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	MIPS32® M14K™
Program Memory Type	External Program Memory
Controller Series	-
RAM Size	192KB
Interface	I²C, LPC, SMBus, SPI, UART
Number of I/O	106
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/mec1418-i-sz

Note	Description
Note 15	The KSI and KSO Key Scan pins require pull-up resistors. The system designer may opt to use either use the internal pull-up resistors or populate external pull-up resistors.
Note 16	If the eSPI Flash Channel is used for booting, the GPIO123/SHD_CS# pin must be used as RSMRST#. This pin will be driven high by the boot ROM code in order to activate the eSPI flash channel. If the SHD_SPI port is used for booting, then any unused GPIO may be used for RSMRST#.
Note 17	If the eSPI Flash Channel is used for booting, the GPIO135/SHD_IO2 pin must be used to determine that the primary power rails are stable before RSMRST# can be de-asserted. See the MEC140X/1X eSPI Addendum document for more details.
Note 18	If certain blocks are not used, then the associated voltage reference pin may be connected to ground, as follows: • if the ADC is not used and the block is disabled, ADC_VREF can be connected to VSS • if the DAC is not used and the block is disabled, DAC_VREF can be connected to VSS • if both PECI and SB TSI are not used and the GPIO033/PECI_DAT/SB_TSI_DAT and GPIO035/ SB-TSI_CLK pins are configured as GPIOs, then VREF_CPU can be connected to VSS.

2.4 Pin Lists

Note: The GPIO Pin Control registers for the Pads that are not bonded out to pins or balls in the smaller package have been defaulted to their inactive state and are read-only. These pins cannot be modified by the downloaded firmware located in SRAM. No special handling required.

2.4.1 MEC140X PIN LIST

MEC140x		
128-pin VTQFP	**144-pin WFBGA**	**Pin Name**
1	L10	GPIO157/LED0/TST_CLK_OUT
2	N13	GPIO027/KSO00/PVT_IO1
3	M12	GPIO001/SPI_CS#/32KHZ_OUT
4	M10	GPIO002/PWM7
5	G5	VTR
6	M13	GPIO005/SMB00_DATA/SMB00_DATA18/KSI2
7	L12	GPIO006/SMB00_CLK/SMB00_CLK18/KSI3
8	K11	GPIO007/SMB01_DATA/SMB01_DATA18
9	J11	GPIO010/SMB01_CLK/SMB01_CLK18
10	G9	GPIO011/nSMI/nEMI_INT
11	J7	GPIO012/SMB02_DATA/SMB02_DATA18
12	H12	GPIO013/SMB02_CLK/SMB02_CLK18
13	H8	nRESET_IN/GPIO014
14	L11	GPIO015/KSO01/PVT_CS#
15	H11	GPIO016/KSO02/PVT_SCLK
16	J12	GPIO017/KSO03/PVT_IO0
17	C9	VSS
18	F1	VR_CAP
19	H5	VTR
20	G11	GPIO020/CMP_VIN0

4.10.2 DATA REGISTER

Offset	01h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:0	DATA The DATA register, which is part of the Configuration Port, is used to read or write data to the register currently being selected by the INDEX Register. **Note:** For a description of accessing the Configuration Port see Section 4.8.3, "Configuration Port," on page 105	R/W	0h	nSYSRST

4.11 EC-Only Registers

Note: EC-Only registers are not accessible by the LPC interface.

The registers listed in Table 4-20, "EC-Only Register Summary" are for a single instance of the LPC Interface. Their addresses are defined as a relative offset to the host base address defined in TABLE 4-19:.

The following table defines the fixed host base address for each LPC Interface instance.

TABLE 4-19: EC-ONLY REGISTER ADDRESS RANGE TABLE

INSTANCE NAME	INSTANCE NUMBER	HOST	ADDRESS SPACE	BEGIN ADDRESS
LPC Interface	0	EC	32-bit internal address space	000F_3100h

Note: The Begin Address indicates where the first register can be accessed in a particular address space for a block instance.

TABLE 4-20: EC-ONLY REGISTER SUMMARY

Offset	Register Name
04h	LPC Bus Monitor Register
08h	Host Bus Error Register
0Ch	EC SERIRQ Register
10h	EC Clock Control Register
14h	Test Register
18h	Test Register
20h	I/O BAR Inhibit Register
24h	Reserved
28h	Reserved
2Ch	Reserved
30h	LPC BAR Init Register
40h	Device Memory BAR Inhibit Register
FCh	SRAM Memory Host Configuration Register

Note 4-13 Some Test registers are read/write registers. Modifying these registers may have unwanted results.

Offset	08h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
2	BAR_CONFLICT This bit is set to 1 whenever a BAR conflict occurs on an LPC address. A Bar conflict occurs when more than one BAR matches the address during of an LPC cycle access. Once this bit is set, it remains set until cleared by being written with a 1.	R/WC	0h	nSYSRST
1	EN_INTERNAL_ERR When this bit is 0, only a BAR conflict, which occurs when two BARs match the same LPC I/O address, will cause LPC_INTERNAL_ERR to be set. When this bit is 1, internal bus errors will also cause LPC_INTERNAL_ERR to be set.	R/W	0h	nSYSRST
0	LPC_INTERNAL_ERR This bit is set whenever a BAR conflict or an internal bus error occurs as a result of an LPC access. Once set, it remains set until cleared by being written with a 1. This signal may be used to generate interrupts. See Section 4.6, "Interrupts," on page 98.	R/WC	0h	nSYSRST

4.11.3 EC SERIRQ REGISTER

Offset	0Ch			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:1	RESERVED	RES	-	-
0	EC_IRQ If the LPC Logical Device is selected as the source for a Serial Interrupt Request by an Interrupt Configuration register (see Section 4.8.4.8, "SERIRQ Interrupts," on page 110), this bit is used as the interrupt source.	R/W	0h	nSYSRST

4.11.4 EC CLOCK CONTROL REGISTER

Offset	10h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:3	RESERVED	RES	-	-
2	Handshake This bit controls throughput of LPC transactions. When this bit is a ‘0’ the part supports a 33MHz PCI Clock. When this bit is a ‘1’, the part supports a PCI Clock from 24MHz to 33MHz.	RES	1h	nSYSRST

Offset	30h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
6	RX_TRANSFER_ENABLE This bit enables the receive function of the SPI interface. 1=Receive is enabled. Data received from the SPI Slave is stored in the Receive Buffer 0=Receive is disabled	R/W	0h	RESET
5:4	TX_DMA_ENABLE This bit enables DMA support for Transmit Transfer. If enabled, DMA will be requested to fill the FIFO until either the interface reaches TRANSFER_LENGTH or the DMA sends a termination request. The size defined here must match DMA programmed access size. 1=DMA is enabled.and set to 1 Byte 2=DMA is enabled and set to 2 Bytes 3=DMA is enabled and set to 4 Bytes 0=DMA is disabled. All data in the Transmit Buffer must be emptied by firmware	R/W	0h	RESET
3:2	TX_TRANSFER_ENABLE This field bit selects the transmit function of the SPI interface. 3=Transmit Enabled in 1 Mode. The MOSI or IO Bus will send out only 1's. The Transmit Buffer will not be used 2=Transmit Enabled in 0 Mode. The MOSI or IO Bus will send out only 0's. The Transmit Buffer will not be used. 1=Transmit Enabled. Data will be fetched from the Transmit Buffer and sent out on the MOSI or IO Bus. 0=Transmit is Disabled. No data is sent. This will cause the MOSI be to be undriven, or the IO bus to be undriven if Receive is also disabled.	R/W	0h	RESET
1:0	INTERFACE_MODE This field sets the transmission mode. If this field is set for Dual Mode or Quad Mode then either TX_TRANSFER_ENABLE or RX_TRANSFER_ENABLE **must** be 0. 3=Reserved 2=Quad Mode 1=Dual Mode 0=Single/Duplex Mode	R/W	0h	RESET

6.11.12 QMSPI DESCRIPTION BUFFER 1 REGISTER

The format for this register is the same as the format o the QMSPI Description Buffer 0 Register.

6.11.13 QMSPI DESCRIPTION BUFFER 2 REGISTER

The format for this register is the same as the format o the QMSPI Description Buffer 0 Register.

TABLE 10-2: INTERRUPT SOURCE, ENABLE SET, ENABLE CLEAR, AND RESULT BIT ASSIGNMENTS (CONTINUED)

Aggregator IRQ	Aggregator Bit	HWB Instance Name	Interrupt Event	Wake Event	Source Description
GIRQ10	21-22	Test	Test	-	-
GIRQ10	23	GPIO067	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	1	GPIO001	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	2	GPIO002	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	3	GPIO003	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	4	GPIO004	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	5	GPIO005	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	6	GPIO006	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	7	GPIO007	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	8	GPIO010	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	9	GPIO011	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	10	GPIO012	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	11	GPIO013	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	12	GPIO014	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	13	GPIO015	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	14	GPIO016	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	15	GPIO017	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	16	GPIO020	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	17	GPIO021	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	18	GPIO022	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	19	GPIO023	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	20	GPIO024	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	21	GPIO025	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	22	GPIO026	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	23	GPIO027	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	24	GPIO030	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	25	GPIO031	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	26	GPIO032	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	27	GPIO033	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	28	GPIO034	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	29	GPIO035	GPIO Event	Yes	GPIO Interrupt Event
GIRQ11	30	GPIO036	GPIO Event	Yes	GPIO Interrupt Event
GIRQ12	0	SMBus Controller 0	SMB	No	SMBus Controller 0 Interrupt Event
GIRQ12	1	SMBus Controller 1	SMB	No	SMBus Controller 1 Interrupt Event
GIRQ12	2	SMBus Controller 2	SMB	No	SMBus Controller 2 Interrupt Event
GIRQ13	0	DMA Controller	DMA0	No	DMA Controller - Channel 0 Interrupt Event
GIRQ13	1	DMA Controller	DMA1	No	DMA Controller - Channel 1 Interrupt Event

16.17.1 PORT 92 REGISTER

Offset	0h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:2	Reserved	R	-	-
1	ALT_GATE_A20 This bit provides an alternate means for system control of the GATEA20 pin. ALT_A20 low drives GATEA20 low, if A20 from the keyboard controller is also low. When Port 92 is enabled, writing a 1 to this bit forces ALT_A20 high. ALT_A20 high drives GATEA20 high regardless of the state of A20 from the keyboard controller. 0=ALT_A20 is driven low 1=ALT_A20 is driven high	R/W	0h	nSIO_ RESET
0	ALT_CPU_RESET This bit provides an alternate means to generate a CPU_RESET pulse. The CPU_RESET output provides a means to reset the system CPU to effect a mode switch from Protected Virtual Address Mode to the Real Address Mode. This provides a faster means of reset than is provided through the EC keyboard controller. Writing a “1” to this bit will cause the ALT_RST# internal signal to pulse (active low) for a minimum of 6µs after a delay of 14µs. Before another ALT_RST# pulse can be generated, this bit must be written back to “0”.	R/W	0h	nSIO_ RESET

16.18 Emulated 8042 Interface EC-Only Registers

The registers listed in the EC-Only Register Summary table are for a single instance of the Legacy Port92/GATEA20 logic. The addresses of each register listed in this table are defined as a relative offset to the host “Base Address” defined in the EC-Only Register Base Address Table.

TABLE 16-15: EC-ONLY REGISTER BASE ADDRESS

Block Instance	Instance Number	Host	Address Space	Base Address
Port92-Legacy	0	EC	32-bit address space	000F_1900h

The Base Address indicates where the first register can be accessed in a particular address space for a block instance.

TABLE 16-16: EC-ONLY REGISTER SUMMARY

Offset	Register Name (Mnemonic)
0h	GATEA20 Control Register
8h	SETGA20L Register
Ch	RSTGA20L Register

17.11.4 PROGRAMMABLE BAUD RATE GENERATOR MSB REGISTER

Offset	01h (DLAB=1)			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7	BAUD_CLK_SEL 0=If CLK_SRC is '0', the baud clock is derived from the 1.8432MHz_Clk. If CLK_SRC is '1', this bit has no effect 1=If CLK_SRC is '0', the baud clock is derived from the 24MHz_Clk. If CLK_SRC is '1', this bit has no effect	R/W	0h	RESET
6:0	BAUD_RATE_DIVISOR_MSB See Section 17.9.1, "Programmable Baud Rate".	R/W	0h	RESET

17.11.5 INTERRUPT ENABLE REGISTER

The lower four bits of this register control the enables of the five interrupt sources of the Serial Port interrupt. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of this register. Similarly, setting the appropriate bits of this register to a high, selected interrupts can be enabled. Disabling the interrupt system inhibits the Interrupt Identification Register and disables any Serial Port interrupt out of the MEC140x/1x. All other system functions operate in their normal manner, including the Line Status and MODEM Status Registers. The contents of the Interrupt Enable Register are described below.

Offset	01h (DLAB=0)			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7:4	Reserved	R	-	-
3	EMSI This bit enables the MODEM Status Interrupt when set to logic "1". This is caused when one of the Modem Status Register bits changes state.	R/W	0h	RESET
2	ELSI This bit enables the Received Line Status Interrupt when set to logic "1". The error sources causing the interrupt are Overrun, Parity, Framing and Break. The Line Status Register must be read to determine the source.	R/W	0h	RESET
1	ETHREI This bit enables the Transmitter Holding Register Empty Interrupt when set to logic "1".	R/W	0h	RESET
0	ERDAI This bit enables the Received Data Available Interrupt (and timeout interrupts in the FIFO mode) when set to logic "1".	R/W	0h	RESET

The Base Address indicates where the first register can be accessed in a particular address space for a block instance.

TABLE 18-2: RUNTIME REGISTER SUMMARY

Offset	Register Name
00h	Timer Count Register
04h	Timer Preload Register
08h	Timer Status Register
0Ch	Timer Int Enable Register
10h	Timer Control Register

18.9.1 TIMER COUNT REGISTER

Offset	00h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:0	COUNTER This is the value of the Timer counter. This is updated by Hardware but may be set by Firmware. If it is set while the Hardware Timer is operating, functionality can not be ensured. When read, it is buffered so single byte reads will be able to catch the full 4 byte register without it changing. The size of the Counter is indicated by the instance name. Bits 0 to (size-1) are r/w counter bits. Bits 31 down to size are reserved. Reads return 0 and writes have no effect.	R/W	0h	Timer_Reset

18.9.2 TIMER PRELOAD REGISTER

Offset	04h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:0	PRE_LOAD This is the value of the Timer pre-load for the counter. This is used by H/W when the counter is to be restarted automatically; this will become the new value of the counter upon restart. The size of the Pre-Load value is the same as the size of the counter. The size of the Counter is indicated by the instance name. Bits 0 to (size-1) are r/w pre-load bits. Bits 31 down to size are reserved. Reads return 0 and writes have no effect.	R/W	0h	Timer_Reset

MEC140x				
GPIO Name (Octal)	**Mux Control = 00**	**Mux Control = 01**	**Mux Control = 10**	**Mux Control = 11**
GPIO112	GPIO112	PS2_CLK1A	KSO13	Reserved
GPIO113	GPIO113	PS2_DAT1A	KSO14	Reserved
GPIO114	GPIO114	PS2_CLK0	Reserved	Reserved
GPIO115	GPIO115	PS2_DAT0	Reserved	Reserved
GPIO116	GPIO116	TFDP_DATA	UART_RX	Reserved
GPIO117	GPIO117	TFDP_CLK	UART_TX	Reserved
GPIO120	GPIO120	CMP_VOUT1	Reserved	Reserved
GPIO121	GPIO121	ADC0	Reserved	Reserved
GPIO122	GPIO122	ADC1	Reserved	Reserved
GPIO123	GPIO123	SHD_CS#	Reserved	Reserved
GPIO124	GPIO124	CMP_VOUT0	Reserved	Reserved
GPIO125	GPIO125	KSO15	Reserved	Reserved
GPIO126	GPIO126	SHD_SCLK	Reserved	Reserved
GPIO127	GPIO127	PS2_DAT1B	Reserved	Reserved
GPIO130	GPIO130	SMB03_DATA	SMB03_DATA18	Reserved
GPIO131	GPIO131	SMB03_CLK	SMB03_CLK18	Reserved
GPIO132	GPIO132	KSO16	Reserved	Reserved
GPIO133	GPIO133	SHD_IO0	Reserved	Reserved
GPIO134	GPIO134	SHD_IO1	Reserved	Reserved
GPIO135	GPIO135	SHD_IO2	Reserved	Reserved
GPIO136	GPIO136	SHD_IO3	Reserved	Reserved
GPIO140	GPIO140	KSO17	Reserved	Reserved
GPIO141	GPIO141	SMB04_DATA	SMB04_DATA18	Reserved
GPIO142	GPIO142	SMB04_CLK	SMB04_CLK18	Reserved
GPIO143	GPIO143	KSI0	DTR#	Reserved
GPIO144	GPIO144	KSI1	DCD#	Reserved
GPIO145	GPIO145	Reserved	Reserved	Reserved
GPIO146	GPIO146	Reserved	Reserved	Reserved
GPIO147	GPIO147	KSI4	DSR#	Reserved
GPIO150	GPIO150	KSI5	RI#	Reserved
GPIO151	GPIO151	KSI6	RTS#	Reserved
GPIO152	GPIO152	KSI7	CTS#	Reserved
GPIO153	GPIO153	ADC4	Reserved	Reserved
GPIO154	GPIO154	ADC3	Reserved	Reserved
GPIO155	GPIO155	ADC2	Reserved	Reserved
GPIO156	GPIO156	LED1	Reserved	Reserved
GPIO157	GPIO157	LED0	TST_CLK_OUT	Reserved
GPIO160	GPIO160	DAC_0	Reserved	Reserved
GPIO161	GPIO161	DAC_1	Reserved	Reserved
GPIO162	GPIO162	VCI_IN1#	Reserved	Reserved
GPIO163	GPIO163	VCI_IN0#	Reserved	Reserved
GPIO164	GPIO164	VCI_OVRD_IN	Reserved	Reserved
GPIO165	GPIO165	CMP_VREF0	Reserved	Reserved
GPIO166	GPIO166	CMP_VREF1	UART_CLK	Reserved

22.6.1 PIN CONTROL REGISTERS

Two Pin Control Registers are implemented for each GPIO. The Pin Control Register format is described in Section 22.6.1.1, "Pin Control Register," on page 329. The Pin Control Register 2 format is described in Section 22.6.1.2, "Pin Control Register 2," on page 332. Pin Control Register address offsets and defaults for each product are defined in Section 22.5.4.1, "MEC140x Pin Control Registers Defaults," on page 320, and Section 22.5.4.2, "MEC141x Pin Control Registers Defaults," on page 323.

22.6.1.1 Pin Control Register

Offset	See Table 22-2, "Register Summary"			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:25	RESERVED	RES	-	-
24	GPIO input from pad On reads, Bit [24] reflects the state of GPIO input from the pad regardless of setting of Bit [10]. **Note:** This bit is forced high when the selected power well is off as selected by the Power Gating Signal bits. See bits[3:2].	R	Note 22-6	nSYSRST
23:17	RESERVED	RES	-	-
16	GPIO output data If enabled by the GPIO Output Control Select bit, the GPIO output data bit determines the level on the GPIO pin when the pin is configured for the GPIO output function. On writes: If enabled via the GPIO Output Control Select 0: GPIO[x] out = '0' 1: GPIO[x] out = '1' **Note:** If disabled via the GPIO Output Control Select then the GPIO[x] out pin is unaffected by writing this bit. On reads: Bit [16] returns the last programmed value, not the value on the pin.	R/W (GPIO Output Control Select = 0) R (GPIO Output Control Select=1)	Note 22-6	nSYSRST
15:14	RESERVED	RES	-	-
13:12	Mux Control The Mux Control field determines the active signal function for a pin. 00 = GPIO Function Selected 01 = Signal Function 1 Selected 10 = Signal Function 2 Selected 11 = Signal Function 3 Selected	R/W	Note 22-6	nSYSRST
11	Polarity 0 = Non-inverted 1 = Inverted When the Polarity bit is set to '1' and the Mux Control bits are greater than '00,' the selected signal function outputs are inverted and Interrupt Detection (int_det) sense defined in Table 22-3, "Edge Enable and Interrupt Detection Bits Definition" is inverted. When the Mux Control field selects the GPIO signal function (Mux = '00'), the Polarity bit does not effect the output. Regardless of the state of the Mux Control field and the Polarity bit, the state of the pin is always reported without inversion in the GPIO input register.	R/W	Note 22-6	nSYSRST

24.0 INTERNAL DMA CONTROLLER

24.1 Features

- Supports Memory-to-Memory BYTE, WORD, and DWORD transfers
- Used to Perform DMA transactions for DMA capable hardware IP blocks
- Supports 7 DMA Channels that may be configured for any Hardware Device or Memory transfer
- Channel 0 Supports CRC-32 generation

24.2 Introduction

The Internal DMA Controller transfers data to/from the source from/to the destination. The firmware is responsible for setting up each channel. Afterwards either the firmware or the hardware may perform the flow control. The hardware flow control exists entirely inside the source device. Each transfer may be 1, 2, or 4 bytes in size, so long as the device supports a transfer of that size. Every device must be on the internal 32-bit address space.

24.3 References

No references have been cited for this chapter

24.4 Terminology

TABLE 24-1: TERMINOLOGY

Term	Definition
DMA Transfer	This is a complete **DMA Transfer** which is done after the **Master Device** terminates the transfer, the Firmware Aborts the transfer or the DMA reaches its transfer limit. A DMA Transfer may consist of one or more data packets.
Data Packet	Each data packet may be composed of 1, 2, or 4 bytes. The size of the data packet is limited by the max size supported by both the source and the destination. Both source and destination will transfer the same number of bytes per packet.
Channel	The Channel is responsible for end-to-end (source-to-destination) Data Packet delivery.
Device	A Device may refer to a Master or Slave connected to the DMA Channel. Each DMA Channel may be assigned one or more devices.
Master Device	This is the master of the DMA, which determines when it is active. The Firmware is the master while operating in Firmware Flow Control. The Hardware is the master while operating in Hardware Flow Control. The Master Device in Hardware Mode is selected by **DMA Channel Control:Hardware Flow Control Device**. It is the index of the **Flow Control** Port.
Slave Device	The Slave Device is defined as the device associated with the targeted Memory Address.

29.5 Signal Description

TABLE 29-1: SIGNAL DESCRIPTION TABLE

Name	Direction	Description
PS2_DAT	INPUT/ OUTPUT	Data from the PS/2 device
PS2_CLK	INPUT/ OUTPUT	Clock from the PS/2 device

29.6 Host Interface

The registers defined for the Keyboard Scan Interface are accessible by the various hosts as indicated in Section 29.15, "EC-Only Registers".

29.7 Power, Clocks and Reset

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

29.7.1 POWER DOMAINS

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

29.7.2 CLOCK INPUTS

Name	Description
48 MHz Ring Oscillator	This is the clock source for PS/2 Interface logic.
2 MHz Clock	The PS/2 state machine is clocked using the 2 MHz clock.

29.7.3 RESETS

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

30.11.3 KSI STATUS REGISTER

Offset	0Ch			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:8	Reserved	R	-	-
7:0	KSI_STATUS Each bit in this field is set on the falling edge of the corresponding KSI input pin. A KSI interrupt is generated when its corresponding status bit and interrupt enable bit are both set. KSI interrupts are logically ORed together to produce KSC_INT and KSC_INT_WAKE. Writing a '1' to a bit will clear it. Writing a '0' to a bit has no effect.	R/WC	0h	nSYSRST

30.11.4 KSI INTERRUPT ENABLE REGISTER

Offset	10h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:8	Reserved	R	-	-
7:0	KSI_INT_EN Each bit in KSI_INT_EN enables interrupt generation due to high-to-low transition on a KSI input. An interrupt is generated when the corresponding bits in KSI_STATUS and KSI_INT_EN are both set.	R/W	0h	nSYSRST

30.11.5 KEYSCAN EXTENDED CONTROL REGISTER

Offset	14h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
32:1	Reserved	R	-	-
0	PREDRIVE_ENABLE PREDRIVE_ENABLE enables the PREDRIVE mode to actively drive the KSO pins high for approximately 100 ns before switching to open-drain operation. 0=Disable predrive on KSO pins 1=Enable predrive on KSO pins.	RW	0	nSYSRST

Note: Steps 3 thorough 7 should be completed as a contiguous sequence. If not the interface could be presenting incorrect data when software thinks it is accessing a valid register read.

31.10.2 BC-LINK MASTER WRITE OPERATION

1. Software starts by checking the status of the BUSY bit in the BC-Link Status Register. If the BUSY bit is '0', proceed. If BUSY is '1', firmware must wait until it is '0'.
2. Software writes the address of the register to be written into the BC-Link Address Register.
3. Software writes the data to be written into the addressed register in to the BC-Link Data Register.
4. The write to the Data Register starts the BC_Link write operation. The Master state machine sets the BUSY bit.
5. The BC-Link Master Interface transmits the write request packet.
6. When the write request packet is received by the BC-Link companion, the CRC is checked and data is written to the addressed companion register.
7. The companion sends an ACK if the write is completed. A time-out will occur approximately 16 BC-Link clocks after the packet is sent by the Master state machine. If a time-out occurs, the state machine will set the BC_ERR bit in the Status Register to '1' approximately 48 clocks later and then clear the BUSY bit.
8. The Master state machine issues the Bit Clear interrupt and clears the BUSY bit after receiving the ACK from the Companion
9. If a Bus Error occurs, firmware must issue a soft reset by setting the RESET bit in the Status Register to '1'.
10. The write can re-tried once BUSY is cleared.\

31.11 EC-Only Registers

The registers listed in the EC-Only Register Summary table are for a single instance of the BC-Link Master interface. The addresses of each register listed in this table are defined as a relative offset to the host "Base Address" defined in the EC-Only Register Base Address Table.

TABLE 31-3: EC-ONLY REGISTER BASE ADDRESS TABLE

Block Instance	Instance Number	Host	Address Space	Base Address ()
BC-LINK	0	EC	32-bit internal address space	0000_BC00h
BC-LINK	1	EC	32-bit internal address space	0000_BD00h

The Base Address indicates where the first register can be accessed in a particular address space for a block instance.

TABLE 31-4: EC-ONLY REGISTER SUMMARY

Register Name	EC Offset
BC-Link Status Register	00h
BC-Link Address Register	04h
BC-Link Data Register	08h
BC-Link Clock Select Register	0Ch

35.7 EC-Only Registers

TABLE 35-1: EC-ONLY REGISTER BASE ADDRESS

Block Instance	Instance Number	Host	Address Space	Base Address
VBAT_REG_BANK	0	EC	32-bit internal address space	0000_A400h

Note 35-1 The Base Address indicates where the first register can be accessed in a particular address space for a block instance.

TABLE 35-2: RUNTIME REGISTER SUMMARY

Offset	Register Name
00h	Power-Fail and Reset Status Register
04h	Test Register
08h	Clock Enable Register
10h	Test Register
14h	Test Register
18h	Alternate Function VTR Control
1Ch	Test Register

35.7.1 POWER-FAIL AND RESET STATUS REGISTER

The Power-Fail and Reset Status Register collects and retains the VBAT RST and WDT event status when VTR is unpowered.

Address	00h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
7	VBAT_RST The VBAT RST bit is set to ‘1’ by hardware when a VBAT_POR is detected. This is the register default value. To clear VBAT RST EC firmware must write a ‘1’ to this bit; writing a ‘0’ to VBAT RST has no affect.	R/WC	1	VBAT_POR
6	Reserved	RES	-	-
5	WDT The WDT bit is asserted (‘1’) following a Watch-Dog Timer Forced Reset (WDT Event). To clear the WDT bit EC firmware must write a ‘1’ to this bit; writing a ‘0’ to the WDT bit has no affect.	R/WC	0 (Note 35-2)	VBAT_POR (Note 35-2)

TABLE 38-3: ANALOG TO DIGITAL CONVERTER REGISTER SUMMARY

Offset	Register Name (Mnemonic)
00h	ADC Control Register
04h	ADC Delay Register
08h	ADC Status Register
0Ch	ADC Single Register
10h	ADC Repeat Register
14h	ADC Channel 0 Reading Register
18h	ADC Channel 1 Reading Register
1Ch	ADC Channel 2 Reading Register
20h	ADC Channel 3 Reading Register
24h	ADC Channel 4 Reading Register
28h	ADC Channel 5 Reading Register
2Ch	ADC Channel 6 Reading Register
30h	ADC Channel 7 Reading Register
34h	ADC Channel 8 Reading Register

Note: The unused channel reading registers are reserved. See Products on page 3 for the specific number of channels supported for a particular device.

38.11.1 ADC CONTROL REGISTER

The ADC Control Register is used to control the behavior of the Analog to Digital Converter.

Offset	00h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:8	RESERVED	RES		
7	**Single_Done_Status** This bit is cleared when it is written with a 1. Writing a 0 to this bit has no effect. This bit can be used to generate an EC interrupt. 0: ADC single-sample conversion is not complete. This bit is cleared whenever an ADC conversion cycle begins for a single conversion cycle. 1: ADC single-sample conversion is completed. This bit is set to 1 when all enabled channels in the single conversion cycle.	R/WC	0h	nSYSRST

38.11.2 ADC DELAY REGISTER

The ADC Delay register determines the delay from setting Start_Repeat in the ADC Control Register and the start of a conversion cycle. This register also controls the interval between conversion cycles in repeat mode.

Offset	04h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:16	**Repeat_Delay[15:0]** This field determines the interval between conversion cycles when Start_Repeat is 1. The delay is in units of 40µs. A value of 0 means no delay between conversion cycles, and a value of 0xFFFF means a delay of 2.6 seconds. This field has no effect when Start_Single is written with a 1.	R/W	0000h	nSYSRST
15:0	**Start_Delay[15:0]** This field determines the starting delay before a conversion cycle is begun when Start_Repeat is written with a 1. The delay is in units of 40µs. A value of 0 means no delay before the start of a conversion cycle, and a value of 0xFFFF means a delay of 2.6 seconds. This field has no effect when Start_Single is written with a 1.	R/W	0000h	nSYSRST

38.11.3 ADC STATUS REGISTER

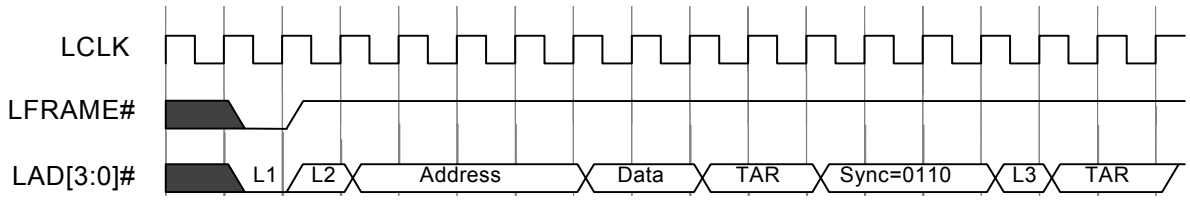
The ADC Status Register indicates whether the ADC has completed a conversion cycle.

Offset	08h			
Bits	**Description**	**Type**	**Default**	**Reset Event**
31:8	RESERVED	RES		
7:0	**ADC_Ch_Status[7:0]** **All bits are cleared by being written with a '1'.** 0: conversion of the corresponding ADC channel is not complete 1: conversion of the corresponding ADC channel is complete Note: 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. See Note 38-2.	R/WC	00h	nSYSRST

Note 38-2 Bits that correspond to the unused channels are reserved. See Products on page 3 for the specific number of channels supported for a particular device.

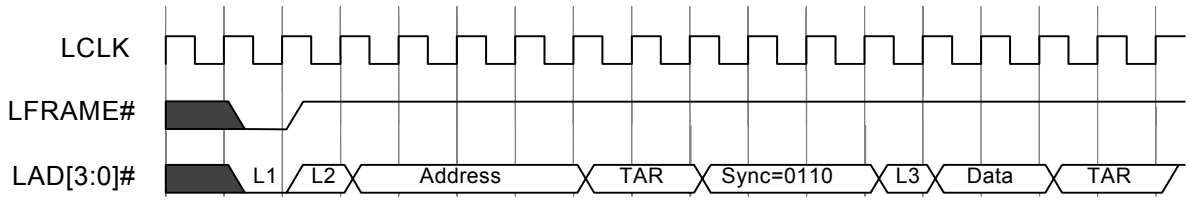
43.17.5 LPC I/O TIMING

FIGURE 43-18: I/O WRITE



Note: L1=Start; L2=CYCTYP+DIR; L3=Sync of 0000

FIGURE 43-19: I/O READ



Note: L1=Start; L2=CYCTYP+DIR; L3=Sync of 0000

43.17.6 SERIAL IRQ TIMING

FIGURE 43-20: SETUP AND HOLD TIME

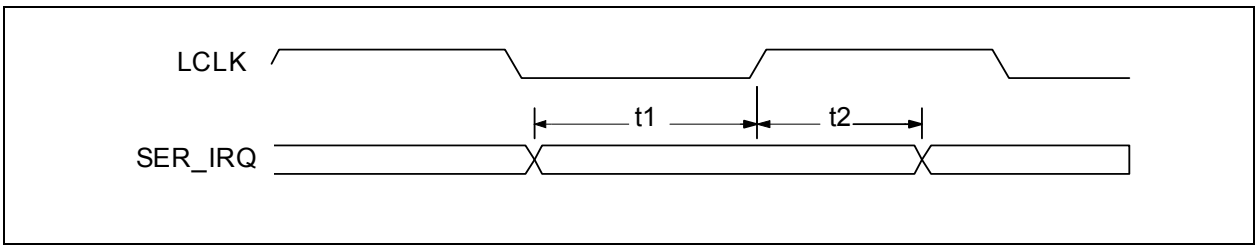


TABLE 43-21: SETUP AND HOLD TIME

Name	Description	MIN	TYP	MAX	Units
t1	SER_IRQ Setup Time to LCLK Rising	7			nsec
t2	SER_IRQ Hold Time to LCLK Rising	0			

43.17.7 nEC_SCI TIMING

nEC_SCI pin has the same minimum timing requirements as GPIO signals. See Section 43.5, "GPIO Timings," on page 507.

TABLE 44-1: REGISTER MEMORY MAP (CONTINUED)

Addr. (Hex)	HW Block Instance Name	HW Block Instance No.	Reg. Bank Name	Reg. Instance Name	Size (Bytes)
815F0	GPIO	0	GPIO Registers	GPIO104 Pin Control 2	4
815F4	GPIO	0	GPIO Registers	GPIO105 Pin Control 2	4
815F8	GPIO	0	GPIO Registers	GPIO106 Pin Control 2	4
815FC	GPIO	0	GPIO Registers	GPIO107 Pin Control 2	4
81600	GPIO	0	GPIO Registers	GPIO110 Pin Control 2	4
81604	GPIO	0	GPIO Registers	GPIO111 Pin Control 2	4
81608	GPIO	0	GPIO Registers	GPIO112 Pin Control 2	4
8160C	GPIO	0	GPIO Registers	GPIO113 Pin Control 2	4
81610	GPIO	0	GPIO Registers	GPIO114 Pin Control 2	4
81614	GPIO	0	GPIO Registers	GPIO115 Pin Control 2	4
81618	GPIO	0	GPIO Registers	GPIO116 Pin Control 2	4
8161C	GPIO	0	GPIO Registers	GPIO117 Pin Control 2	4
81620	GPIO	0	GPIO Registers	GPIO120 Pin Control 2	4
81624	GPIO	0	GPIO Registers	GPIO121 Pin Control 2	4
81628	GPIO	0	GPIO Registers	GPIO122 Pin Control 2	4
8162C	GPIO	0	GPIO Registers	GPIO123 Pin Control 2	4
81630	GPIO	0	GPIO Registers	GPIO124 Pin Control 2	4
81634	GPIO	0	GPIO Registers	GPIO125 Pin Control 2	4
81638	GPIO	0	GPIO Registers	GPIO126 Pin Control 2	4
8163C	GPIO	0	GPIO Registers	GPIO127 Pin Control 2	4
81640	GPIO	0	GPIO Registers	GPIO130 Pin Control 2	4
81644	GPIO	0	GPIO Registers	GPIO131 Pin Control 2	4
81648	GPIO	0	GPIO Registers	GPIO132 Pin Control 2	4
8164C	GPIO	0	GPIO Registers	GPIO133 Pin Control 2	4
81650	GPIO	0	GPIO Registers	GPIO134 Pin Control 2	4
81654	GPIO	0	GPIO Registers	GPIO135 Pin Control 2	4
81658	GPIO	0	GPIO Registers	GPIO136 Pin Control 2	4
81660	GPIO	0	GPIO Registers	GPIO140 Pin Control 2	4
81664	GPIO	0	GPIO Registers	GPIO141 Pin Control 2	4
81668	GPIO	0	GPIO Registers	GPIO142 Pin Control 2	4
8166C	GPIO	0	GPIO Registers	GPIO143 Pin Control 2	4
81670	GPIO	0	GPIO Registers	GPIO144 Pin Control 2	4
81674	GPIO	0	GPIO Registers	GPIO145 Pin Control 2	4
81678	GPIO	0	GPIO Registers	GPIO146 Pin Control 2	4
8167C	GPIO	0	GPIO Registers	GPIO147 Pin Control 2	4
81680	GPIO	0	GPIO Registers	GPIO150 Pin Control 2	4
81684	GPIO	0	GPIO Registers	GPIO151 Pin Control 2	4
81688	GPIO	0	GPIO Registers	GPIO152 Pin Control 2	4
8168C	GPIO	0	GPIO Registers	GPIO153 Pin Control 2	4

TABLE 44-1: REGISTER MEMORY MAP (CONTINUED)

Addr. (Hex)	HW Block Instance Name	HW Block Instance No.	Reg. Bank Name	Reg. Instance Name	Size (Bytes)
F2902	ACPI EC Interface	2	ACPI_EC_Only	EC2OS Data EC-Register Byte 2	1
F2903	ACPI EC Interface	2	ACPI_EC_Only	EC2OS Data EC-Register Byte 3	1
F2904	ACPI EC Interface	2	ACPI_EC_Only	STATUS EC-Register	1
F2905	ACPI EC Interface	2	ACPI_EC_Only	Byte Control EC-Register	1
F2908	ACPI EC Interface	2	ACPI_EC_Only	OS2EC Data EC-Register Byte 0 - CMD	1
F2908	ACPI EC Interface	2	ACPI_EC_Only	OS2EC Data EC-Register Byte 0 - DATA	1
F2909	ACPI EC Interface	2	ACPI_EC_Only	OS2EC Data EC-Register Byte 1	1
F290A	ACPI EC Interface	2	ACPI_EC_Only	OS2EC Data EC-Register Byte 2	1
F290B	ACPI EC Interface	2	ACPI_EC_Only	OS2EC Data EC-Register Byte 3	1
F2C00	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 0 - Read	1
F2C00	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 0 - Write	1
F2C01	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 1 - Read	1
F2C01	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 1 - Write	1
F2C02	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 2 - Read	1
F2C02	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 2 - Write	1
F2C03	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 3 - Read	1
F2C03	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Data Register Byte 3 - Write	1
F2C04	ACPI EC Interface	3	ACPI_Runtime	ACPI OS Command Register	1
F2C04	ACPI EC Interface	3	ACPI_Runtime	STATUS OS-Register	1
F2C05	ACPI EC Interface	3	ACPI_Runtime	Byte Control OS-Register	1
F2D00	ACPI EC Interface	3	ACPI_EC_Only	EC2OS Data EC-Register Byte 0	1
F2D01	ACPI EC Interface	3	ACPI_EC_Only	EC2OS Data EC-Register Byte 1	1
F2D02	ACPI EC Interface	3	ACPI_EC_Only	EC2OS Data EC-Register Byte 2	1
F2D03	ACPI EC Interface	3	ACPI_EC_Only	EC2OS Data EC-Register Byte 3	1
F2D04	ACPI EC Interface	3	ACPI_EC_Only	STATUS EC-Register	1