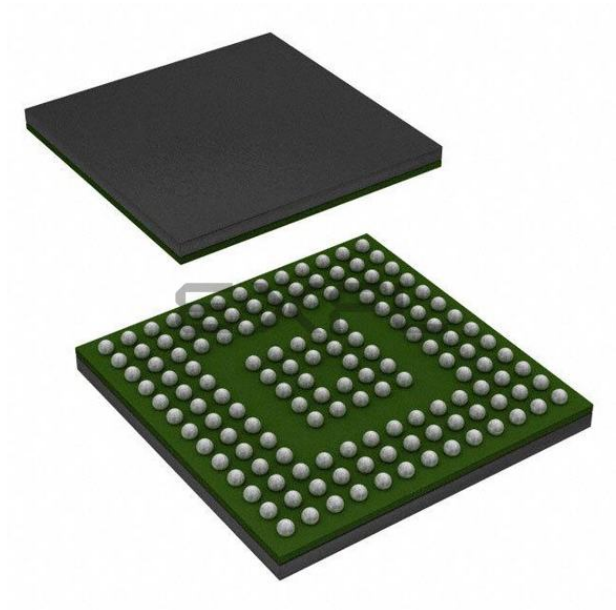


Welcome to E-XFL.COM

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	123
Voltage - Supply	1.71V ~ 3.465V
Operating Temperature	-40°C ~ 85°C
Mounting Type	Surface Mount
Package / Case	144-WFBGA
Supplier Device Package	144-WFBGA (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/mec1705q-c2-i-sz

- ADC Interface
  - 10-bit Conversion in 1μs
  - 16 Channels
  - Integral Non-Linearity of ±1.5 LSB; Differential Non-Linearity of ±1.0 LSB
- Two Standard 16C550 UARTs
  - Accessible from Host and EC
  - One UART with full 8-pin Modem Control
  - One UART with 4-pin Interface
  - Programmable Input/output Pin Polarity Inversion
  - Programmable Main Power or Standby Power Functionality
- Two Port 80h Debug Ports for BIOS Debug
  - Ports, Assignable to Any LPC IO Address
  - 24-bit Timestamp with Adjustable Timebase
  - 16-Entry FIFO
- Trace FIFO Debug Port (TFDP)
- Integrated Standby Power Reset Generator
  - Reset Input Pin
  - Reset Output Pin
- Clock Generator
  - 32.768KHz Clock Source
    - Low power 32KHz crystal oscillator
    - Optional use of a crystal-free silicon oscillator with ±2% Accuracy
    - Optional use of 32.768 KHz input Clock
    - Operational on Suspend Power
  - Programmable Clock Power Management Control and Distribution
  - 48 MHz PLL
- Multi-purpose AES Cryptographic Engine
  - Hardware support for ECB, CTR, CBC and OFB AES modes
  - Support for 128-bit, 192-bit and 256-bit key length
  - DMA interface to SRAM, shared with Hash engine
- Cryptographic Hash Engine
  - Support for SHA-1, SHA-256, SHA-384, SHA-512
  - DMA interface to SRAM, shared with AES engine
- Public Key Cryptographic Engine
  - Hardware support for RSA and Elliptic Curve public key algorithms
  - RSA keys length of 1024 or 2048 bis
  - ECC Prime Field and Binary Field keys up to 640 bits
  - Microcoded support for standard public key algorithms
- Cryptographic Features
  - True Random Number Generator
    - 1K bit FIFO
  - Monotonic Counter
- Packages
  - 144 Pin WFBGA RoHS Compliant package
  - 169 Pin WFBGA RoHS Compliant package

MEC1705/MEC1704-144-SZ	MEC1701/MEC1703-144-SZ	Pin Name	Power Rail	Pad Type	Glitch Prot	Over-voltage Prot	Under-voltage Prot	Backdrive Prot
C7	C7	GPIO147/I2C08_SDA/JTAG_CLK	VTR1	PIO	X		X	X
C8	C8	GPIO243	VTR1	PIO	X		X	X
C9	C9	GPIO155/I2C02_SCL/PS2_DAT1B	VTR1	PIO	X	X	X	X
C10	C10	GPIO010/I2C03_SCL/PS2_DAT0B	VTR1	PIO	X	X	X	X
C11	C11	GPIO124/GPTP-OUT6/PVT_CS#/KSO11	VTR1	PIO	X		X	X
C12		GPIO156/LED0	VTR1	PIO	X	X	X	X
	C12	GPIO153/LED2	VTR1	PIO	X	X	X	X
C13	C13	GPIO175/KSO17	VTR1	PIO	X		X	X
D1	D1	GPIO057/VCC_PWRGD	VTR1	PIO	X		X	X
D2	D2	GPIO060/KBRST/48MHZ_OUT	VTR1	PIO	X		X	X
D3	D3	RESETI#	VTR1	In	X		X	X
D11	D11	GPIO125/GPTP-OUT5/PVT_CLK/KSO12	VTR1	PIO	X		X	X
D12	D12	GPIO126/PVT_IO3/KSO13	VTR1	PIO	X		X	X
D13	D13	GPIO127/A20M/UART0_CTS#	VTR1	PIO	X		X	X
E1	E1	VTR_PLL		PWR				
E2	E2	GPIO022/GPTP-IN0	VTR1	PIO	X		X	X
E3	E3	GPIO106/PWROK	VTR1	PIO	X		X	X
E5	E5	VBAT		PWR				
E6	E6	GPIO163/VCI_IN0#	VBAT	PIO	X		X	X
E7	E7	GPIO144/I2C04_SCL/SPI1_CS#/UART0_RI#/TRACEDAT3	VTR1	PIO	X		X	X
E8	E8	GPIO045/KSO01	VTR1	PIO	X		X	X
E9	E9	GPIO046/BCM1_DAT/KSO02	VTR1	PIO	X		X	X
E11	E11	GPIO122/BCM0_DAT/PVT_IO1/KSO09	VTR1	PIO	X		X	X
E12	E12	GPIO123/BCM0_CLK/PVT_IO2/KSO10	VTR1	PIO	X		X	X
E13		GPIO134/PWM10/UART1_RTS#	VTR1	PIO	X		X	X
	E13	GPIO035/PWM8/CTOUT1	VTR1	PIO	X	X	X	X
F1	F1	VFLT_PLL		PWR				
F2	F2	GPIO226/LED3	VTR1	PIO	X	X	X	X
F3	F3	GPIO050/FAN_TACH0/GTACH0	VTR1	PIO	X	X	X	X
F5	F5	GPIO003/I2C00_SDA/SPI0_CS#	VTR1	PIO	X		X	X
F6	F6	VSS1		PWR				
F7	F7	GPIO142/I2C05_SCL/SPI1_MOSI/UART0_DSR#/TRACEDAT1	VTR1	PIO	X		X	X
F8	F8	GPIO150/I2C08_SCL/JTAG_TMS	VTR1	PIO	X		X	X
F9	F9	GPIO105/UART0_RX	VTR1	PIO	X		X	X
F11	F11	GPIO104/UART0_TX	VTR1	PIO	X		X	X
F12	F12	GPIO121/PVT_IO0/KSO08	VTR1	PIO	X		X	X

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	Interface		Notes
	C2	C2	C2	E2	E2	G4	GPTP-IN2	General Purpose Pass Through Port Input 2	
	C1	C1	C1	G7	D3	E1	GPTP-IN3	General Purpose Pass Through Port Input 3	
	M5	M5	M5	N6	N3	R6	GPTP-IN4	General Purpose Pass Through Port Input 4	
	J8	J8	J8	J9	N10	K11	GPTP-IN5	General Purpose Pass Through Port Input 5	
	M9	M9	M9	M10	N9	P11	GPTP-IN6	General Purpose Pass Through Port Input 6	
	N3	N3	N3	N8	M5	M7	GPTP-IN7	General Purpose Pass Through Port Input 7	
	L10	L10	L10	K9	L10	M10	GPTP-OUT0	General Purpose Pass Through Port Output 0	
	M11	M11	M11	M13	M11	M11	GPTP-OUT1	General Purpose Pass Through Port Output 1	
	M10	M10	M10	M11	N11	L11	GPTP-OUT2	General Purpose Pass Through Port Output 2	
	N10	N10	N10	N12	M10	R13	GPTP-OUT3	General Purpose Pass Through Port Output 3	
				A5	B4	E7	GPTP-OUT4	General Purpose Pass Through Port Output 4	
	D11	D11	D11	B12	C12	D14	GPTP-OUT5	General Purpose Pass Through Port Output 5	
	C11	C11	C11	B13	E11	E12	GPTP-OUT6	General Purpose Pass Through Port Output 6	
				B6	A6	A7	GPTP-OUT7	General Purpose Pass Through Port Output 7	
I2C/SMBus Interface									
D7	C6	C6	C6	B5	B6	D7	I2C00_SCL	SMB-I2C Controller Port 0 Clock	Note 14
B5	F5	F5	F5	C6	D7	B7	I2C00_SDA	SMB-I2C Controller Port 0 Data	Note 14
				B6	A6	A7	I2C01_SCL	SMB-I2C Controller Port 1 Clock	Note 14
				A5	B4	E7	I2C01_SDA	SMB-I2C Controller Port 1 Data	Note 14
B9	C9	C9	C9	C9	A11	E10	I2C02_SCL	SMB-I2C Controller Port 2 Clock	Note 14
A10	A12	A12	A12	A9	A12	B13	I2C02_SDA	SMB-I2C Controller Port 2 Data	Note 14
B10	C10	C10	C10	C10	B11	D11	I2C03_SCL	SMB-I2C Controller Port 3 Clock	Note 14
B11	B12	B12	B12	B10	A13	A14	I2C03_SDA	SMB-I2C Controller Port 3 Data	Note 14
D8	E7	E7	E7	D4	B5	E8	I2C04_SCL	SMB-I2C Controller Port 4 Clock	Note 14
A5	A7	A7	A7	A6	B7	A8	I2C04_SDA	SMB-I2C Controller Port 4 Data	Note 14
B6	F7	F7	F7	F7	A5	D8	I2C05_SCL	SMB-I2C Controller Port 5 Clock	Note 14
A6	B7	B7	B7	A7	A7	B8	I2C05_SDA	SMB-I2C Controller Port 5 Data	Note 14
M13	L11	L11	L11	K13	M13	J11	I2C06_SCL	SMB-I2C Controller Port 6 Clock	Note 14
N12	N11	N11	N11	L10	N12	R14	I2C06_SDA	SMB-I2C Controller Port 6 Data	Note 14
K5	M7	M7	M7	H7	N4	R8	I2C07_SCL	SMB-I2C Controller Port 7 Clock	Note 14
M8	N6	N6	N6	L7	N3	P8	I2C07_SDA	SMB-I2C Controller Port 7 Data	Note 14
D9	F8	F8	F8	E7	E7	A9	I2C08_SCL	SMB-I2C Controller Port 8 Clock	Note 14
B7	C7	C7	C7	C7	B8	D9	I2C08_SDA	SMB-I2C Controller Port 8 Data	Note 14
A7	G8	G8	G8	F8	D8	B9	I2C09_SCL	SMB-I2C Controller Port 9 Clock	Note 14

Agg IRQ	Agg Bit	HWB Instance Name	Interrupt Event	Wake Event	Source Description	Agg NVIC	Direct NVIC
GIRQ23	0	16-Bit Basic Timer 0	Timer_Event	No	Basic Timer Event	14	136
	1	16-Bit Basic Timer 1	Timer_Event	No	Basic Timer Event		137
	2	16-Bit Basic Timer 2	Timer_Event	No	Basic Timer Event		138
	3	16-Bit Basic Timer 3	Timer_Event	No	Basic Timer Event		139
	4	32-Bit Basic Timer 0	Timer_Event	No	Basic Timer Event		140
	5	32-Bit Basic Timer 1	Timer_Event	No	Basic Timer Event		141
	6	Counter/Timer 0	Timer_Event	No	16-bit Timer/Counter Event		142
	7	Counter/Timer 1	Timer_Event	No	16-bit Timer/Counter Event		143
	8	Counter/Timer 2	Timer_Event	No	16-bit Timer/Counter Event		144
	9	Counter/Timer 3	Timer_Event	No	16-bit Timer/Counter Event		145
	10	Capture Compare Timer	CAPTURE TIMER	No	CCT Counter Event		146
	11	Capture Compare Timer	CAPTURE 0	No	CCT Capture 0 Event		147
	12	Capture Compare Timer	CAPTURE 1	No	CCT Capture 1 Event		148
	13	Capture Compare Timer	CAPTURE 2	No	CCT Capture 2 Event		149
	14	Capture Compare Timer	CAPTURE 3	No	CCT Capture 3 Event		150
	15	Capture Compare Timer	CAPTURE 4	No	CCT Capture 4 Event		151
	16	Capture Compare Timer	CAPTURE 5	No	CCT Capture 5 Event		152
	17	Capture Compare Timer	COMPARE 0	No	CCT Compare 0 Event		153
	18	Capture Compare Timer	COMPARE 1	No	CCT Compare 1 Event		154
	19-31	Reserved					

Offset	14h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:1	Reserved	R	-	-
0	PWR_INV This bit allows firmware to control when the Host receives an indication that the VCC power is valid, by controlling the state of the PWROK pin. This bit is used by firmware to control the internal RESET_VCC signal function and the external PWROK pin. This bit is read-only when VCC_PWRGD is de-asserted low. The internal RESET_VCC signal is asserted when this bit is asserted even if the PWROK pin is configured as an alternate function.	R / R/W	1h	RESET_SYS

4.9.10 SYSTEM RESET REGISTER

Offset	18h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:9	Reserved	R	-	-
8	SOFT_SYS_RESET A write of a ‘1’ to this bit will force an assertion of the RESET_SYS reset signal, resetting the device. A write of a ‘0’ has no effect. Reads always return ‘0’.	W	-	-
7:0	Reserved	R	-	-

6.0 RAM AND ROM

6.1 References

None.

6.2 SRAM

The MEC170x contains two blocks of SRAM. he two SRAM blocks in the MEC170x total 480KB. Both SRAM blocks can be used for either program or data accesses. Performance is enhanced when program fetches and data accesses are to different SRAM blocks, but a program will operate correctly even if both program and data accesses are targeting the same block simultaneously.

- The first SRAM, which is optimized for code access, is 416KB
- The second SRAM, which is optimized for data access, is 64KB

6.3 ROM

The MEC170x contains a 64KB block of ROM, located at address 00000000h in the ARM address space. The ROM contains boot code that is executed after the de-assertion of RESET_SYS. The boot code loads an executable code image into SRAM. The ROM also includes a set of API functions that can be used for cryptographic functions, as well as loading SRAM with programs or data.

6.4 Additional Memory Regions

6.4.1 ALIAS RAM

The Alias RAM region, starting at address 20000000h, is an alias of the SRAM located at 118000h, and is the same size as that SRAM block. EC software can access memory in either the primary address or in the alias region; however, access is considerably slower to the alias region. The alias region exists in order to enable the ARM bit-band region located at address 20000000h.

6.4.2 RAM BIT-BAND REGION

The RAM bit-band region is an alias of the SRAM located at 118000h, except that each bit is aliased to bit 0 of a 32-bit doubleword in the bit-band region. The upper 31 bits in each doubleword of the bit-band region are always 0. The bit-band region is therefore 32 times the size of the SRAM region. It can be used for atomic updates of individual bits of the SRAM, and is a feature of the ARM architecture.

The bit-band region can only be accessed by the ARM processor. Accesses by any other bus master will cause a memory fault.

6.4.3 CRYPTOGRAPHIC RAM

The cryptographic RAM is used by the cryptographic API functions in the ROM

6.4.4 REGISTER BIT-BAND REGION

The Register bit-band region is an 32-to-1 alias of the device register space starting at address 40000000h and ending with the Host register space at 400FFFFF. Every bit in the register space is aliased to a byte in the Register bit-band region, and like the RAM bit-band region, can be used by EC software to read and write individual register bits. Only the EC Device Registers and the GPIO Registers can be accessed via the bit-band region.

A one bit write operation to a register bit in the bit-band region is implemented by the ARM processor by performing a read, a bit modification, followed by a write back to the same register. Software must be careful when using bit-banding if a register contains bits have side effects triggered by a read.

The bit-band region can only be accessed by the ARM processor. Accesses by any other bus master will cause a memory fault.

6.5 Memory Map

The memory map of the RAM and ROM is represented as follows:

7.9.8 DMA CHANNEL N INTERRUPT STATUS REGISTER

Offset	14h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:3	Reserved	R	-	-
2	STATUS_DONE This is an interrupt source register. This flags when the DMA Channel has completed a transfer successfully on its side. A completed transfer is defined as when the DMA Channel reaches its limit; **Memory Start Address** equals **Memory End Address**. A completion due to a **Hardware Flow Control Terminate** will not flag this interrupt. 1=MEMORY_START_ADDRESS equals MEMORY_END_ADDRESS 0=MEMORY_START_ADDRESS does not equal MEMORY_END_ADDRESS	R/WC	0h	RESET
1	STATUS_FLOW_CONTROL This is an interrupt source register. This flags when the DMA Channel has encountered a **Hardware Flow Control Request** after the DMA Channel has completed the transfer. This means the Master Device is attempting to overflow the DMA. 1=Hardware Flow Control is requesting after the transfer has completed 0=No Hardware Flow Control event	R/WC	0h	RESET
0	STATUS_BUS_ERROR This is an interrupt source register. This flags when there is an Error detected over the internal 32-bit Bus. 1=Error detected.	R/WC	0h	RESET

7.9.9 DMA CHANNEL N INTERRUPT ENABLE REGISTER

Offset	18h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:3	Reserved	R	-	-
2	STATUS_ENABLE_DONE This is an interrupt enable for STATUS_DONE. 1=Enable Interrupt 0=Disable Interrupt	R/W	0h	RESET
1	STATUS_ENABLE_FLOW_CONTROL_ERROR This is an interrupt enable for STATUS_FLOW_CONTROL. 1=Enable Interrupt 0=Disable Interrupt	R/W	0h	RESET

TABLE 10-5: ESPI I/O BASE ADDRESS REGISTER DEFAULT VALUES (CONTINUED)

Host Config Index	EC Offset	Logical Device	Reset Default	EC-Only Offset	Reset Default	LDN	Mask (Note 1)
6Ch	36Ch	Embedded Memory Interface (EMI) 1	0000_0000h	16Ch	0000_110Fh	11h	Fh
70h	370h	Embedded Memory Interface (EMI) 2	0000_0000h	170h	0000_120Fh	12h	Fh
74h	374h	BIOS Debug Port (Port 80) 0	0000_0000h	174h	0000_2000h	20h	0h
78h	378h	BIOS Debug Port (Port 80) 1	0000_0000h	178h	0000_2100h	21h	0h
7Ch	37Ch	RTC	0000_0000h	17Ch	0000_141Fh	14h	1Fh
Note 1: The ACPI-EC0 Mask bit field is a read/write bit field. All other MASK bit fields are read-only as defined in the register description							

TABLE 10-6: ESPI MEMORY BASE ADDRESS REGISTER DEFAULT VALUES

Host Config Index	EC Offset	Logical Device	Reset Default	EC-Only Offset	Reset Default	LDN	Mask (Note 1)
30h	330h	Mailbox	00_0000h	130h	0000_0001h	0h	1h
3Ah	33Ah	ACPI EC Channel 0	62_0000h	13Ah	0000_0204h	2h	4h
44h	344h	ACPI EC Channel 1	00_0000h	144h	0000_0307h	3h	7h
4Eh	34Eh	ACPI EC Channel 2	00_0000h	14Eh	0000_0407h	4h	7h
58h	358h	ACPI EC Channel 3	00_0000h	158h	0000_0507h	5h	7h
62h	362h	ACPI EC Channel 4	00_0000h	162h	0000_0607h	6h	7h
6Ch	376h	Embedded Memory Interface (EMI) 0	00_0000h	16Ch	0000_100Fh	10h	Fh
76h	380h	Embedded Memory Interface (EMI) 1	00_0000h	176h	0000_110Fh	11h	Fh
80h	38Ah	Embedded Memory Interface (EMI) 2	00_0000h	180h	0000_120Fh	12h	Fh
Note 1: The ACPI-EC0 Mask bit field is a read/write bit field. All other MASK bit fields are read-only as defined in the register description							

TABLE 10-7: SRAM BASE ADDRESS REGISTER DEFAULT VALUES, HOST CONFIG

Host Config Index	EC Offset	Logical Device	Reset Default	LPC Host Address [79:16]	Size [7:4]	Access [2:1]
ACh	3ACh	SRAM BAR 0	0h	0h	0h	0h
B6h	3B6h	SRAM BAR 1	0h	0h	0h	0h

TABLE 10-8: SRAM BASE ADDRESS REGISTER DEFAULT VALUES, EC-ONLY

EC Offset	Logical Device	Reset Default	Base Address [47:16]	Size [7:4]	Access [2:1]	Valid [0]
1ACh	SRAM BAR 0	80h	0h	0h	0h	1h
1B6h	SRAM BAR 1	80h	0h	0h	0h	1h

12.15.5 EC AUX DATA REGISTER

Offset	10Ch			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:0	EC_AUX_DATA This 8-bit register is write-only. When written, the C/D in the Keyboard Status Read Register is cleared to '0', signifying data, and the IBF in the same register is set to '1'. When the Runtime Register at offset 0h is read by the Host, it functions as the EC_HOST Data / AUX Data Register.	W	0h	RESET_SYS

12.15.6 PCOBF REGISTER

Offset	114h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:1	Reserved	R	-	-
0	PCOBF For a description of this bit, see Section 12.10.1, "PCOBF Description".	R/W	0h	RESET_SYS

12.16 Legacy Port92/GATEA20 Configuration Registers

Configuration Registers for an instance of the Port92-Legacy block are listed in the following table. Host access to Configuration Registers is through the Configuration Port using the Logical Device Number of the Port 92 instance and the Index shown in the "Host Index" column of the table. The EC can access Configuration Registers directly. The EC address for each register is formed by adding the Base Address for each instance of the Port 92 block shown in the Block Overview and Base Address Table in Section 3.0, "Device Inventory" to the offset shown in the "EC Offset" column.

TABLE 12-13: CONFIGURATION REGISTER SUMMARY

EC Offset	Host Index	Register Name
330h	30h	Port 92 Enable Register

18.3 Interrupts

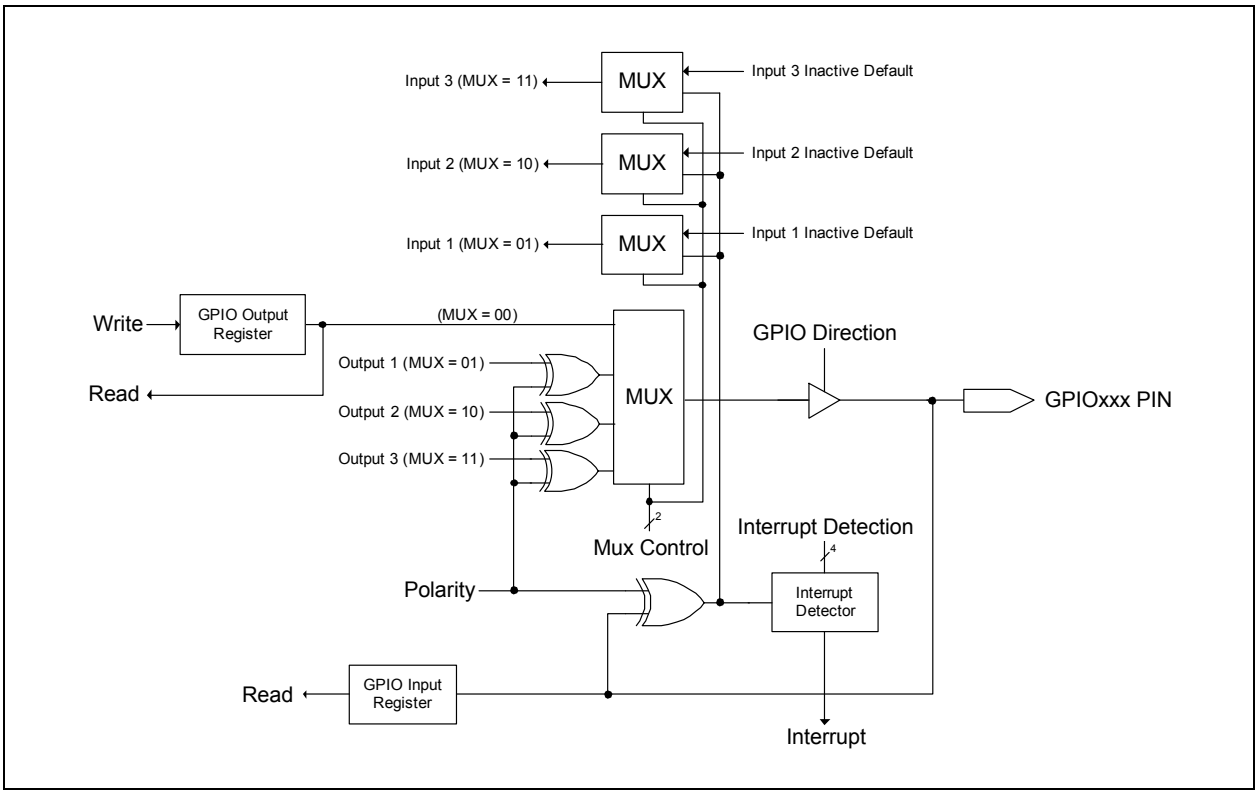
This section defines the Interrupt Sources generated from this block.

TABLE 18-4: INTERRUPTS

Source	Description
GPIO_Event	Each GPIO pin has the ability to generate an interrupt event. This event may be used as a wake event. The event can be generated on a high level, low level, rising edge, falling edge or either edge input, as configured by the INTERRUPT_DETECTION bits in the Pin Control Registers associated with the GPIO signal function. The minimum pulse width to generate in interrupt / wakeup event must be at least 5ns.

18.4 Description

FIGURE 18-1: GPIO BLOCK DIAGRAM



18.5 GPIO Pass-Through Ports

GPIO Pass-Through Ports (GPTP) can multiplex two general purpose I/O pins as shown in Figure 18-2. GPIO Pass-Through Ports connect the GPTP-IN pin to the GPTP-OUT pin. The GPTP are sequentially assigned values 0:7. The GPTP port assignment have no relation to the GPIO Indexing assignments. The GPTP ports are controlled by the Mux Control bits in the Pin Control Register associated with the GPTP-OUT signal function.

In order to enable the GPTP Pass-Through Mode, the GPTP-IN (GPIOm in Figure 18-2) Pin Control Register must assign the Mux Control to the GPTP_IN signal function and the GPIO Direction bit to 0 (input); the GPTP-OUT (GPIOn in Figure 18-2) Pin Control Register must assign the Mux Control to the GPTP_OUT signal function and the GPIO Direction bit to 1 (output). The GPTP-OUT signal function can differ from pin to pin.

TABLE 21-8: MEASUREMENT MODE OPERATIONAL SUMMARY (CONTINUED)

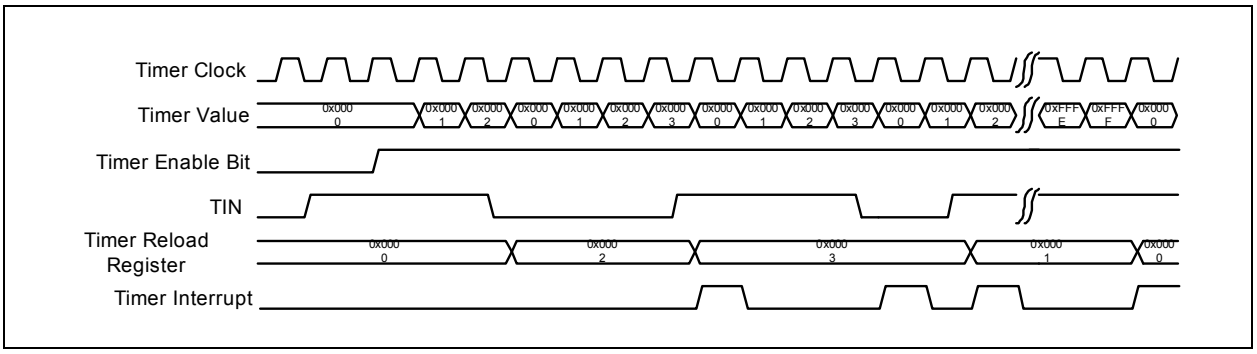
Item	Description
Count Operation	• Up Count • At measurement pulse's effective edge, the count value is transferred to the Timer Reload Register and the timer is loaded with 0000h and continues counting.
Count Start Condition	• Timer enable is set (ENABLE = 1)
Count Stop Condition	• Timer is reset (RESET = 1) • Timer overflows • Timer enable is cleared (ENABLE = 0)
Interrupt Request Generation Timing	• When timer overflows • When a measurement pulse’s effective edge is input. (An interrupt is not generated on the first effective edge after the timer is started.)
TINx Pin Function	Programmable Input port or Measurement input
Read From Timer	When the Timer x Reload Register is read it indicates the measurement result from the last measurement made. The Timer x Reload Register reads 0000h if the timer overflows before a measurement is made.
Write to Timer	Timer x Reload Register is Read-Only in Measurement mode

21.10.8.1 Pulse Width Measurements

The timers measure pulse width by counting the number of timer clocks since the last rising or falling edge of the TINx input. To measure the pulse width of a signal on the TINx pin, the EDGE bits in the Clock and Event Control Register, must be set to start counting on rising and falling edges. The timer starts measuring on the next edge (rising or falling) on the TINx pin after the ENABLE bit is set. The Reload register stores the result of the last measurement taken. If the timer overflows, 0x0000 is written to the Reload register and the ENABLE bit is cleared stopping the timer. Figure 21-11 shows the timer behavior when measuring pulse widths.

The timer will not assert an interrupt in Pulse Measurement mode until the timer detects both a rising and a falling edge.

FIGURE 21-11: PULSE WIDTH MEASUREMENT



21.10.8.2 Period Measurements

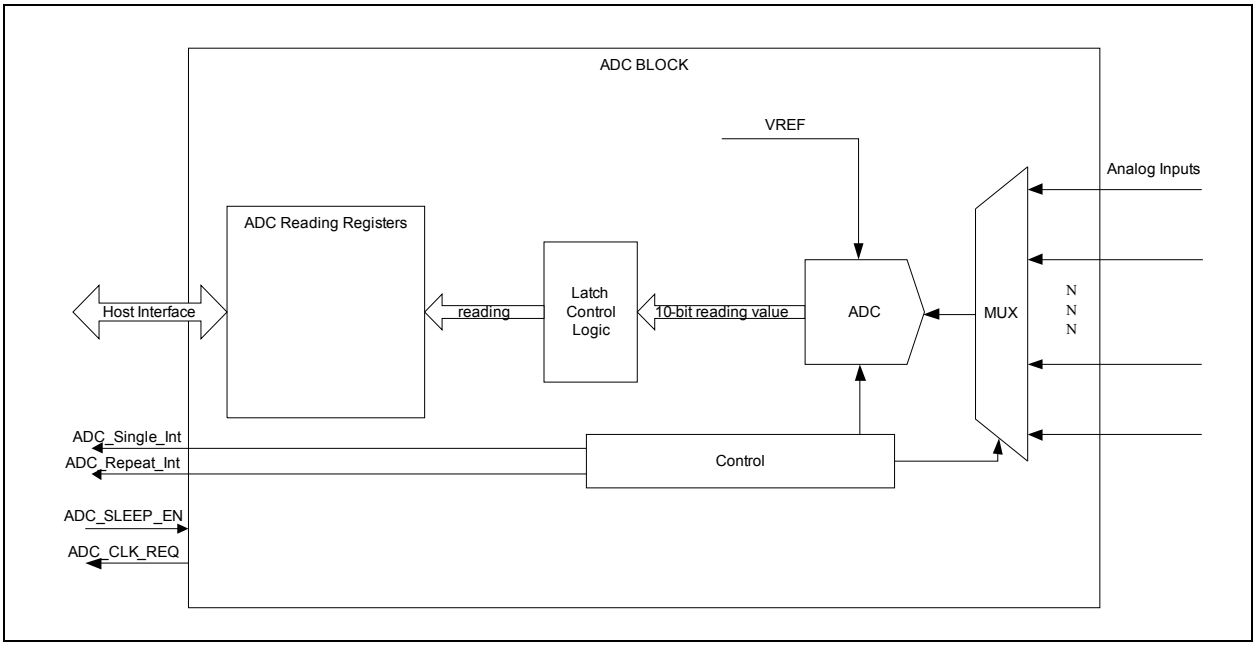
The 16-bit timer measures the period of a signal by counting the number of timer clocks between either rising or falling edges of the TINx input. The measurement edge is determined by the EDGE bits in the Clock and Event Control Register. The timer starts measuring on the next edge (rising or falling) on the TINx pin after the ENABLE bit is set. The reload register stores the result of the last measurement taken. If the timer overflows, 0x0000 is written to the reload register. Figure 21-12 shows the timer behavior when measuring the period of a signal.

The timer will not signal an interrupt in period measurement mode until the timer detects either two rising edges or two falling edges.

Offset	00h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
17	COMPARE_SET0 When read, returns the current value off the Compare Timer Output 0 state. • If written with a '1b', the output state is set to '1'. • Writes of '0b' have no effect	R/WS	0	RESET_SYS
16	COMPARE_SET1 When read, returns the current value off the Compare Timer Output 1 state. If written with a '1b', the output state is set to '1'. Writes of '0b' have no effect	R/WS	0	RESET_SYS
15:10	Reserved	R	-	-
9	COMPARE_ENABLE1 Compare Enable for Compare 1 Register. When enabled, a match between the Compare 1 Register and the Free Running Timer Register will cause the TOUT1 output to toggle and will send a COMPARE event to the Interrupt Aggregator. 1=Enabled 0=Disabled	R/W	0b	RESET_SYS
8	COMPARE_ENABLE0 Compare Enable for Compare 0 Register. When enabled, a match between the Compare 0 Register and the Free Running Timer Register will cause the TOUT0 output to toggle and will send a COMPARE event to the Interrupt Aggregator. 1=Enabled 0=Disabled	R/W	0b	RESET_SYS
7	Reserved	R	-	-
6:4	TCLK This 3-bit field sets the clock source for the Free-Running Counter. See Table 22-2, "Timer Clock Frequencies" for a list of available frequencies.	R/W	0b	RESET_SYS
3	Reserved	R	-	-
2	FREE_RESET Free Running Timer Reset. This bit stops the timer and resets the internal counter to 0000_0000h. This bit does not affect the FREE_ENABLE bit. This bit is self clearing after the timer is reset. 1=Timer reset 0=Normal timer operation	R/W	0h	RESET_SYS

30.10 Description

FIGURE 30-2: ADC BLOCK DIAGRAM



The MEC170x features a sixteen channel successive approximation Analog to Digital Converter. The ADC architecture features excellent linearity and converts analog signals to 10 bit words. Conversion takes less than 1.125 microseconds per 10-bit word. The sixteen channels are implemented with a single high speed ADC fed by a sixteen input analog multiplexer. The multiplexer cycles through the sixteen voltage channels, starting with the lowest-numbered channel and proceeding to the highest-number channel, selecting only those channels that are programmed to be active.

The input range on the voltage channels spans from 0V to the voltage reference. With an voltage reference of 3.3V, this provides resolutions of 3.2mV. The range can easily be extended with the aid of resistor dividers. The accuracy of any voltage reading depends on the accuracy and stability of the voltage reference input.

Note:	The ADC pins are 3.3V tolerant.

The ADC conversion cycle starts either when the START_SINGLE bit in the ADC to set to 1 or when the ADC Repeat Timer counts down to 0. When the START_SINGLE is set to 1 the conversion cycle converts channels enabled by configuration bits in the ADC Single Register. When the Repeat Timer counts down to 0 the conversion cycle converts channels enabled by configuration bits in the ADC Repeat Register. When both the START_SINGLE bit and the Repeat Timer request conversions the START_SINGLE conversion is completed first.

Conversions always start with the lowest-numbered enabled channel and proceed to the highest-numbered enabled channel.

Note:	If software repeatedly sets Start_Single to 1 at a rate faster than the Repeat Timer count down interval, the conversion cycle defined by the ADC Repeat Register will not be executed.

30.10.1 REPEAT MODE

- Repeat Mode will start a conversion cycle of all ADC channels enabled by bits RPT_EN in the ADC Repeat Register. The conversion cycle will begin after a delay determined by START_DELAY in the ADC Delay Register.
- After all channels enabled by RPT_EN are complete, REPEAT_DONE_STATUS will be set to 1. This status bit is cleared when the next repeating conversion cycle begins to give a reflection of when the conversion is in progress.

30.11.3 ADC STATUS REGISTER

The ADC Status Register indicates whether the ADC has completed a conversion cycle.

Offset	08h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:16	Reserved	R	-	-
15:0	ADC_CH_STATUS All bits are cleared by being written with a '1'. 1=conversion of the corresponding ADC channel is complete 0=conversion of the corresponding ADC channel is not complete For enabled single cycles, the SINGLE_DONE_STATUS bit in the ADC Control Register is also set after all enabled channel conversion are done; for enabled repeat cycles, the REPEAT_DONE_STATUS in the ADC Control Register is also set after all enabled channel conversion are done.	R/WC	00h	RESET_SYS

30.11.4 ADC SINGLE REGISTER

The ADC Single Register is used to control which ADC channel is captured during a Single-Sample conversion cycle initiated by the START_SINGLE bit in the ADC Control Register.

Note: Do not change the bits in this register in the middle of a conversion cycle to insure proper operation.

Offset	0Ch			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:16	Reserved	R	-	-
15:0	SINGLE_EN Each bit in this field enables the corresponding ADC channel when a single cycle of conversions is started when the START_SINGLE bit in the ADC Control Register is written with a 1. 1=single cycle conversions for this channel are enabled 0=single cycle conversions for this channel are disabled	R/W	0h	RESET_SYS

30.11.5 ADC REPEAT REGISTER

The ADC Repeat Register is used to control which ADC channels are captured during a repeat conversion cycle initiated by the START_REPEAT bit in the ADC Control Register.

Offset	10h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:16	Reserved	R	-	-
15:0	RPT_EN Each bit in this field enables the corresponding ADC channel for each pass of the Repeated ADC Conversion that is controlled by bit START_REPEAT in the ADC Control Register. 1=repeat conversions for this channel are enabled 0=repeat conversions for this channel are disabled	R/W	00h	RESET_SYS

Offset	05h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
3:2	GAINI The integral gain term. Gain Factor: 3=8x 2=4x 1=2x 0=1x	R/W	2h	RESET_SYS
1:0	GAINP The proportional gain term. Gain Factor: 3=8x 2=4x 1=2x 0=1x	R/W	2h	RESET_SYS

31.9.5 FAN SPIN UP CONFIGURATION REGISTER

Offset	06h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:6	DRIVE_FAIL_CNT Determines how many update cycles are used for the Drive Fail detection function. This circuitry determines whether the fan can be driven to the desired Tach target. These settings only apply if the Fan Speed Control Algorithm is enabled. 3=Drive Fail detection circuitry will count for 64 update periods 2=Drive Fail detection circuitry will count for 32 update periods 1=Drive Fail detection circuitry will count for 16 update periods 0=Drive Fail detection circuitry is disabled	R/W	00b	RESET_SYS
5	NOKICK Determines if the Spin Up Routine will drive the fan to 100% duty cycle for 1/4 of the programmed spin up time before driving it at the programmed level. 1=The Spin Up Routine will not drive the PWM to 100%. It will set the drive at the programmed spin level for the entire duration of the programmed spin up time 0=The Spin Up Routine will drive the PWM to 100% for 1/4 of the programmed spin up time before reverting to the programmed spin level	R/W	0b	RESET_SYS

TABLE 33-3: BLINKING MODE CALCULATIONS

Parameter	Unit	Equation
Frequency	Hz	(32KHz frequency) /(PRESCALE + 1)/256
'H' Width	Seconds	(1/Frequency) x (DutyCycle/256)
'L' Width	Seconds	(1/Frequency) x ((1-DutyCycle)/256)

33.8.3 GENERAL PURPOSE PWM

When used in the Blinking configuration with the 48MHz, the LED module can be used as a general-purpose programmable Pulse-Width Modulator with an 8-bit programmable pulse width. It can be used for fan speed control, sound volume, etc. With the 48MHz source, the PWM frequency can be configured in the range shown in Table 33-4.

TABLE 33-4: PWM CONFIGURATION EXAMPLES

Prescale	PWM Frequency
000h	187.5 KHz
001h	94 KHz
003h	47 KHz
006h	26.8 KHz
00Bh	15.625 KHz
07Fh	1.46 KHz
1FFh	366 Hz
FFFh	46 Hz

TABLE 33-5: GENERAL PURPOSE PWM MODE CALCULATIONS

Parameter	Unit	Equation
Frequency	Hz	(48MHz frequency) / (PRESCALE + 1) / 256
'H' Width	Seconds	(1/Frequency) x (DutyCycle/256)
'L' Width	Seconds	(1/Frequency) x (256 - DutyCycle)

33.8.3.1 PWM WDT

When the PWM is configured as a general-purpose PWM (in the Blinking configuration with the Main system clock), the PWM includes a Watch Dog Timer (WDT). The WDT consists of an internal 8-bit counter and an 8-bit reload value (the field WDTLD in LED Configuration Register register). The internal counter is loaded with the reset value of WDTLD (14h, or 4 seconds) on system RESET_SYS and loaded with the contents of WDTLD whenever either the LED Configuration Register register is written or the MIN byte in the LED Limits Register register is written (the MIN byte controls the duty cycle of the PWM).

Whenever the internal counter is non-zero, it is decremented by 1 for every tick of the 5 Hz clock. If the counter decrements from 1 to 0, a WDT Terminal Count causes an interrupt to be generated and reset sets the CONTROL bit in the LED Configuration Register to 3h, which forces the PWM to be full on. No other PWM registers or fields are affected.

If the 5 Hz clock halts, the watchdog timer stops decrementing but retains its value, provided the device continues to be powered. When the 5 Hz clock restarts, the watchdog counter will continue decrementing where it left off.

Setting the WDTLD bits to 0 disables the PWM WDT. Other sample values for WDTLD are:

01h = 200 ms

02h = 400 ms

03h = 600 ms

04h = 800 ms

. . .

14h = 4seconds

FFh = 51 seconds

34.11 Time Constants

This section lists a set of R and C values which can be connected to the RC_ID pin. Note that risetime generally follow RC time Tau. Firmware should use the Max and Min Counts in the tables to create quantized states.

In the following tables, the CLOCK_SET field in the RC_ID Control Register is set to '1', so the time base for measuring the rise time is 24MHz, the speed of the system clock. All capacitor values are ±10% and all resistor values are ±5%. Minimum and maximum count values are suggested ranges, calculated to provide reasonable margins around the nominal rise times. Rise times have been confirmed by laboratory measurements.

TABLE 34-2: SAMPLE RC VALUES, C=2200PF @24MHZ

R (KΩ)	Nominal Tau (µS)	Minimum Count	Maximum Count
1	2.2	60.00	72.00
2	4.4	115.00	140.00
4.3	9.5	241.00	294.00
8.2	18.04	456.00	557.00
33	72.6	1819.00	2224.00
62	136.4	3456.00	4224.00
130	286	7470.00	9130.00
240	528	14400.00	17600.00

TABLE 34-3: SAMPLE RC VALUES, C=3000PF @24MHZ

R (KΩ)	Nominal Tau (µS)	Minimum Count	Maximum Count
1	3	77.00	95.00
2	6	151.00	184.00
4.3	12.9	320.00	391.00
8.2	24.6	604.00	739.00
33	99	2439.00	2981.00
62	186	4647.00	5680.00
130	390	9990.00	12210.00
240	720	193508.00	23650.00

TABLE 34-4: SAMPLE RC VALUES, C=4700PF @24MHZ

R (KΩ)	Nominal Tau (µS)	Minimum Count	Maximum Count
1	4.7	116.00	142.00
2	9.4	229.00	280.00
4.3	20.2	495.00	605.00
8.2	38.5	945.00	1160.00
33	155.1	3780.00	4650.00
62	291.4	7249.00	8859.00
130	611	15480.00	18920.00
240	1128	29880.00	36520.00

37.5 Signal Description

See the Pin Description chapter for the pins and the signal names associated with the following signals.

TABLE 37-1: EXTERNAL SIGNAL DESCRIPTION

Name	Direction	Description
SP_DIN	Input	Serial Data In pin
SP_DOUT	Input/Output	Serial Data Output pin. Switches to input when used in double-data-rate mode
SP_CLK	Output	SPI Clock output used to drive the SPCLK pin.
SP_CS#	Output	SPI chip select

TABLE 37-2: INTERNAL SIGNAL DESCRIPTION

Name	Direction	Description
SPI_TDMA_REQ	Output	DMA Request control for GP-SPI Controller Transmit Channel
SPI_RDMA_REQ	Output	DMA Request control for GP-SPI Controller Receive Channel

37.6 Host Interface

The registers defined for the General Purpose Serial Peripheral Interface are accessible by the various hosts as indicated in Section 37.12, "EC-Only/Runtime Registers".

37.7 Power, Clocks and Reset

This section defines the Power, Clock, and Reset parameters of the block.

37.7.1 POWER DOMAINS

Name	Description
VTR	The logic and registers implemented in this block are powered by this power well.

37.7.2 CLOCK INPUTS

Name	Description
48MHz	This is a clock source for the SPI clock generator.
2MHz	This is a clock source for the SPI clock generator. It is derived from the 48MHz clock domain.

37.7.3 RESETS

Name	Description
RESET_SYS	This signal resets all the registers and logic in this block to their default state.

37.8 Interrupts

This section defines the Interrupt Sources generated from this block.

TABLE 37-3: EC INTERRUPTS

Source	Description
TXBE_STS	Transmit buffer empty status (TXBE), in the SPI Status Register, sent as an interrupt request to the Interrupt Aggregator.
RXBF_STS	Receive buffer full status (RXBF), in the SPI Status Register, sent as an interrupt request to the Interrupt Aggregator.

The Port 80 BIOS Debug Port consists of a 32-bit wide x 16 deep FIFO and a 24-bit free running timer. Host and EC access to the Port 80 device is through a set of registers. The Host can write the FIFO via the Runtime Registers and the EC can read the FIFO can control the device via the EC Registers.

Writes to the Host Data Register are concatenated with the 24-bit timestamp and written to the FIFO. Reads of the Host Data Register return zero. If writes to the Host Data Register overrun the FIFO, the oldest data are discarded and the OVERRUN status bit in the Status Register is asserted.

Only the EC can read data from the FIFO, using the EC Data Register. The use of this data is determined by EC Firmware alone.

42.10 Configuration Registers

Configuration Registers for an instance of the Port 80 BIOS Debug Port are listed in the following table. Host access to Configuration Registers is through the Configuration Port using the Logical Device Number of each instance of the Port 80 BIOS Debug Port and the Index shown in the “Host Index” column of the table. The EC can access Configuration Registers directly. The EC address for each register is formed by adding the Base Address for each instance of the Port 80 BIOS Debug Port shown in the Block Overview and Base Address Table in Section 3.0, "Device Inventory" to the offset shown in the “EC Offset” column.

TABLE 42-1: CONFIGURATION REGISTER SUMMARY

EC Offset	Host Index	Register Name
330h	30h	Activate Register

42.10.1 ACTIVATE REGISTER

Offset	330h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:1	Reserved	R	-	-
0	ACTIVATE When this bit is asserted ‘1’, the block is enabled. When this bit is ‘0’, writes by the Host interface to the Host Data Register are not claimed, the FIFO is flushed, the 24-bit Timer is reset, and the timer clock is stopped. Control bits in the Configuration Register are not affected by the state of ACTIVATE.	R/W	0h	RESET_SYS

42.11 Runtime Registers

The registers listed in the Runtime Register Summary table are for a single instance of the Port 80 BIOS Debug Port. Host access for each register listed in this table is defined as an offset in the Host address space to the Block’s Base Address, as defined by the instance’s Base Address Register.

The EC address for each register is formed by adding the Base Address for each instance of the Port 80 BIOS Debug Port shown in the Block Overview and Base Address Table in Section 3.0, "Device Inventory" to the offset shown in the “Offset” column.

The Port 80 BIOS Debug Port can be accessed from the LPC Host Interface, the ESPI Host Interface, or the internal embedded controller (EC). The following table summarizes the host access types supported for each interface.

51.11.1 LPC BUS TIMING

FIGURE 51-13: OUTPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS

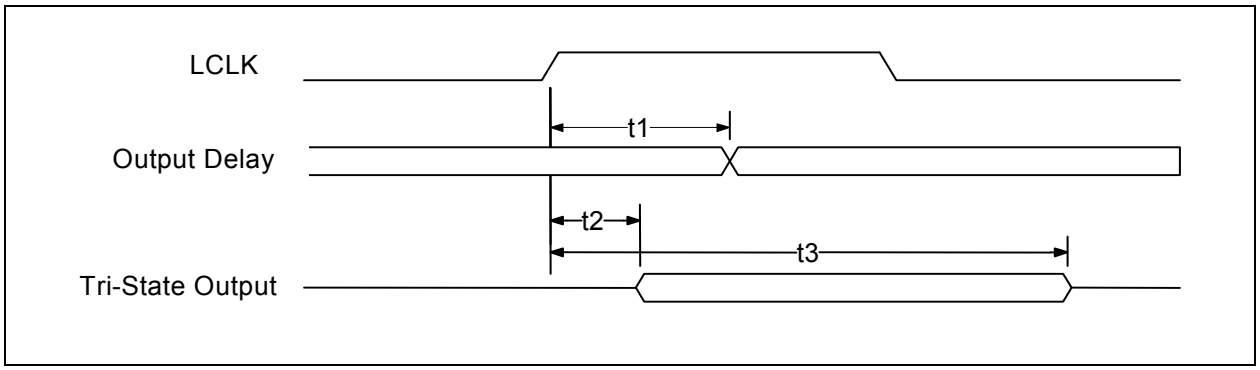


TABLE 51-12: OUTPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	LCLK to Signal Valid Delay – Bused Signals	2		11	ns
t2	Float to Active Delay				
t3	Active to Float Delay			28	

51.11.2 LPC INPUT TIMING

FIGURE 51-14: INPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS

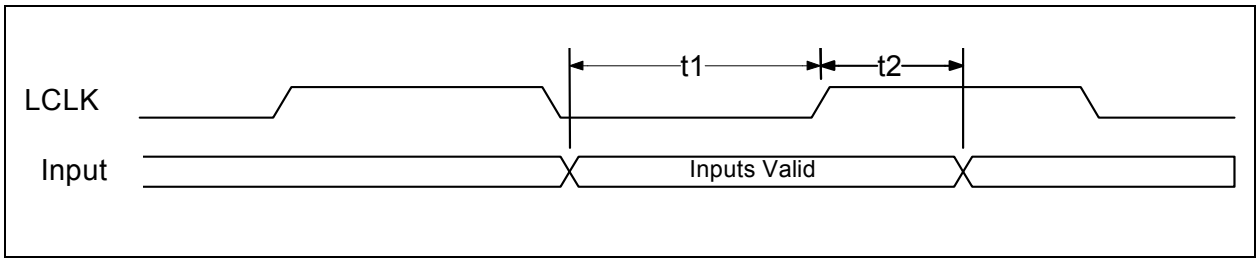


TABLE 51-13: INPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Input Set Up Time to LCLK – Bused Signals	7			ns
t2	Input Hold Time from LCLK	0			