

A.1.13 Analog to Digital Converter (ADC)

(See Section 4.12.5 on page 145.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F800h	ADCSTS	Word	Varies per bit	0000h	
FF F802h	ADCCNF	Word	Varies per bit	0000h	
FF F804h	ATCTL	Word	R/W	033Fh	
FF F806h	ASCADD	Word	R/W	0000h	
FF F808h	ADCCS	Word	R/W	0000h	
FF F840h	CHN0DAT	Word	RO	(0**X**)**XXX**h	'X' = Undefined; bit 15 = 0
FF F842h	CHN1DAT	Word	RO	(0**X**)**XXX**h	
FF F844h	CHN2DAT	Word	RO	(0**X**)**XXX**h	
FF F846h	CHN3DAT	Word	RO	(0**X**)**XXX**h	

A.1.14 Digital to Analog Converter (DAC)

(See Section 4.13.5 on page 153.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F880h	DACCTRL	Byte	R/W	00h	
FF F882h	DACDAT0	Byte	R/W	00h	
FF F884h	DACDAT1	Byte	R/W	00h	

Appendix A. (Continued)

A.1.15 SMBus Interface (SMB)

(See Section 4.14.8 on page 160.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F500h	SMB1SDA	Byte	R/W	**XX**h	'X' = Undefined
FF F502h	SMB1ST	Byte	Varies per bit	00h	
FF F504h	SMB1CST	Byte	Varies per bit	000**X** 0000b	'X' = Undefined bit
FF F506h	SMB1CTL1	Byte	R/W	00h	
FF F508h	SMB1ADDR1	Byte	R/W	00h	
FF F50Ah	SMB1CTL2	Byte	R/W	00h	
FF F50Ch	SMB1ADDR2	Byte	R/W	00h	
FF F50Eh	SMB1CTL3	Byte	R/W	00h	
FF F540h	SMB2SDA	Byte	R/W	**XX**h	
FF F542h	SMB2ST	Byte	Varies per bit	00h	
FF F544h	SMB2CST	Byte	Varies per bit	000**X** 0000b	
FF F546h	SMB2CTL1	Byte	R/W	00h	
FF F548h	SMB21ADDR1	Byte	R/W	00h	
FF F54Ah	SMB2CTL2	Byte	R/W	00h	
FF F54Ch	SMB2ADDR2	Byte	R/W	00h	
FF F54Eh	SMB2CTL3	Byte	R/W	00h	

A.1.16 SensorPath Bus Interface (SPB)

(See Section 4.15.3 on page 169.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F140h	SPB_CTL	Byte	Varies per bit or RUP	00h	
FF F141h	SPB_BUST	Byte	R/W1C or RUP	00h	
FF F142h	SPB_WKST	Byte	Varies per bit	00h	
FF F143h	SPB_INTA	Byte	R/W or RUP	00h	
FF F144h	SPB_DAT_0	Byte	R/W or RUP	00h	
FF F145h	SPB_DAT_1	Byte	R/W or RUP	00h	
FF F146h	SPB_DAT_2	Byte	R/W or RUP	00h	
FF F147h	SPB_DAT_3	Byte	R/W or RUP	00h	
FF F148h	SPB_CFG	Byte	R/W	00h	
FF F149h	CLK_CFG	Byte	R/W	06h	

A.1.17 Power Management Controller (PMC)

(See Section 4.18.4 on page 179.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F100h	PMCSR	Byte	Varies per bit	20h	

Appendix A. (Continued)

A.1.18 High-Frequency Clock Generator (HFCG)

(See Section 4.19.3 on page 183.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F080h	HFCGCTRL	Byte	Varies per bit	00h	
FF F082h	HFCGML	Byte	R/W or RO	89h	
FF F084h	HFCGMH	Byte	R/W or RO	09h	
FF F086h	HFCGN	Byte	R/W or RO	02h	
FF F088h	HFCGP	Byte	R/W	21h	

A.1.19 Debugger and Development Support (JTAG)

(See Section 4.20.4 on page 188.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F030h	DBG_DID	2-Word	RO	0FE2 201Fh	
FF F034h	DBGFRZEN1	Byte	R/W	FFh	

A.1.20 Core Bus Watcher

(See Section 4.21.3 on page 190.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F700h	WADR	2-Word	RO	0000 0000h	
FF F704h	WSTAT	2-Word	RO	0000 0000h	
FF F708h	WICLR	2-Word	WO	–	
FF F70Ch	WCNTRL	2-Word	Varies per bit	0000 0000h	

A.1.21 Keyboard and Mouse Controller Interface

(See Section 5.1.4 on page 197.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F780h	HICTRL	Byte	R/W	00h	
FF F782h	HIIRQC	Byte	R/W	07h	
FF F784h	HIKMST	Byte	Varies per bit	00h	
FF F786h	HIKDO	Byte	WO	–	
FF F788h	HIMDO	Byte	WO	–	
FF F78Ah	HIKMDI	Byte	RO	**XX**h	'X' = Undefined

Appendix A. (Continued)

A.1.22 Power Management (PM) Channels

(See Section 5.2.4 on page 206.

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F78Ch	HIPM1ST	Byte	Varies per bit	00h	
FF F78Eh	HIPM1DO	Byte	WO	–	
FF F790h	HIPM1DI	Byte	RO	**XX**h	'X' = Undefined
FF F792h	HIPM1DOC	Byte	WO	–	
FF F794h	HIPM1DOM	Byte	WO	–	
FF F796h	HIPM1DIC	Byte	RO	**XX**h	
FF F798h	HIPM1CTL	Byte	R/W	40h	
FF F79Ah	HIPM1IC	Byte	R/W	41h	
FF F79Ch	HIPM1IE	Byte	R/W	00h	
FF F79Eh	HIPM2ST	Byte	Varies per bit	00h	
FF F7A0h	HIPM2DO	Byte	WO	–	
FF F7A2h	HIPM2DI	Byte	RO	**XX**h	
FF F7A4h	HIPM2DOC	Byte	WO	–	
FF F7A6h	HIPM2DOM	Byte	WO	–	
FF F7A8h	HIPM2DIC	Byte	RO	**XX**h	
FF F7AAh	HIPM2CTL	Byte	R/W	C0h	
FF F7ACh	HIPM2IC	Byte	R/W	41h	
FF F7AEh	HIPM2IE	Byte	R/W	00h	

A.1.23 Shared Memory (SHM)

(See Section 5.4.10 on page 223.)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF 0400h	SMC_STS	Byte	R/W1C	00h	
FF 0401h	SMC_CTL	Byte	Varies per bit	C0h	
FF 0402h	FLASH_SIZE	Byte	R/W	21h	
FF 0403h	FWH_ID_LPC	Byte	R/W	00h	
FF 0406h	WIN_PROT	Byte	R/W	00h	
FF 0407h	WIN_SIZE	Byte	R/W	44h	
FF 0408h	SHAW1_SEM	Byte	Varies per bit	00h	Only if Shared Access Window 1 is enabled when FLASH_ACC_EN = 0.
FF 0409h	SHAW2_SEM	Byte	Varies per bit	00h	Only if Shared Access Window 2 is enabled.
FF 040Ah	WIN_BASE1	Word	R/W	FFFFh	
FF 040Ch	WIN_BASE2	Word	R/W	FFFFh	
FF 0410h	SMCORP0	Word	R/W1S	0000h	
FF 0412h	SMCORP1	Word	R/W1S	0000h	

Appendix A. (Continued)

Location	Mnemonic	Size	Type	Reset Value	Comments
FF 0414h	SMCORP2	Word	R/W1S	0000h	
FF 0416h	SMCORP3	Word	R/W1S	0000h	
FF 0418h	SMCOWP0	Word	R/W1S	0000h	
FF 041Ah	SMCOWP1	Word	R/W1S	0000h	
FF 041Ch	SMCOWP2	Word	R/W1S	0000h	
FF 041Eh	SMCOWP3	Word	R/W1S	0000h	

A.1.24 Core Access to Host

(See Section 5.3.1 on page 212

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F740h	IHIOA	Word	R/W	0000h	
FF F742h	IHD	Byte	R/W	00h	
FF F744h	LKSIOHA	Word	R/W	0000h	
FF F746h	SIOLV	Word	R/W1C	0000h	
FF F748h	CRSMAE	Word	R/W	0000h	
FF F74Ah	SIBCTRL	Byte	Varies per bit	00h	

A.1.25 Mobile System Wake-Up Control (MSWC)

(See Section 5.5.6 on page 241.

Location	Mnemonic	Size	Type	Reset Value	Comments
FF F040h	MSWCTL1	Byte	Varies per bit	1000 0**XX**0b	'X' = Undefined bit
FF F042h	MSWCTL2	Byte	Varies per bit	00h	
FF F044h	MSWCTL3	Byte	Varies per bit	09h	
FF F048h	HCBAL	Byte	R/W or RO	00h	
FF F04Ah	HCBAH	Byte	R/W or RO	00h	
FF F04Ch	MSIEN2	Byte	R/W	00h	
FF F04Eh	MSHES0	Byte	R/W1C	00h	
FF F050h	MSHEIE0	Byte	R/W	00h	

Appendix A. (Continued)

A.2 HOST DOMAIN REGISTERS

A.2.1 Keyboard and Mouse Controller Interface

(See Section 5.1.3 on page 196.)

Address	Mnemonic	Size	Type	Reset Value	Comments
In index 60h, 61h of LDN 6	DBBOUT	Byte	R	**XX**h	Address defaults to 60h; 'X' = Undefined
In index 62h, 63h of LDN 6	STATUS	Byte	R	00h	Address defaults to 64h
In index 60h, 61h of LDN 6	DBBIN	Byte	W	**XX**h	Address defaults to 60h
In index 62h, 63h of LDN 6	COMAND	Byte	W	**XX**h	Address defaults to 64h

A.2.2 Power Management Channel 1 (PM1)

(See Section 5.2.2 on page 201.)

Address	Mnemonic	Size	Type	Reset Value	Comments
In index 60h, 61h of LDN 17	DBBOUT	Byte	R	**XX**h	Address defaults to 62h; 'X' = Undefined
In index 62h, 63h of LDN 17	STATUS	Byte	R	00h	Address defaults to 66h
In index 60h, 61h of LDN 17	DBBIN	Byte	W	**XX**h	Address defaults to 62h
In index 62h, 63h of LDN 17	COMAND	Byte	W	**XX**h	Address defaults to 66h

A.2.3 Power Management Channel 2 (PM2)

(See Section 5.2.2 on page 201.)

Address	Mnemonic	Size	Type	Reset Value	Comments
In index 60h, 61h of LDN 18	DBBOUT	Byte	R	**XX**h	Address defaults to 68h; 'X' = Undefined
In index 62h, 63h of LDN 18	STATUS	Byte	R	00h	Address defaults to 6Ch
In index 60h, 61h of LDN 18	DBBIN	Byte	W	**XX**h	Address defaults to 68h
In index 62h, 63h of LDN 18	COMAND	Byte	W	**XX**h	Address defaults to 6Ch

Appendix A. (Continued)

A.2.4 Shared Memory (SHM)

(See Section 5.4.9 on page 221.)

Offset	Mnemonic	Size	Type	Reset Value	Comments
00h	SMHAP0	Byte	Varies per bit	02h	Offset from base address in index 60h, 61h of LDN 15 (I/O access)
01h	SMHAP1	Byte	Varies per bit	02h	
02h	SMHAP2	Byte	Varies per bit	02h	
03h	SMHAP3	Byte	Varies per bit	02h	
00h[1]	SHAW1_SEM	Byte	Varies per bit	00h	Offset from base address in SHAW1BA_3-0 of LDN 15 (Memory access)
00h[2]	SHAW2_SEM	Byte	Varies per bit	00h	Offset from base address in SHAW2BA_3-0 of LDN 15 (Memory access)

1. Only if Shared Access Window 1 is enabled when FLASH_ACC_EN = 0.
2. Only if Shared Access Window 2 is enabled.

A.2.5 Mobile System Wake-Up Control (MSWC)

(See Section 5.5.5 on page 236.)

Offset	Mnemonic	Size	Type	Reset Value	Comments
Banks 0, 1, 2 and 3 - Common Control and Status Registers					
00h	WK_STS0	Byte	R/W1C	00h	Offset from base address in index 60h, 61h of LDN 4
02h	WK_EN0	Byte	R/W	00h	
04h	WK_CFG	Byte	R/W	20h	
06h	WK_SIGV	Byte	RO	**XX**h	'X' = Undefined
07h	WK_STATE	Byte	WO	–	
08h-12h	Reserved				
Bank 2 - Event Routing Configuration Registers					
13h	WK_SMIEN0	Byte	R/W	00h	
14h	Reserved				
15h	WK_IRQEN0	Byte	R/W	00h	
16h-1Fh	Reserved				

A.2.6 SuperI/O Configuration

(See Section 6.1.9 on page 254.)

Index	Mnemonic	Size	Type	Reset Value	Comments
20h	SID	Byte	RO	FCh	
21h	SIOCF1	Byte	Varies per bit or RO	11h	
22h-24h	Reserved exclusively for Winbond use				
25h	SIOCF5	Byte	R/W	00h	
26h	Reserved exclusively for Winbond use				

Appendix A. (Continued)

Index	Mnemonic	Size	Type	Reset Value	Comments
27h	SRID	Byte	RO	**X XXXX**b	'X' = Undefined bit
28h-2C	Reserved exclusively for Winbond use				
2Dh	SIOCFD	Byte	Varies per bit	00h	
2Eh-2Fh	Reserved exclusively for Winbond use				

A.2.7 Host GPIO (HGPIO) Configuration

(See Section 6.1.12 on page 259.)

Index	Mnemonic	Size	Type	Reset Value	Comments
F0h	HGPSEL	Byte	R/W	00h	
F1h	HGPCFG1	Byte	Varies per bit or RO	44h	
F2h	HGPEVR	Byte	R/W or RO	00h	
F3h	HGPCFG2	Byte	R/W or RO	00h	

A.2.8 Shared Memory (SHM) Configuration

(See Section 6.1.13 on page 263.)

Index	Mnemonic	Size	Type	Reset Value	Comments
F0h	SHM_CFG	Byte	R/W	0000 **S**00**S**b	'S' = Strap dependent bit
F1h	WIN_CFG	Byte	Varies per bit	07h	
F2h-F3h	Reserved				
F4h	SHAW1BA_0	Byte	R/W	00h	
F5h	SHAW1BA_1	Byte	R/W	00h	
F6h	SHAW1BA_2	Byte	R/W	00h	
F7h	SHAW1BA_3	Byte	R/W	00h	
F8h	SHAW2BA_0	Byte	R/W	00h	
F9h	SHAW2BA_1	Byte	R/W	00h	
FAh	SHAW2BA_2	Byte	R/W	00h	
FBh	SHAW2BA_3	Byte	R/W	00h	

A.2.9 Host GPIO (HGPIO)

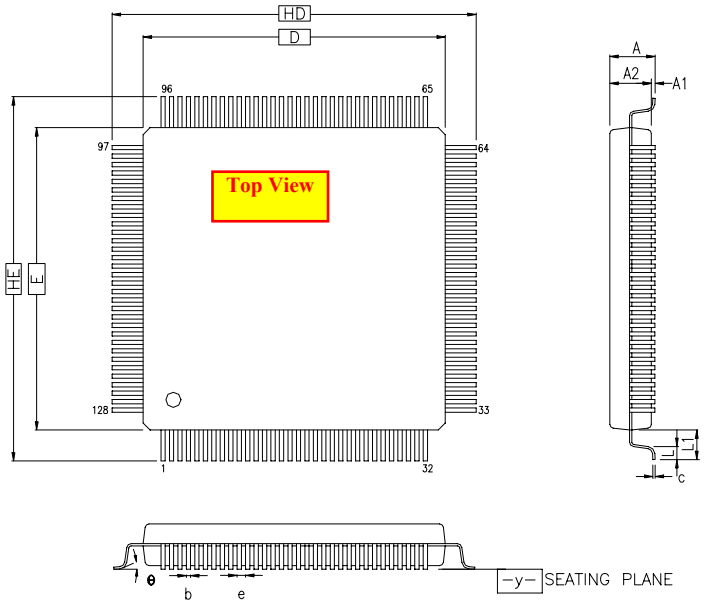
(See Section 6.2.4 on page 273.)

Offset	Mnemonic	Size	Type	Reset Value	Comments
00h	HGPDO	Byte	R/W or RO	1111 1111b	Offset from base address in index 60h, 61h of LDN 7
01h	HGPDI	Byte	RO	**XX**h	'X' = Undefined
02h	HGPEVEN	Byte	R/W	00**X**0 0000b	'X' = Undefined bit
03h	HGPEVST	Byte	R/W1C	00**X**0 0000b	

Physical Dimensions

Control dimensions are in millimeters

The symbol and geometry same as 71.08763.A0G



SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	—	—	1.60	—	—	0.063
A1	0.05	—	0.15	0.002	—	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
HD	16.00 BSC.			0.630 BSC.		
D	14.00 BSC.			0.551 BSC.		
HE	16.00 BSC.			0.630 BSC.		
E	14.00 BSC.			0.551 BSC.		
b	0.13	0.16	0.23	0.005	0.006	0.009
e	0.40 BSC.			0.016 BSC.		
θ	0°	3.5°	7°	0°	3.5°	7°
c	0.09	—	0.20	0.004	—	0.008
L	0.45	0.60	0.75	0.018	0.024	0.030
L1	1.00 REF			0.039 REF		
y	–	–	0.1	–	–	0.004

128-Pin Low-Profile Plastic Quad Flatpack (LQFP)
Order Number WPC8763LDG

Important Notice

Winbond products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Furthermore, Winbond products are not intended for applications wherein failure of Winbond products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.

Winbond customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Winbond for any damages resulting from such improper use or sales.

Headquarters
No. 4, Creation Rd. III,
Science-Based Industrial Park,
Hsinchu, Taiwan
TEL: 886-3-5770066
FAX: 886-3-5665577
http://www.winbond.com.tw/

Taipei Office
9F, No.480, Rueiguang Rd.,
Neihu District, Taipei, 114,
Taiwan, R.O.C.
TEL: 886-2-8177-7168
FAX: 886-2-8751-3579

Winbond Electronics Corporation America
2727 North First Street, San Jose,
CA 95134, U.S.A.
TEL: 1-408-9436666
FAX: 1-408-5441798

Winbond Electronics Corporation Japan
7F Daini-ueno BLDG, 3-7-18
Shinyokohama Kohoku-ku,
Yokohama, 222-0033
TEL: 81-45-4781881
FAX: 81-45-4781800

Winbond Electronics (Shanghai) Ltd.
27F, 2299 Yan An W. Rd. Shanghai,
200336 China
TEL: 86-21-62365999
FAX: 86-21-62365998

Winbond Electronics (H.K.) Ltd.
Unit 9-15, 22F, Millennium City,
No. 378 Kwun Tong Rd.,
Kowloon, Hong Kong
TEL: 852-27513100
FAX: 852-27552064

For Advanced PC Product Line information contact: APC.Support@winbond.com