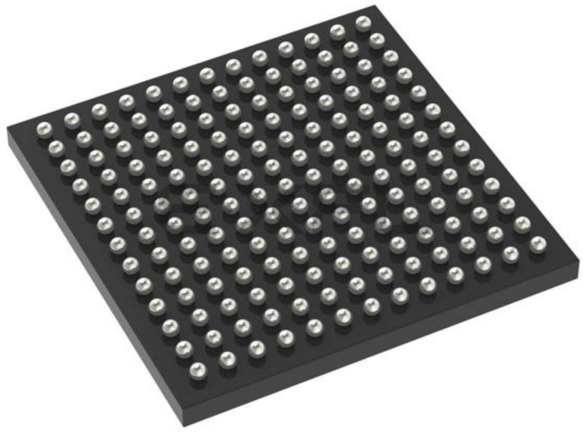
Microchip Technology - MEC1703Q-C2-XY Datasheet



Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application specific microcontrollers are engineered to

Details	
Product Status	Active
Applications	Keyboard and Embedded Controller
Core Processor	ARM® Cortex®-M4
Program Memory Type	-
Controller Series	MEC170x
RAM Size	480KB
Interface	ACPI, EBI/EMI, eSPI, I²C, LPC, PECI, PS/2, QSPI, SPI
Number of I/O	148
Voltage - Supply	1.71V ~ 3.465V
Operating Temperature	0°C ~ 70°C
Mounting Type	Surface Mount
Package / Case	169-WFBGA
Supplier Device Package	169-WFBGA (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/mec1703q-c2-xy

Table of Contents

MEC1701/MEC1703-128 WFBGA-TF	MEC1704-144 WFBGA-SZ	MEC1705-144 WFBGA-SZ	MEC1701/MEC1703-144 WFBGA-SZ	MEC1701-169 WFBGA-TN	MEC1703-169 WFBGA-TN	MEC1703-169 WFBGA-XY	Pin Name	Default (if not GPIO)	Default State (if not In)
M7	M5	M5	M5	N6	M3	R6	GPIO224/GPTP-IN4/SHD_IO1		
	A13	A13		A10	E9	C13	GPIO225/UART0_RTS#		
	F2	F2	F2	E3	C2	F1	GPIO226/LED3		
M6	L6	L6	L6	N7	M4	P7	GPIO227/SHD_IO2		
				L8	N8	R11	GPIO230		
				K7	M7	J8	GPIO231		
				G8	K8	L10	GPIO233		
				B1	A1	E4	GPIO234/VCI_IN4#		
			G13	G10	G10	H14	GPIO240		
	B8	B8	B8	D6	D9	B10	GPIO241		
	A10	A10	A10	D5	A9	A12	GPIO242		
	C8	C8	C8	D8	C9	E9	GPIO243		
	A11	A11	A11	B8	B10	A13	GPIO244		
	B11	B11	B11	C8	A10	B12	GPIO245		
	B10	B10	B10	B9	C10	D10	GPIO246		
	B9	B9	B9	A8	B9	B11	GPIO254		
F12	G11	G11	G11	F10	F12	G11	JTAG_RST#	JTAG_RST#	
A1	D3	D3	D3	G6	F4	G5	RESETI#	RESETI#	
E5	E5	E5	E5	E6	F7	D5	VBAT	VBAT	
B4	C5	C5	C5	B2	B3	B4	VCI_OUT	VCI_OUT	O2ma-High
D1	F1	F1	F1	E1	E1	D1	VFLT_PLL	VFLT_PLL	
G1	H1	H1	H1	J1	F1	G1	VR_CAP	VR_CAP	
G2	G2	G2	G2	H5	G1	J5	VREF_ADC	VREF_ADC	
E7	F6	F6	F6	E5	H8	B2	VSS1	VSS1	
F9	G3	G3	G3	J7	G7	H8	VSS2	VSS2	
E8	H5	H5	H5	K6	G6	J7	VSS3	VSS3	
H5	K1	K1	K1	L1	H6	M1	VSS_ADC	VSS_ADC	
E6	B4	B4	B4	C4	F6	A3	VSS_ANALOG	VSS_ANALOG	
E9	G5	G5	G5	G4	F8	F4	VTR1	VTR1	
J8	H8	H8	H8	L12	J8	N13	VTR2	VTR2	
J5	H6	H6	H6	J6	H7	P4	VTR3	VTR3	
F5	G6	G6	G6	H8	G8	L5	VTR_ANALOG	VTR_ANALOG	

MEC1705/MEC1704-144-SZ	MEC1701/MEC1703-144-SZ	Pin Name	Power Rail	Pad Type	Glitch Prot	Over-voltage Prot	Under-voltage Prot	Backdrive Prot
J1	J1	GPIO203/ADC03	VTR1					
		GPIO203		PIO	X		X	
		ADC03		I_AN	X		X	
J2	J2	GPIO202/ADC02	VTR1					
		GPIO202		PIO	X		X	
		ADC02		I_AN	X		X	
J3	J3	GPIO201/ADC01	VTR1					
		GPIO201		PIO	X		X	
		ADC01		I_AN	X		X	
J5	J5	GPIO064/LRESET#	VTR3	PIO	X		X	X
J6	J6	GPIO011/nSMI	VTR3	PIO	X		X	X
J7	J7	GPIO015/PWM7	VTR2	PIO	X		X	X
J8	J8	GPIO017/GPTP-IN5/KSI0	VTR2	PIO	X		X	X
J9	J9	GPIO030/TIN3/KSI5	VTR2	PIO	X		X	X
J11	J11	GPIO113/PS2_DAT1A/KSO06	VTR2	PIO	X		X	X
J12	J12	GPIO042/PECI_DAT/SB-TSI_DAT	VTR1					
		GPIO042		PIO	X		X	
		PECI_DAT/SB-TSI_DAT		PECI	X		X	
J13	J13	GPIO043/SB-TSI_CLK	VTR1					
		GPIO043		PIO	X		X	
		SB-TSI_CLK		PECI	X		X	
K1	K1	VSS_ADC		PWR				
K2	K2	GPIO207/ADC07	VTR1					
		GPIO207		PIO	X		X	
		ADC07		I_AN	X		X	
K3	K3	GPIO205/ADC05	VTR1					
		GPIO205		PIO	X		X	
		ADC05		I_AN	X		X	
K11	K11	GPIO027/TIN2/KSI4	VTR2	PIO	X		X	X
K12	K12	GPIO120/KSO07	VTR2	PIO	X		X	X
K13	K13	GPIO112/PS2_CLK1A/KSO05	VTR2	PIO	X		X	X
L1	L1	GPIO206/ADC06	VTR1					
		GPIO206		PIO	X		X	
		ADC06		I_AN	X		X	
L2	L2	GPIO067/CLKRUN#	VTR3	PIO	X		X	X

Signal	Emulated Power Rail	Gated State	Notes
GTACH0	VTR	Low	
GTACH1	VTR	Low	
I2C00_SCL	VTR	High	Note 14
I2C00_SDA	VTR	High	Note 14
I2C01_SCL	VTR	High	Note 14
I2C01_SDA	VTR	High	Note 14
I2C02_SCL	VTR	High	Note 14
I2C02_SDA	VTR	High	Note 14
I2C03_SCL	VTR	High	Note 14
I2C03_SDA	VTR	High	Note 14
I2C04_SCL	VTR	High	Note 14
I2C04_SDA	VTR	High	Note 14
I2C05_SCL	VTR	High	Note 14
I2C05_SDA	VTR	High	Note 14
I2C06_SCL	VTR	High	Note 14
I2C06_SDA	VTR	High	Note 14
I2C07_SCL	VTR	High	Note 14
I2C07_SDA	VTR	High	Note 14
I2C08_SCL	VTR	High	Note 14
I2C08_SDA	VTR	High	Note 14
I2C09_SCL	VTR	High	Note 14
I2C09_SDA	VTR	High	Note 14
I2C10_SCL	VTR	High	Note 14
I2C10_SDA	VTR	High	Note 14
ICT3	VTR	Low	
ICT4	VTR	Low	
ICT5	VTR	Low	
JTAG_CLK	VTR	Low	
JTAG_RST#	VTR	High	Note 5
JTAG_TDI	VTR	Low	
JTAG_TDO	VTR	Low	
JTAG_TMS	VTR	Low	
KBRST	VTR	Low	
KSI0	VTR	Low	
KSI1	VTR	Low	
KSI2	VTR	Low	
KSI3	VTR	Low	
KSI4	VTR	Low	
KSI5	VTR	Low	
KSI6	VTR	Low	
KSI7	VTR	Low	
KSO00	VTR	Low	
KSO01	VTR	Low	

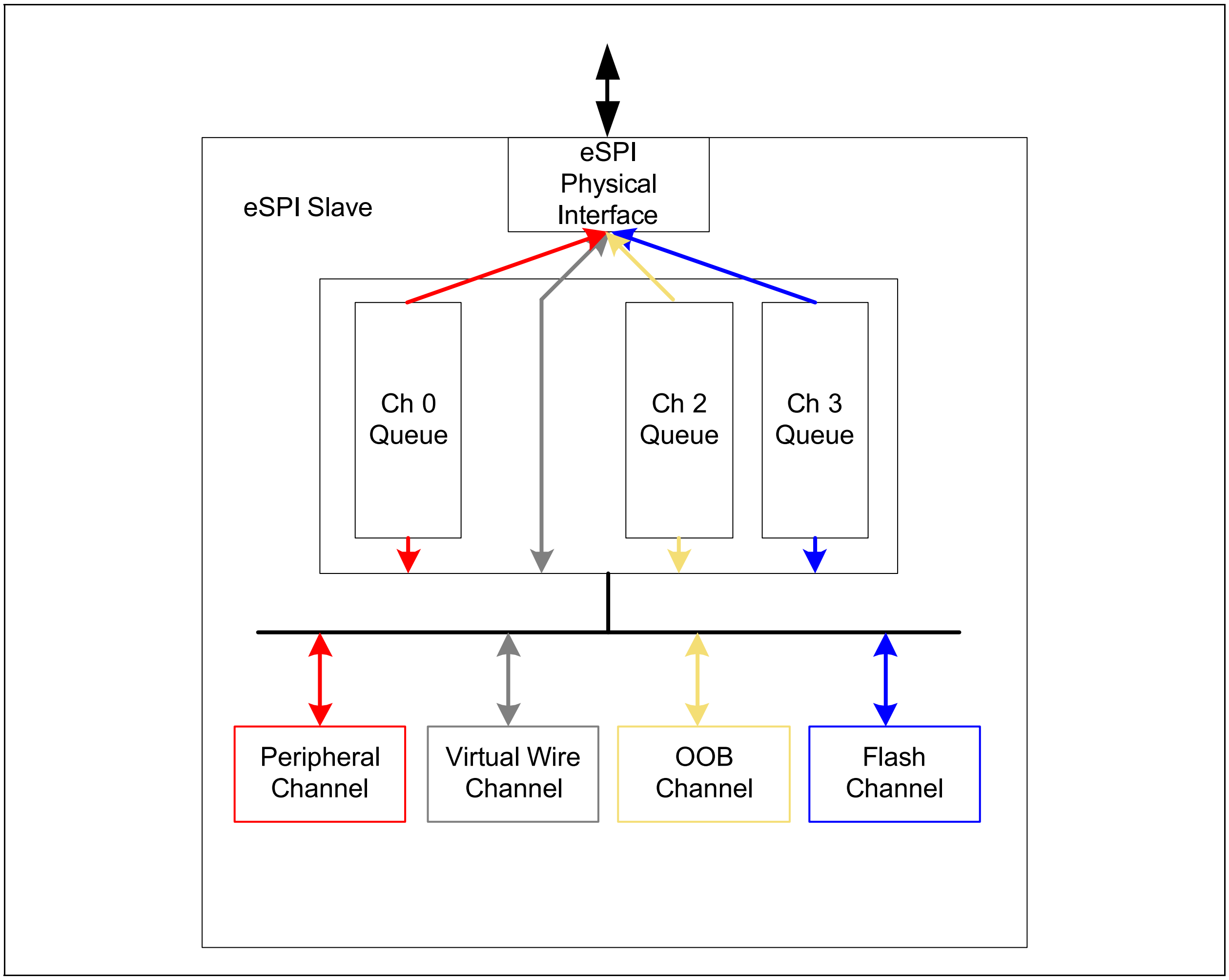
8.11.5 BLOCK ENABLE SET REGISTER

Offset	200h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:27	Reserved	R	-	-
26:8	IRQ_VECTOR_ENABLE_SET Each GIRQx register can be individually enabled to assert an interrupt event. Reads always return the current value of the enable bits for each of the GIRQs. The state of the GIRQX_ENABLE bit is determined by the corresponding GIRQX_ENABLE_SET bit and the GIRQX_ENABLE_CLEAR bit. (0=disabled, 1-enabled) 1=Interrupts in the GIRQx Source Register may be enabled 0=No effect	R/WS	0h	RESET _SYS
7:0	Reserved	R	-	-

8.11.6 BLOCK ENABLE CLEAR REGISTER

Offset	204h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:27	Reserved	R	-	-
26:8	IRQ_VECTOR_ENABLE_CLEAR Each GIRQx register can be individually disabled to inhibit interrupt events. Reads always return the current value of the internal GIRQX_ENABLE bit. The state of the GIRQX_ENABLE bit is determined by the corresponding GIRQX_ENABLE_SET bit and the GIRQX_ENABLE_CLEAR bit. (0=disabled, 1-enabled) 1=All interrupts in the GIRQx Source Register are disabled 0=No effect	R/WC	0h	RESET _SYS
7:0	Reserved	R	-	-

FIGURE 10-2: ESPI BLOCK DIAGRAM



The Flash Channel permits the EC to access the System SPI Flash through the eSPI interface.

The Out of Band (OOB) Channel enables messaging between the Out-Of-Band Processor in the system chipset and the EC. This messaging is implemented by tunneling SMBus packets over the eSPI port.

The Peripheral Channel (PC) enables the system Host to read and write locations inside the EC. The PC encapsulates legacy I/O operation as well as generic memory read and write operations. Like the Flash and the OOB Channels, all PC accesses are multiplexed over the eSPI port.

The Virtual Wire Channel provides in-band emulation of sideband pin signals between the system Core Logic and the EC, including the legacy SERIRQ interrupt signal to the system Host.

10.7 eSPI Register Summary

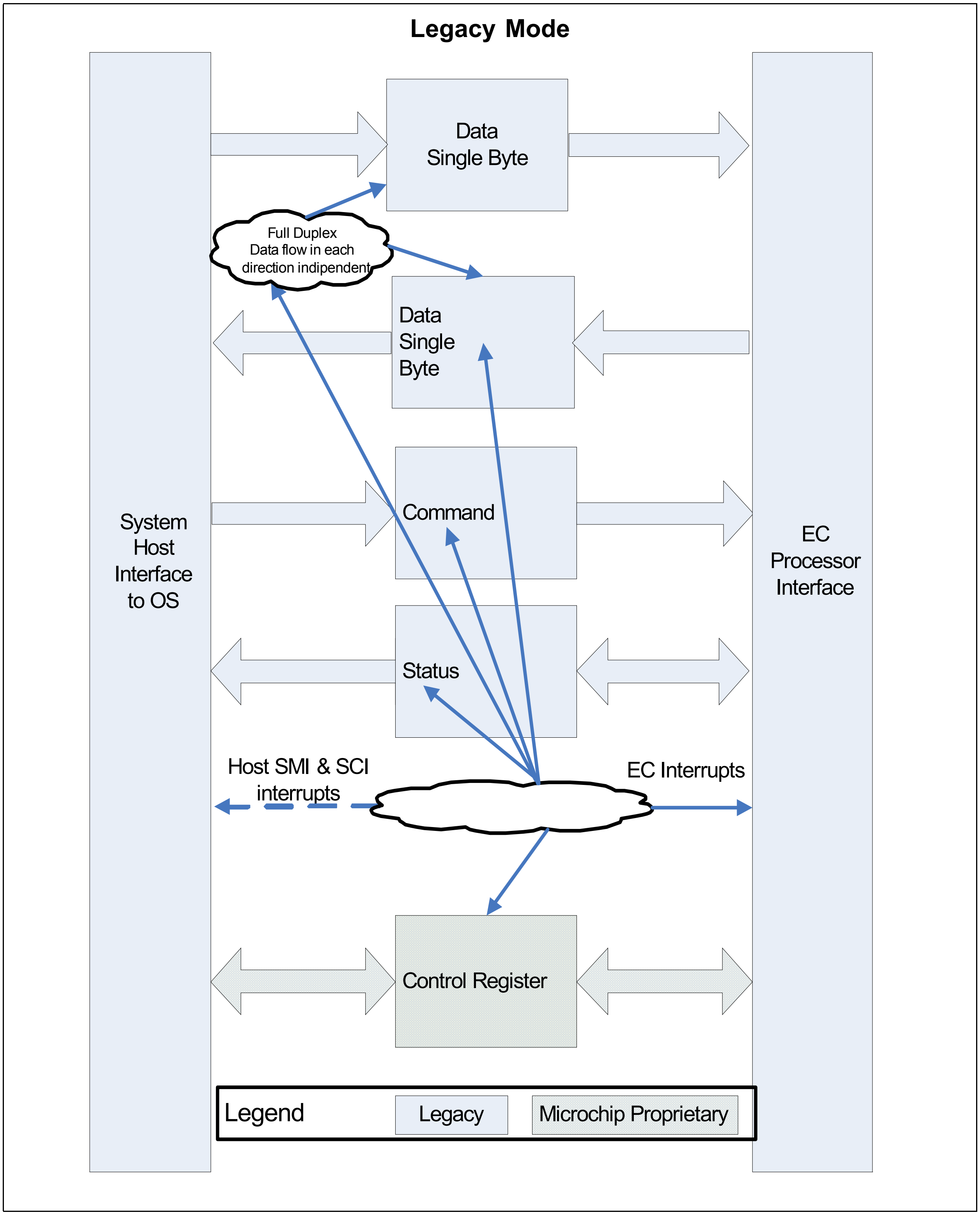
The following sections list the registers associated with the eSPI logic. The eSPI logic required three Logical Devices in order to provide access to all the required registers. These Logical Devices are called the I/O Component, the Memory Component and the Virtual Wire Component. The Base Addresses for these three blocks are shown in the Block Overview and Base Address Table in Section 3.0, "Device Inventory".

The EC may access all registers in all three Logical Devices. Host access is restricted to three ranges: Runtime Registers, located at offsets 00h through FFh from the Logical Device Base Address of the I/O Component, and Configuration Registers, located at offsets 330h through 3FFh in both the I/O Component and the Memory Component. The Runtime Registers may be mapped into the Hosts address space, either I/O or Memory, by setting the associated BAR for the

12.15.4 KEYBOARD CONTROL REGISTER

Offset	108h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7	AUXH AUX in Hardware. 1=AUXOBF of the Keyboard Status Read Register is set in hardware by a write to the EC AUX Data Register 0=AUXOBF is not modified in hardware, but can be read and written by the EC using the EC-Only alias of the EC Keyboard Status Register	R/W	0h	RESET _SYS
6	UD5 User-defined data. Readable and writable by the EC when written by the EC at its EC-only alias.	R/W	0h	RESET _SYS
5	OBFEN When this bit is ‘1’, the system interrupt signal KIRQ is driven by the bit PCOBF and MIRQ is driven by AUXOBF. When this bit is ‘0’, KIRQ and MIRQ are driven low. This bit must not be changed when OBF of the status register is equal to ‘1’.	R/W	0h	RESET _SYS
4:3	UD4 User-defined data. Readable and writable by the EC when written by the EC at its EC-only alias.	R/W	0h	RESET _SYS
2	PCOBFEN 1=reflects the value written to the PCOBF Register 0=PCOBF reflects the status of writes to the EC Data Register	R/W	0h	RESET _SYS
1	SAEN Software-assist enable. 1=This bit allows control of the GATEA20 signal via firmware 0=GATEA20 corresponds to either the last Host-initiated control of GATEA20 or the firmware write to the Keyboard Control Register or the EC AUX Data Register.	R/W	0h	RESET _SYS
0	UD3 User-defined data. Readable and writable by the EC when written by the EC at its EC-only alias.	R/W	0h	RESET _SYS

FIGURE 13-2: BLOCK DIAGRAM CORRESPONDING TO THE ACPI SPECIFICATION



14.0 ACPI PM1 BLOCK

14.1 Introduction

The MEC170x supports ACPI as described in this section. These features comply with the ACPI Specification through a combination of hardware and EC software.

14.2 References

ACPI Specification, Revision 1.0

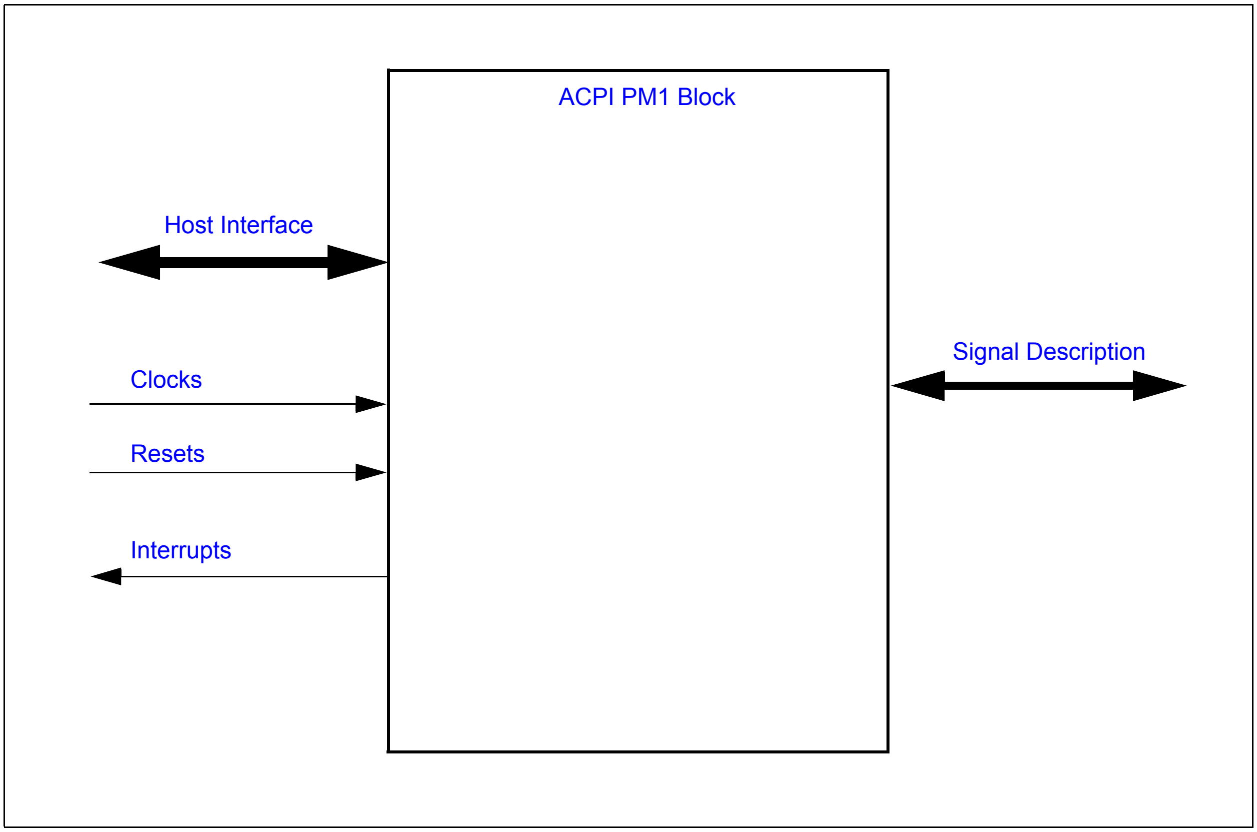
14.3 Terminology

None

14.4 Interface

This block is an IP block designed to be incorporated into a chip. It is designed to be accessed externally via the pin interface and internally via a registered host interface. The following diagram illustrates the various interfaces to the block.

FIGURE 14-1: I/O DIAGRAM OF BLOCK



15.10.5 MEMORY WRITE LIMIT 0 REGISTER

Offset	10Ah			
Bits	**Description**	**Type**	**Default**	**Reset Event**
15	Reserved	R	-	-
14:2	MEMORY_WRITE_LIMIT_0 Whenever a write of any byte in EC DATA Register is attempted and bit 15 of EC_Address is 0, the field EC_ADDRESS_MSB in the EC_Address Register is compared to this field. As long as EC_Address[14:2] is less than Memory_Write_Limit_0[14:2] the addressed bytes in the EC DATA Register will be written into the internal 32-bit address space. If EC_Address[14:2] is greater than or equal to the Memory_Write_Limit_0[14:2] no writes will take place.	R/W	0h	RESET_SYS
1:0	Reserved	R	-	-

15.10.6 MEMORY BASE ADDRESS 1 REGISTER

Offset	10Ch			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:2	MEMORY_BASE_ADDRESS_1 This memory base address defines the beginning of region 1 in the Embedded Controller's 32-bit internal address space. Memory allocated to region 1 is intended to be shared between the Host and the EC. The region defined by this base register is used when bit 15 of the EC Address Register is 1. The access will be to a memory location at an offset defined by the EC_Address relative to the beginning of the region defined by this register. Therefore, a read or write to the memory that is triggered by the EC Data Register will occur at Memory_Base_Address_1 + EC_Address.	R/W	0h	RESET_SYS
1:0	Reserved	R	-	-

15.10.7 MEMORY READ LIMIT 1 REGISTER

Offset	110h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
15	Reserved	R	-	-
14:2	MEMORY_READ_LIMIT_1 Whenever a read of any byte in the EC Data Register is attempted, and bit 15 of EC_ADDRESS is 1, the field EC_ADDRESS in the EC_Address_Register is compared to this field. As long as EC_ADDRESS is less than this value, the EC_Data_Register will be loaded from the 32-bit internal address space.	R/W	0h	RESET_SYS
1:0	Reserved	R	-	-

16.0 MAILBOX INTERFACE

16.1 Overview

The Mailbox provides a standard run-time mechanism for the host to communicate with the Embedded Controller (EC)

16.2 References

No references have been cited for this feature.

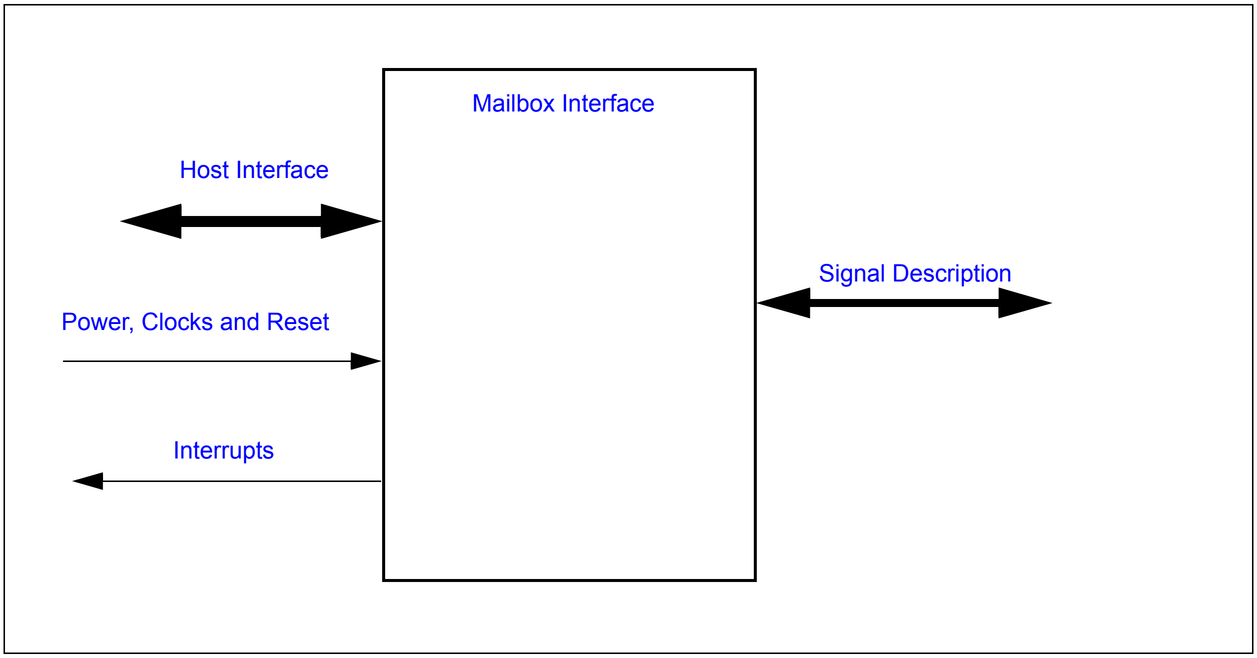
16.3 Terminology

There is no terminology defined for this section.

16.4 Interface

This block is designed to be accessed externally via the pin interface and internally via a registered host interface.

FIGURE 16-1: I/O DIAGRAM OF BLOCK



16.5 Signal Description

TABLE 16-1: SIGNAL DESCRIPTION TABLE

Name	Direction	Description
nSMI	OUTPUT	SMI alert signal to the Host.

16.6 Host Interface

The Mailbox interface is accessed by host software via a registered interface, as defined in Section 16.11, "Runtime Registers" and Section 16.12, "EC Registers".

17.0 UART

17.1 Introduction

The 16550 UART (Universal Asynchronous Receiver/Transmitter) is a full-function Serial Port that supports the standard RS-232 Interface.

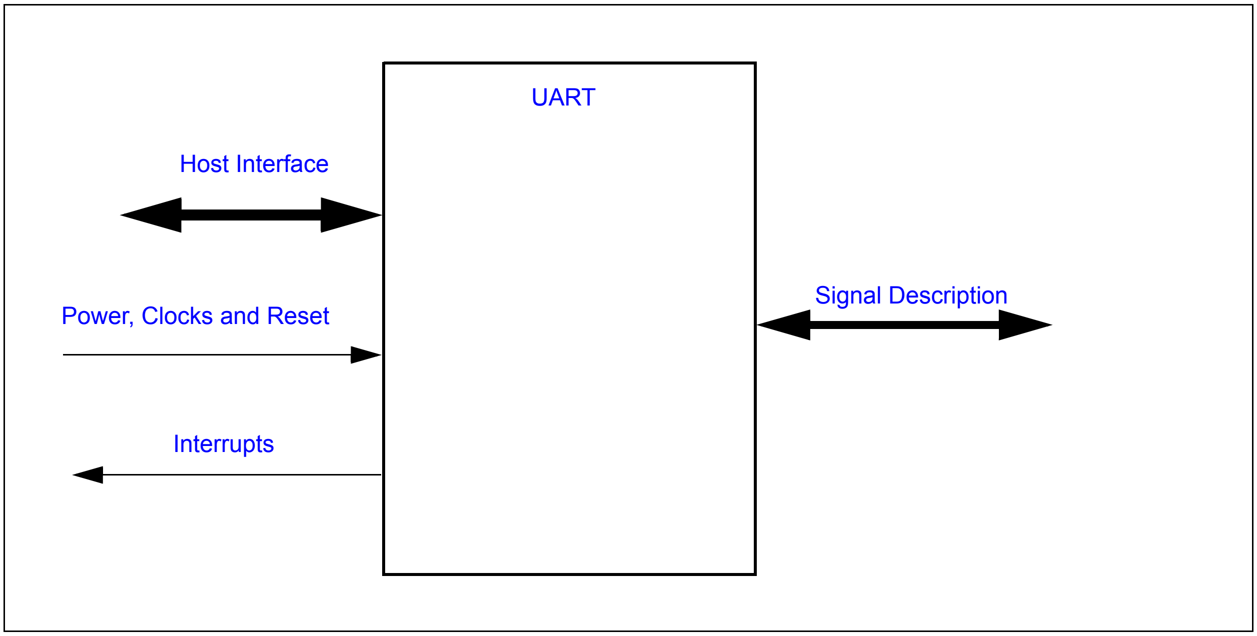
17.2 References

- EIA Standard RS-232-C specification

17.3 Interface

This block is designed to be accessed externally via the pin interface and internally via a registered host interface.

FIGURE 17-1: I/O DIAGRAM OF BLOCK



17.4 Signal Description

TABLE 17-1: SIGNAL DESCRIPTION TABLE

Name	Direction	Description
DTR#	Output	Active low Data Terminal ready output for the Serial Port. Handshake output signal notifies modem that the UART is ready to transmit data. This signal can be programmed by writing to bit 1 of the Modem Control Register (MCR). **Note:** Defaults to tri-state on V3_DUAL power on.
DCD#	Output	Active low Data Carrier Detect input for the serial port. Handshake signal which notifies the UART that carrier signal is detected by the modem. The CPU can monitor the status of DCD# signal by reading bit 7 of Modem Status Register (MSR). A DCD# signal state change from low to high after the last MSR read will set MSR bit 3 to a 1. If bit 3 of Interrupt Enable Register is set, the interrupt is generated when DCD # changes state.

TABLE 21-3: TIMER CLOCK FREQUENCIES (CONTINUED)

Timer Clock Select	Frequency Divide Select	Frequency Selected
0011b	Divide by 8	6MHz
0100b	Divide by 16	3MHz
0101b	Divide by 32	1.5MHz
0110b	Divide by 64	750KHz
0111b	Divide by 128	375KHz
1xxxb	Reserved	Reserved

For the Timer Clock, the **Timer Clock Select** value is defined by the TCLK field in the Timer x Clock and Event Control Register

21.10.2 FILTER CLOCK AND NOISE FILTER

The noise filter uses the Filter Clock (FCLK) to filter the signal on the TINx pins. for Event Mode and One-Shot Mode.

In Event Mode, the Event input is synchronized to FCLK and (if enabled) filtered by a three stage filter. The resulting recreated clock is used to clock the timer in Event mode. In Bypass Mode, configured by the FILTER_BYPASS bit in the Timer x Control Register, the pulse width of the external signal must be at least 2x the pulse width of the FCLK source. In Filter Mode, the pulse width of the external signal must be at least 4x the pulse width of the sync and filter clock

In One-Shot mode, the TIN duration could be smaller than a TCLK period. The filtered signal is latched until the signal is seen in the TCLK domain. This also applies in the filter bypass mode

Frequencies for the Filter Clock are the as those available for the Timer Clock, and are listed in Table 21-3. For the Filter Clock, the **Timer Clock Select** value is defined by the FCLK field in the Timer x Clock and Event Control Register. The choice of frequency is independent of the value chosen for the Timer Clock.

21.10.3 TIMER CONNECTIONS

For external inputs/outputs (TINx/TOUTx) to/from timers, please see Pin Configuration chapter for a description of the 16-bit Counter/Timer Interface.

TABLE 21-4: TIMER CASCADING DESCRIPTION

Timer Name	Timer Type	Over-Flow/ Under-flow Input’s Connection
Timer 0	General Purpose	from Timer 3
Timer 1	General Purpose	from Timer 0
Timer 2	General Purpose	from Timer 1
Timer 3	General Purpose	from Timer 2

Note: The cascading connections are independent of the TINx/TOUTx connections.

21.10.4 STARTING AND STOPPING

The 16-bit timers can be started and stopped by setting and clearing the ENABLE bit in the Timer x Control Register in all modes, except one-shot.

21.10.5 TIMER MODE

Timer mode is used to generate periodic interrupts to the EC. When operating in this mode the timer always counts down based on one of the internally generated clock sources. The Timer mode is selected by setting the Timer Mode Select bits in the Timer Control Register. See Section 21.11.1, "Timer x Control Register".

The period between timer interrupts and the width of the output pulse is determined by the speed of the clock source, the clock divide ratio and the value programmed into the Timer Reload Register. The timer clock source and clock rate are selected using the Clock Source Select bits (TCLK) in the Timer x Clock and Event Control Register. See Section 21.11.2, "Timer x Clock and Event Control Register".

24.0 RTOS TIMER

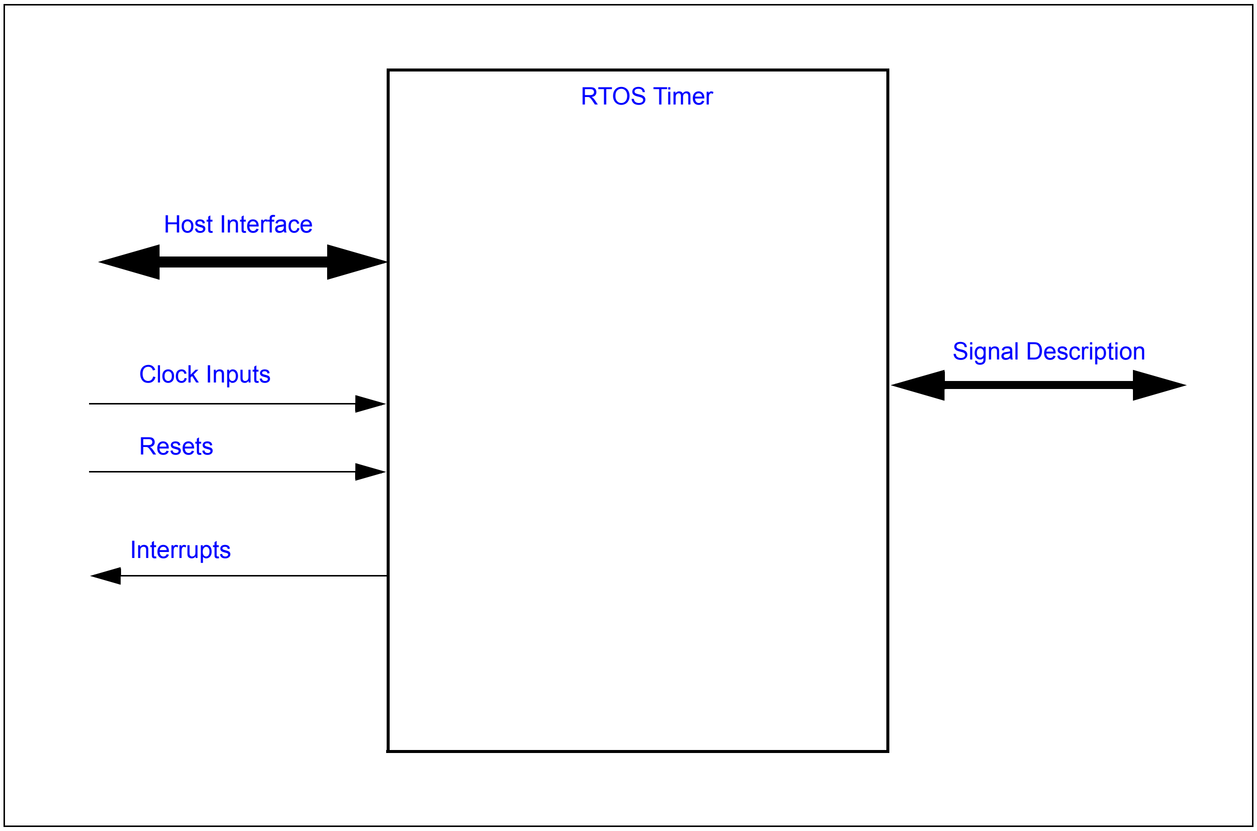
24.1 Introduction

The RTOS Timer is a low-power, 32-bit timer designed to operate on the 32kHz oscillator which is available during all chip sleep states. This allows firmware the option to sleep the processor and wake after a programmed amount of time. The timer may be used as a one-shot timer or a continuous timer. When the timer transitions to 0 it is capable of generating a wake-capable interrupt to the embedded controller. This timer may be halted during debug by hardware or via a software control bit.

24.2 Interface

This block is designed to be accessed externally via the pin interface and internally via a registered host interface.

FIGURE 24-1: I/O DIAGRAM OF BLOCK



24.3 Signal Description

There are no external signals for this block.

Name	Description
HALT	RTOS Timer Halt signal. This signal is connected to the same signal that halts the embedded controller during debug (e.g., JTAG Debugger is active, break points, etc.).

24.4 Host Interface

The embedded controller may access this block via the registers defined in Section 24.9, "EC Registers".

  - Once the fourth SPI cycle is completed, the SPI master takes the pending data in the TX_DATA register (if any) and loads it into the TX shift register. This process will be repeated until all the desired data is received.
- The host software will read and store the EEPROM data value in SPIRD - SPI RX_Data Register.
- If no more data needs to be received by the master, CS# is released and the SPI is idle. Otherwise, master continues reading the data by writing a dummy value to the TX_DATA register after every 8 SPI_CLK cycles.

Write

- The SPI block is activated by setting the enable bit in SPIAR - SPI Enable Register
- The SPIMODE bit is de-asserted '0' to enable the SPI interface in Full Duplex mode.
- The CLKPOL, TCLKPH and RCLKPH bits are de-asserted '0' to match the clocking requirements of the slave device.
- The LSBF bit is de-asserted '0' to indicate that the slave expects data in MSB-first order.
- Assert WR# high using a GPIO pin.
- Assert CS# low using a GPIO pin.
- Write a valid command word (as specified by the slave device) to the SPITD - SPI TX_Data Register with TXFE asserted '1'. The SPI master automatically clears the TXFE bit indicating the byte has been put in the TX buffer. If the shift register is empty the TX_DATA byte is loaded into the shift register and the SPI master reasserts the TXFE bit. Once the data is in the shift register the SPI master begins shifting the data value onto the SPDOUT pin and drives the SPI_CLK pin. Data on the SPDIN pin is also sampled on each clock.
- Once the TXFE bit is asserted the SPI Master is ready to receive its next byte. Before writing the next TX_DATA value, software must clear the RXBF status bit by reading the SPIRD - SPI RX_Data Register.
- Next, EEPROM address A15-A8 is written to the TX_DATA register. The SPI master automatically clears the TXFE bit, but does not begin shifting the dummy data value onto the SPDOUT pin. This byte will remain in the TX_DATA register until the TX shift register is empty.
- After 8 SPI_CLK pulses from the first transmit byte (Command Byte transmitted):
  - The first SPI cycle is complete, RXBF bit is asserted '1', and the SPINT interrupt is asserted, if enabled. The data now contained in SPIRD - SPI RX_Data Register is invalid since the last cycle was initiated solely to transmit command data to the slave. This particular slave device tri-states the SPDIN pin to the master while it is accepting command data. This SPIRD data is ignored.

USER'S NOTE: External pull-up or pull-down is required on the SPDIN pin if it is tri-stated by the slave device.

  - Once the first SPI cycle is completed, the SPI master takes the pending data in the TX_DATA register (EEPROM address A15-A8) and loads it into the TX shift register. Loading the shift register automatically asserts the TXFE bit, begins shifting the dummy data value onto the SPDOUT pin, and drives the SPI_CLK pin. Data on the SPDIN pin is also sampled on each clock. Note: The particular slave device ignores address A15-A13.
- Once the TXFE bit is asserted the SPI Master is ready to receive its next byte. Before writing the next TX_DATA value, software must clear the RXBF status bit by reading the SPIRD - SPI RX_Data Register.
- Next, EEPROM address A7-A0 is written to the TX_DATA register. The SPI master automatically clears the TXFE bit, but does not begin shifting this data value onto the SPDOUT pin. This byte will remain in the TX_DATA register until the TX shift register is empty.
- After 8 SPI_CLK pulses from the second transmit byte (Address Byte (MSB) transmitted):
  - EEPROM address A15-A8 has been transmitted to the slave completing the second SPI cycle. Once again, the RXBF bit is asserted '1' and the SPINT interrupt is asserted, if enabled. The data now contained in SPIRD - SPI RX_Data Register is invalid since the last cycle was initiated solely to transmit address data to the slave.
  - Once the second SPI cycle is completed, the SPI master takes the pending data in the TX_DATA register (EEPROM address A7-A0) and loads it into the TX shift register. Loading the shift register automatically asserts the TXFE bit, begins shifting the dummy data value onto the SPDOUT pin, and drives the SPCLK pin. Data on the SPDIN pin is also sampled on each clock.
- Once the TXFE bit is asserted the SPI Master is ready to receive its next byte. Before writing the next TX_DATA value, software must clear the RXBF status bit by reading the SPIRD - SPI RX_Data Register.
- Next, a data byte (D7:D0) is written to the TX_DATA register. The SPI master automatically clears the TXFE bit, but does not begin shifting this data value onto the SPDOUT pin. This byte will remain in the TX_DATA register until the TX shift register is empty.

37.12.6 SPI CLOCK CONTROL REGISTER

This register should not be changed during an active SPI transaction.

Offset	14h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:5	Reserved	R	-	-
4	CLKSRC Clock Source for the SPI Clock Generator. This bit should not be changed during a SPI transaction. When the field PRELOAD in the SPI Clock Generator Register is 0, this bit is ignored and the Clock Source is always the main system clock (the equivalent of setting this bit to ‘0’). 1=2MHz 0=48MHz	R/W	0h	RESET_SYS
3	Reserved	R	-	-
2	CLKPOL SPI Clock Polarity. 1=The SPI_CLK signal is high when the interface is idle and the first clock edge is a falling edge 0=The SPI_CLK is low when the interface is idle and the first clock edge is a rising edge	R/W	0h	RESET_SYS
1	RCLKPH Receive Clock Phase, the SPI_CLK edge on which the master will sample data. The receive clock phase is not affected by the SPI Clock Polarity. 1=Valid data on SPDIN signal is expected after the first SPI_CLK edge. This data is sampled on the second and following even SPI_CLK edges (i.e., sample data on falling edge) 0=Valid data is expected on the SPDIN signal on the first SPI_CLK edge. This data is sampled on the first and following odd SPI_CLK edges (i.e., sample data on rising edge)	R/W	1h	RESET_SYS
0	TCLKPH Transmit Clock Phase, the SPCLK edge on which the master will clock data out. The transmit clock phase is not affected by the SPI Clock Polarity. 1=Valid data is clocked out on the first SPI_CLK edge on SPDOUT signal. The slave device should sample this data on the second and following even SPI_CLK edges (i.e., sample data on falling edge) 0=Valid data is clocked out on the SPDOUT signal prior to the first SPI_CLK edge. The slave device should sample this data on the first and following odd SPI_CLK edges (i.e., sample data on rising edge)	R/W	0h	RESET_SYS

40.0 BC-LINK MASTER

40.1 Overview

This block provides BC-Link^{TM} connectivity to a slave device. The BC-Link^{TM} protocol includes a start bit to signal the beginning of a message and a turnaround (TAR) period for bus transfer between the Master and Companion devices.

40.2 References

No references have been cited for this feature.

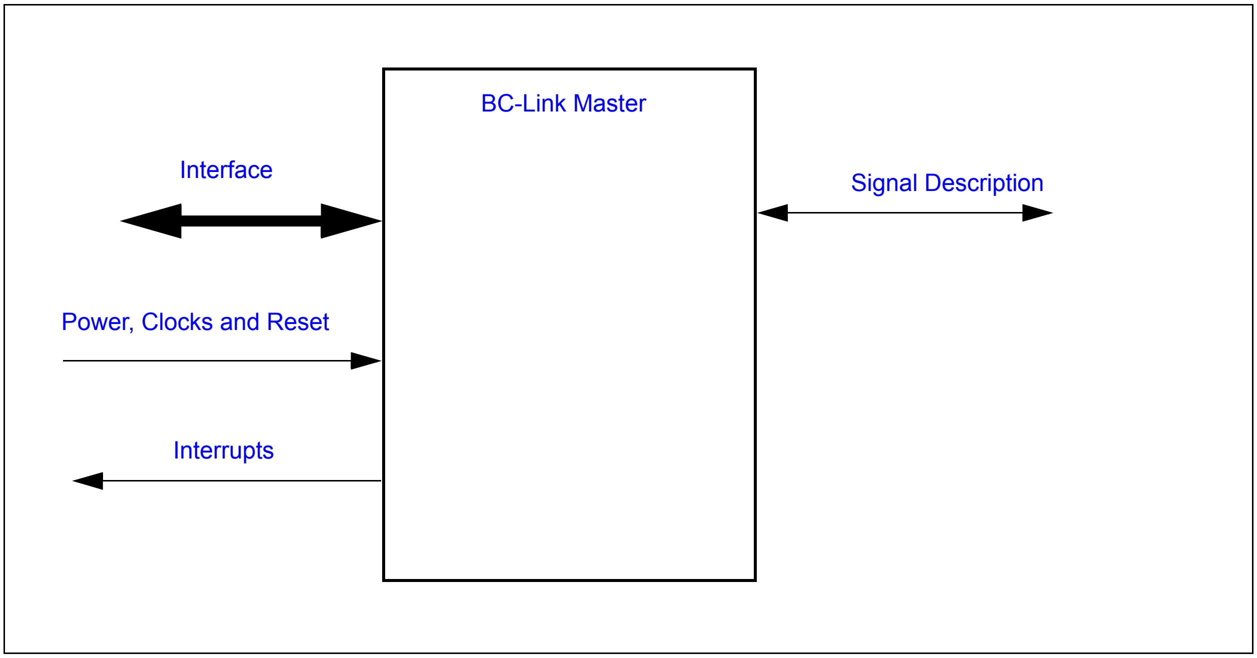
40.3 Terminology

There is no terminology defined for this section.

40.4 Interface

This block is designed to be accessed externally via the pin interface and internally via a registered host interface.

FIGURE 40-1: I/O DIAGRAM OF BLOCK



40.5 Signal Description

TABLE 40-1: SIGNAL DESCRIPTION

Name	Direction	Description
BCM_CLK	Output	BC-Link output clock
BCM_DAT	Input/Output	Bidirectional data line

Note: A weak pull-up resistor is recommended on the data line (100KW).

The maximum speed at which the BC-Link Master Interface can operate reliably depends on the drive strength of the BC-Link BCM_CLK and BCM_DAT pins, as well as the nature of the connection to the Companion device (over ribbon cable or on a PC board). The following table shows the recommended maximum speeds over a PC board as well as a 12 inch ribbon cable for selected drive strengths. The frequency is set with the BC-Link Clock Select Register.

50.2.4 DC ELECTRICAL CHARACTERISTICS FOR I/O BUFFERS

TABLE 50-3: DC ELECTRICAL CHARACTERISTICS

Parameter	Symbol	MIN	TYP	MAX	Units	Comments
PIO Type Buffer						
All PIO Buffers						Internal PU/PD selected via the GPIO Pin Control Register.
Pull-up current	I_{PU}	39	84	162	KΩ	
Pull-down current	I_{PD}	39	65	105	KΩ	
PIO						The drive strength is determined by programming bits[5:4] of the Pin Control 2 Register
DRIVE_STRENGTH = 00b	–	–	–	–	–	Same characteristics as an IO-2 mA.
DRIVE_STRENGTH = 01b	–	–	–	–	–	Same characteristics as an IO-4 mA.
DRIVE_STRENGTH = 10b	–	–	–	–	–	Same characteristics as an IO-8 mA.
DRIVE_STRENGTH = 11b	–	–	–	–	–	Same characteristics as an IO-12 mA.
I Type Input Buffer						TTL Compatible Schmitt Trigger Input
Low Input Level	V_{ILI}			0.3x VTR	V	
High Input Level	V_{IHI}	0.7x VTR			V	
Schmitt Trigger Hysteresis	V_{HYS}		400		mV	
O-2 mA Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 2 mA (min)
High Output Level	V_{OH}	VTR-0.4			V	I_{OH} = -2 mA (min)
IO-2 mA Type Buffer	–	–	–	–	–	Same characteristics as an I and an O-2mA.
OD-2 mA Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 2 mA (min)
IOD-2 mA Type Buffer	–	–	–	–	–	Same characteristics as an I and an OD-2mA.

51.13 PECI Interface

Name	Description	MIN	MAX	Units	Notes
t_{BIT}	Bit time (overall time evident on PECI pin) Bit time driven by an originator	0.495 0.495	500 250	µsec µsec	Note 1
t_{H1}	High level time for logic 1	0.6	0.8	t_{BIT}	Note 2
t_{H0}	High level time for logic 0	0.2	0.4	t_{BIT}	
t_{PECIR}	Rise time (measured from V_{OL} to $V_{IH,min}$, $Vtt_{(nom)}$–5%)	-	30 + (5 x #nodes)	ns	Note 3
t_{PECIF}	Fall time (measured from V_{OH} to $V_{IL,max}$, $Vtt_{(nom)}$+5%)	-	(30 x #nodes)	ns	Note 3

Note 1: The originator must drive a more restrictive time to allow for quantized sampling errors by a client yet still attain the minimum time less than 500 µsec. t_{BIT} limits apply equally to t_{BIT-A} and t_{BIT-M}. The MEC170x is designed to support 2 MHz, or a 500ns bit time. See the PECI 3.1 specification from Intel Corp. for further details.

2: The minimum and maximum bit times are relative to t_{BIT} defined in the Timing Negotiation pulse. See the PECI 3.1 specification from Intel Corp. for further details.

3: "#nodes" is the number of nodes on the PECI bus; host and client nodes are counted as one each. Extended trace lengths may appear as extra nodes. Refer also to Table 29-2, "PECI Routing Guidelines". See the PECI 3.1 specification from Intel Corp. for further details.

FIGURE 51-28: BC-LINK WRITE TIMING

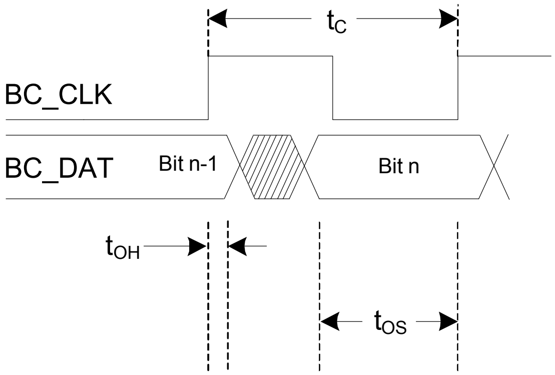


TABLE 51-25: BC-LINK MASTER TIMING DIAGRAM PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t_C	BC Clock Frequency			24	MHz
t_f	BC Clock Period	42			ns
t_{OS}	BC-Link Master DATA output setup time before rising edge of CLK.			t_c-$t_{OH\text{-}MAX}$	nsec
t_{OH}	BC-Link Master Data hold time after falling edge of CLK			10	nsec
t_{IS}	BC-Link Master DATA input setup time before rising edge of CLK.	15			nsec
t_{IH}	BC-Link Master DATA input hold time after rising edge of CLK.	0			nsec

Note 1: The (t_{IH} in Table 51-25) BC-Link Master DATA input must be stable before next rising edge of CLK.

2: The BC-Link Clock frequency is limited by the application usage model (see BC-Link Master Section 40.5, "Signal Description"). The timing budget equation is as follows for data from BC-Link slave to master:

Tc > TOD(master-clk) + Tprop(clk) +TOD(slave) + Tprop(slave data) + TIS(master).