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	Obsolete
Applications	Touchscreen Controller
Core Processor	M8C
Program Memory Type	FLASH (32kB)
Controller Series	CY8CT
RAM Size	2K x 8
Interface	I²C, SPI, UART/USART, USB
Number of I/O	28
Voltage - Supply	1.8V
Operating Temperature	-40°C ~ 85°C
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-QFN
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy8ctmg200a-32lqxi

Contents

Section A: Overview 13

1. Pin Information 19

1.1 Pinouts 19

1.1.1 CY8CTMG200-16LGXI, CY8CTMG200A-16LGXI, CY8CTST200-16LGXI, CY8CTST200A-16LGXI PSoC 16-Pin Part Pinout 19

1.1.2 CY8CTMG200-24LQXI, CY8CTMG200A-24LQXI, CY8CTST200-24LQXI, CY8CTST200A-24LQXI PSoC 24-Pin Part Pinout 20

1.1.3 CY8CTMG200-32LQXI, CY8CTMG200A-32LQXI, CY8CTST200-32LQXI, CY8CTST200A-32LQXI, CY8CTMG201-32LQXI, CY8CTMG201A-32LQXI PSoC 32-Pin Part Pinout21

1.1.4 CY8CTMG200-48LTXI, CY8CTMG200A-48LTXI, CY8CTST200-48LTXI, CY8CTST200A-48LTXI, CY8CTMG201-48LTXI, CY8CTMG201A-48LTXI PSoC 48-Pin Part Pinout22

Section B: PSoC Core 23

2. CPU Core (M8C) 27

2.1 Overview 27

2.2 Internal Registers 27

2.3 Address Spaces 27

2.4 Instruction Set Summary 28

2.5 Instruction Formats 30

2.5.1 One-Byte Instructions 30

2.5.2 Two-Byte Instructions 30

2.5.3 Three-Byte Instructions 31

2.6 Register Definitions 32

2.6.1 CPU_F Register 32

2.6.2 Related Registers 32

3. Supervisory ROM (SROM) 33

3.1 Architectural Description 33

3.1.1 Additional SROM Feature 34

3.1.2 SROM Function Descriptions 34

3.1.2.1 SWBootReset Function 34

3.1.2.2 ReadBlock Function 35

3.1.2.3 WriteBlock Function 35

3.1.2.4 EraseBlock Function 36

3.1.2.5 ProtectBlock Function 36

3.1.2.6 TableRead Function 36

3.1.2.7 EraseAll Function 36

3.1.2.8 Checksum Function 37

3.1.2.9 Calibrate0 Function 37

3.1.2.10 Calibrate1 Function 37

Getting Started

The quickest path to understanding PSoC is by reading the PSoC device's data sheet and using *PSoC Designer™ Integrated Development Environment (IDE)*. This manual is useful for understanding the details of the PSoC integrated circuit.

Important Note For the most up-to-date Ordering, Packaging, or Electrical Specification information, refer to the individual PSoC device's data sheet or go to http://www.cypress.com.

Support

Free support for PSoC products is available online at http://www.cypress.com. Resources include Training Seminars, Discussion Forums, Application Notes, TightLink Technical Support Email/Knowledge Base, and Application Support Technicians.

Technical Support can be reached at http://www.cypress.com/support.

Product Upgrades

Cypress provides scheduled upgrades and version enhancements for PSoC Designer free of charge. You can order the upgrades from your distributor on CD-ROM or download them directly from http://www.cypress.com under Software. Also provided are critical updates to system documentation under http://www.cypress.com >> Documentation.

Development Kits

The Cypress Online Store contains development kits, ***C*** compilers, and all accessories for PSoC development. Go to the Cypress Online Store web site at http://www.cypress.com under Order >> PSoC Kits.

Document History

This section serves as a chronicle of the *PSoC® CY8CTMG20x, CY8CTMG20xA, CY8CTST200, CY8CTST200A Technical Reference Manual*.

Technical Reference Manual History

Version/ Release Date	Originator	Description of Change
** May 2009	DSG	First release of the CY8CTMG20x, CY8CTST200 Technical Reference Manual.
*A August 2009	DSG	Second release of the CY8CTMG20x, CY8CTST200 Technical Reference Manual.
*B November 2009	FSU	Multiple fixes, primarily to the sleep and I2C chapters.
*C December 2009	FSU	Multiple fixes, primarily to the External Crystal Oscillator chapter.

2.4 Instruction Set Summary

The instruction set is summarized in both Table 2-1 and Table 2-2 (in numeric and ***mnemonic*** order, respectively), and serves as a quick reference. If more information is needed, the Instruction Set Summary tables are described in detail in the *PSoC Designer Assembly Language User Guide* (refer to the http://www.cypress.com web site).

Table 2-1. Instruction Set Summary Sorted Numerically by Opcode

Opcode Hex	Cycles	Bytes	Instruction Format	Flags	Opcode Hex	Cycles	Bytes	Instruction Format	Flags	Opcode Hex	Cycles	Bytes	Instruction Format	Flags
00	15	1	SSC		2D	8	2	OR [X+expr], A	Z	5A	5	2	MOV [expr], X	
01	4	2	ADD A, expr	C, Z	2E	9	3	OR [expr], expr	Z	5B	4	1	MOV A, X	Z
02	6	2	ADD A, [expr]	C, Z	2F	10	3	OR [X+expr], expr	Z	5C	4	1	MOV X, A	
03	7	2	ADD A, [X+expr]	C, Z	30	9	1	HALT		5D	6	2	MOV A, reg[expr]	Z
04	7	2	ADD [expr], A	C, Z	31	4	2	XOR A, expr	Z	5E	7	2	MOV A, reg[X+expr]	Z
05	8	2	ADD [X+expr], A	C, Z	32	6	2	XOR A, [expr]	Z	5F	10	3	MOV [expr], [expr]	
06	9	3	ADD [expr], expr	C, Z	33	7	2	XOR A, [X+expr]	Z	60	5	2	MOV reg[expr], A	
07	10	3	ADD [X+expr], expr	C, Z	34	7	2	XOR [expr], A	Z	61	6	2	MOV reg[X+expr], A	
08	4	1	PUSH A		35	8	2	XOR [X+expr], A	Z	62	8	3	MOV reg[expr], expr	
09	4	2	ADC A, expr	C, Z	36	9	3	XOR [expr], expr	Z	63	9	3	MOV reg[X+expr], expr	
0A	6	2	ADC A, [expr]	C, Z	37	10	3	XOR [X+expr], expr	Z	64	4	1	ASL A	C, Z
0B	7	2	ADC A, [X+expr]	C, Z	38	5	2	ADD SP, expr		65	7	2	ASL [expr]	C, Z
0C	7	2	ADC [expr], A	C, Z	39	5	2	CMP A, expr	if (A=B) Z=1 if (A<B) C=1	66	8	2	ASL [X+expr]	C, Z
0D	8	2	ADC [X+expr], A	C, Z	3A	7	2	CMP A, [expr]		67	4	1	ASR A	C, Z
0E	9	3	ADC [expr], expr	C, Z	3B	8	2	CMP A, [X+expr]		68	7	2	ASR [expr]	C, Z
0F	10	3	ADC [X+expr], expr	C, Z	3C	8	3	CMP [expr], expr		69	8	2	ASR [X+expr]	C, Z
10	4	1	PUSH X		3D	9	3	CMP [X+expr], expr		6A	4	1	RLC A	C, Z
11	4	2	SUB A, expr	C, Z	3E	10	2	MVI A, [[expr]++]	Z	6B	7	2	RLC [expr]	C, Z
12	6	2	SUB A, [expr]	C, Z	3F	10	2	MVI [[expr]++], A		6C	8	2	RLC [X+expr]	C, Z
13	7	2	SUB A, [X+expr]	C, Z	40	4	1	NOP		6D	4	1	RRC A	C, Z
14	7	2	SUB [expr], A	C, Z	41	9	3	AND reg[expr], expr	Z	6E	7	2	RRC [expr]	C, Z
15	8	2	SUB [X+expr], A	C, Z	42	10	3	AND reg[X+expr], expr	Z	6F	8	2	RRC [X+expr]	C, Z
16	9	3	SUB [expr], expr	C, Z	43	9	3	OR reg[expr], expr	Z	70	4	2	AND F, expr	C, Z
17	10	3	SUB [X+expr], expr	C, Z	44	10	3	OR reg[X+expr], expr	Z	71	4	2	OR F, expr	C, Z
18	5	1	POP A	Z	45	9	3	XOR reg[expr], expr	Z	72	4	2	XOR F, expr	C, Z
19	4	2	SBB A, expr	C, Z	46	10	3	XOR reg[X+expr], expr	Z	73	4	1	CPL A	Z
1A	6	2	SBB A, [expr]	C, Z	47	8	3	TST [expr], expr	Z	74	4	1	INC A	C, Z
1B	7	2	SBB A, [X+expr]	C, Z	48	9	3	TST [X+expr], expr	Z	75	4	1	INC X	C, Z
1C	7	2	SBB [expr], A	C, Z	49	9	3	TST reg[expr], expr	Z	76	7	2	INC [expr]	C, Z
1D	8	2	SBB [X+expr], A	C, Z	4A	10	3	TST reg[X+expr], expr	Z	77	8	2	INC [X+expr]	C, Z
1E	9	3	SBB [expr], expr	C, Z	4B	5	1	SWAP A, X	Z	78	4	1	DEC A	C, Z
1F	10	3	SBB [X+expr], expr	C, Z	4C	7	2	SWAP A, [expr]	Z	79	4	1	DEC X	C, Z
20	5	1	POP X		4D	7	2	SWAP X, [expr]		7A	7	2	DEC [expr]	C, Z
21	4	2	AND A, expr	Z	4E	5	1	SWAP A, SP	Z	7B	8	2	DEC [X+expr]	C, Z
22	6	2	AND A, [expr]	Z	4F	4	1	MOV X, SP		7C	13	3	LCALL	
23	7	2	AND A, [X+expr]	Z	50	4	2	MOV A, expr	Z	7D	7	3	LJMP	
24	7	2	AND [expr], A	Z	51	5	2	MOV A, [expr]	Z	7E	10	1	RETI	C, Z
25	8	2	AND [X+expr], A	Z	52	6	2	MOV A, [X+expr]	Z	7F	8	1	RET	
26	9	3	AND [expr], expr	Z	53	5	2	MOV [expr], A		8x	5	2	JMP	
27	10	3	AND [X+expr], expr	Z	54	6	2	MOV [X+expr], A		9x	11	2	CALL	
28	11	1	ROMX	Z	55	8	3	MOV [expr], expr		Ax	5	2	JZ	
29	4	2	OR A, expr	Z	56	9	3	MOV [X+expr], expr		Bx	5	2	JNZ	
2A	6	2	OR A, [expr]	Z	57	4	2	MOV X, expr		Cx	5	2	JC	
2B	7	2	OR A, [X+expr]	Z	58	6	2	MOV X, [expr]		Dx	5	2	JNC	
2C	7	2	OR [expr], A	Z	59	7	2	MOV X, [X+expr]		Ex	7	2	JACC	
										Fx	13	2	INDEX	Z

Note 1 Interrupt acknowledge to Interrupt Vector table = 13 cycles.

Note 2 The number of cycles required by an instruction is increased by one for instructions that span 128 byte page boundaries in the Flash memory space.

4.2.3 STK_PP Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
0,D1h	STK_PP						Page Bits[2:0]			RW : 0

The Stack Page Pointer Register (STK_PP) is used to set the effective SRAM page for stack memory accesses in a multi-SRAM page PSoC device.

Bits 2 to 0: Page Bits[2:0]. These bits have the potential to affect two types of memory access.

The purpose of this register is to determine on which SRAM page to store the stack. In the reset state, this register's value is 0x00 and the stack is in SRAM Page 0. However, if the STK_PP register value is changed, the next stack operation occurs on the SRAM page indicated by the new STK_PP value. Therefore, set the value of this register early in the program and never change it. If the program changes the STK_PP value after the stack grows, the program must ensure that the STK_PP value is restored when needed.

Note The impact that the STK_PP register has on the stack is independent of the SRAM Paging bits in the CPU_F register.

The second type of memory accesses that the STK_PP register affects are indexed memory accesses when the CPU_F[7:6] bits are set to 11b. In this mode, Source Indexed and Destination Indexed memory accesses are directed to the stack SRAM page, rather than the SRAM page indicated by the IDX_PP register or SRAM Page 0.

For additional information, refer to the STK_PP register on page 235.

4.2.4 IDX_PP Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
0,D3h	IDX_PP						Page Bits[2:0]			RW : 0

The Index Page Pointer Register (IDX_PP) sets the effective SRAM page for indexed memory accesses in a multi-SRAM page PSoC device.

Bits 2 to 0: Page Bits[2:0]. These bits allow instructions, which use the Source Indexed and Destination Indexed address modes, to operate on an SRAM page that is not equal to the current SRAM page. However, the effect this register has on indexed addressing modes is only enabled when the CPU_F[7:6] is set to 10b.

When CPU_F[7:6] is set to 10b and an indexed memory access is made, the access is directed to the SRAM page indicated by the value of the IDX_PP register.

See the STK_PP register description for more information on other indexed memory access modes. For additional

4.2.5 MVR_PP Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
0,D4h	MVR_PP						Page Bits[2:0]			RW : 0

The MVI Read Page Pointer Register (MVR_PP) sets the effective SRAM page for MVI read memory accesses in a multi-SRAM page PSoC device.

Bits 2 to 0: Page Bits[2:0]. These bits are only used by the `MVI A, [expr]` instruction, not to be confused with the `MVI [expr], A` instruction covered by the MVW_PP register. This instruction is considered a read because data is transferred from SRAM to the microprocessor's A register (CPU_A).

When an `MVI A, [expr]` instruction is executed in a device with more than one page of SRAM, the SRAM address that is read by the instruction is determined by the value of the least significant bits in this register. However, the pointer for the `MVI A, [expr]` instruction is always located in the current SRAM page. See the *PSoC Designer Assembly Language User Guide* for more information on the `MVI A, [expr]` instruction.

The function of this register and the `MVI` instructions are independent of the SRAM Paging bits in the CPU_F register. For additional information, refer to the MVR_PP register on page 237.

6.2.4 IO_CFG1 Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
1,DCh	IO_CFG1	StrongP		Range[1:0]		P1_LOW_THRS	SPICLK_ON_P10	REG_EN	IOINT	RW : 00

The Input/Output Configuration Register 1 (IO_CFG1) configures the Port 1 output regulator and set the Interrupt mode for all GPIO.

Bit 7: StrongP. Setting this bit increases the drive strength and edge ratio for high outputs.

Bit 5 and 4: Range[1:0]. These bits select the regulator output level for Port 1. Available levels are 3.0V, 1.8V, and 2.5V.

Selects the high output level for Port 1 outputs.

Range[1:0]	Output Level
00	3.0 volts
01	3.0 volts
10	1.8 volts
11	2.5 volts

Bit 3 P1_LOW_THRS. This bit reduces the threshold voltage of the P1 port input buffers so that there are no compatibility issues when Port 1 is communicating at regulated voltage levels.

'0' is standard threshold of VIH, VIL. '1' is reduce threshold of VIH, VIL.

Bit 2: SPICLK_ON_P10. When this bit is set to '1', the SPI clock is mapped to Port 1 pin 0. Otherwise, it is mapped to Port 1 pin 3.

Bit 1: REG_EN. The Register Enable bit (REG_EN) controls the regulator on Port 1 outputs.

Bit 0: IO INT. This bit sets the GPIO Interrupt mode for all pins in the CY8CTMG20x, CY8CTST200 PSoC devices. GPIO interrupts are controlled at each pin by the PRTxIE registers, and also by the global GPIO bit in the INT_MSK0 register.

For additional information, refer to the IO_CFG1 register on page 272.

6.2.5 IO_CFG2 Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
1,DEh	IO_CFG2			REG_LEVEL[2:0]				REG_CLOCK[1:0]		RW : 00

The Input/Output Configuration Register 2 (IO_CFG2) selects output regulated supply and clock rates.

Bits 5 to 3: REG_LEVEL[2:0]. These bits select output regulated supply.

REG_LEVEL[2:0]	Approx. Regulated Supply (V)		
000	3	2.5	1.8
001	3.1	2.6	1.9
010	3.2	2.7	2.0
011	3.3	2.8	2.1
100	3.4	2.9	2.2
101	3.5	3.0	2.3
110	3.6	3.1	2.4
111	3.7	3.2	2.5

Bits 1 to 0: REG_CLOCK[1:0]. The Regulated I/O charge pump can operate with a maximum clock speed of 12 MHZ. The REG_CLOCK[1:0] bits select clocking options for the regulator. Setting REG_CLOCK[1:0] to '10' should be used with 24 MHz SYSCLK and '01' should be used with 6/12 MHz SYSCLK.

REG_CLOCK[1:0]	SYSCLK Clock Rate
10	24 MHz
01	6/12 MHz

For additional information, refer to the IO_CFG2 register on page 275.

7.3 Register Definitions

The following registers are associated with the Internal Main Oscillator (IMO). The register descriptions have associated register tables showing the bit structure for that register. The bits in the tables that are grayed out are reserved bits and are not detailed in the register descriptions that follow. Always write reserved bits with a value of '0'. For a complete table showing all oscillator registers, refer to the Summary Table of the Core Registers on page 24.

7.3.1 IMO_TR Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
1,E8h	IMO_TR	Trim[7:0]								RW : 00

The Internal Main Oscillator Trim Register (IMO_TR) manually centers the oscillator's output to a target frequency.

This register is loaded with a factory trim value at boot. When changing frequency ranges, the matching frequency trim value must be loaded into this register.

A TableRead command to the Supervisory ROM returns the trim values to the SRAM. EraseAll Parameters (05h), on page 36 has information on the location of various trim settings stored in Flash tables. Firmware needs to read the right trim value for desired frequency and update the IMO_TR register. The IMO_TR register must be changed at the lower frequency range setting.

For additional information, refer to the IMO_TR register on page 281

7.3.2 IMO_TR1 Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
1,FAh	IMO_TR1						Fine Trim[2:0]			RW : 0

The Internal Main Oscillator Trim Register 1 (IMO_TR1) adjusts the IMO frequency .

Bits 2 to 0: Fine Trim[2:0]. These bits provide a fine tuning capability to the IMO trim. These three bits are the 3 LSB of the IMO trim with the IMO_TR register supplying the 8 MSB. A larger value in this register will increase the speed of the oscillator. The value in these bits varies the IMO frequency: approximately 7.5 kHz/step. When the EnableLock bit is set in the USB_CR1 register, firmware writes to this register are disabled.

For additional information, refer to the IMO_TR1 register on page 286.

9. External Crystal Oscillator (ECO)

This chapter briefly explains the External Crystal Oscillator (ECO) and its associated registers. The 32.768 kHz external crystal oscillator circuit allows the user to replace the internal low speed oscillator with a more precise time source. For a quick reference of all PSoC registers in address order, refer to the Register Reference chapter on page 187.

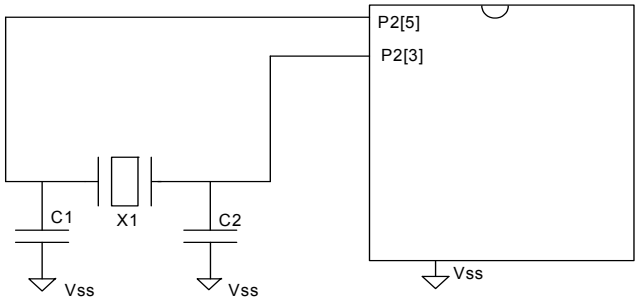
9.1 Architectural Description

The External Crystal Oscillator (ECO) circuit requires only the following external components: an inexpensive watch crystal and two small value capacitors. The XTALIn (P2[3]) and XTALOut (P2[5]) pins connect to a 32.768 kHz watch crystal and the two external capacitors bypass these pins to ground. Figure 9-1 shows the external connections needed to implement the ECO. See the Application Overview on page 70 for information on enabling the ECO. Transitions between the internal and external oscillator domains may produce glitches on the clock bus.

During the process of activating the ECO, there must be a hold-off period before using it as the 32.768 kHz source. This hold-off period is partially implemented in hardware using the sleep timer. Firmware must set up a sleep period of one second (maximum ECO settling time), and then enable the ECO in the OSC_CR0 register. At the one second timeout (the sleep interrupt), the switch is made by hardware to the ECO. If the ECO is subsequently deactivated, the Internal Low Speed Oscillator (ILO) will again be activated and the switch is made back to the ILO immediately.

The ECO Exists bit (ECO EX, bit 0 of ECO_CONFIG) is used to control whether the switch-over is allowed or locked. This is a write once bit. It is written early in code execution after a Power on Reset (POR) or external reset (XRES) event. A '1' in this bit indicates to the hardware that a crystal exists in the system, and firmware is allowed to switch back and forth between ECO and ILO operation. If the bit is '0', switch-over to the ECO is locked out.

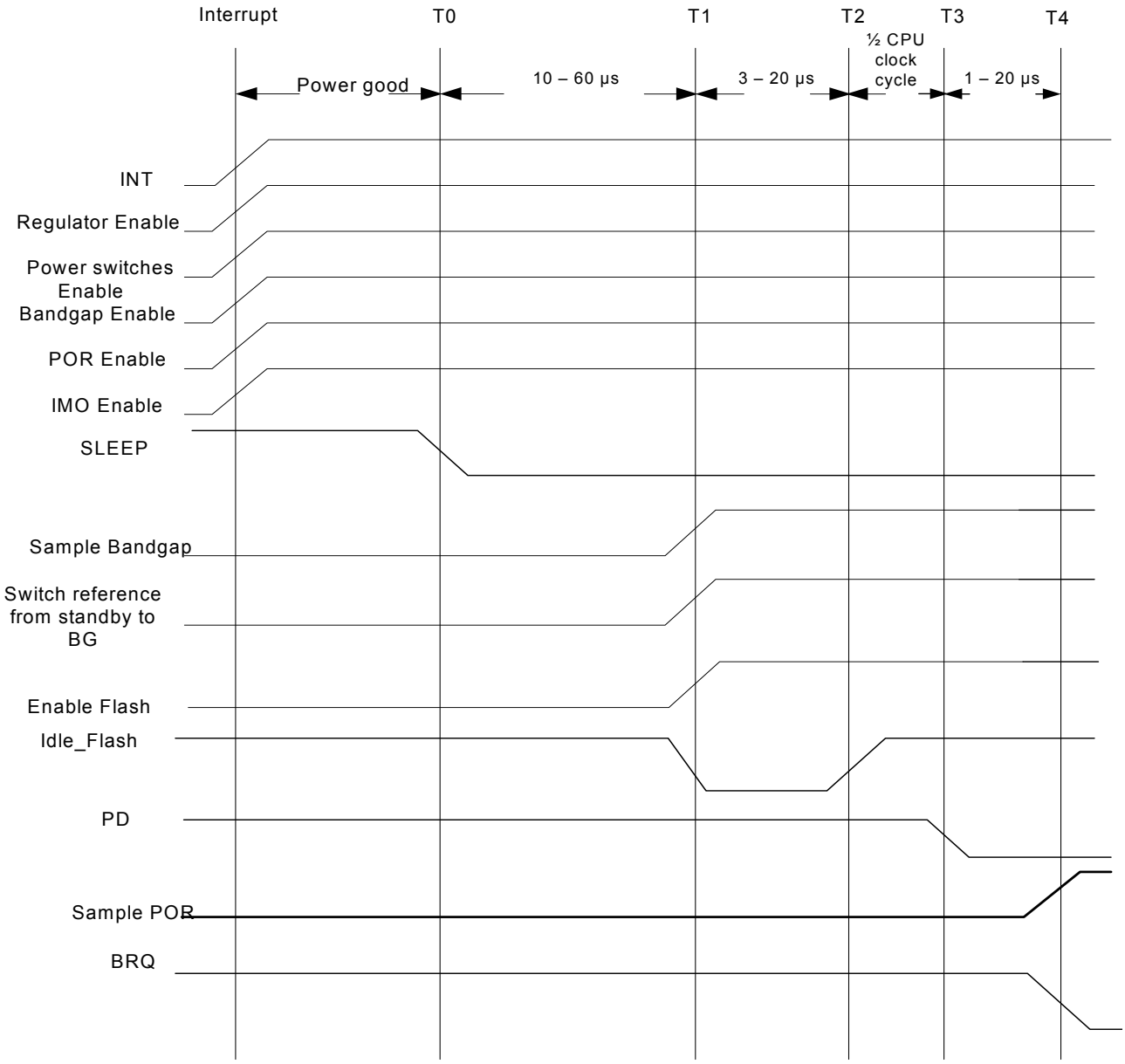
Figure 9-1. External Components for the ECO



The ECO Exists Written bit (ECO EXW, bit 1 of ECO_CONFIG) is read only and is set on the first write to this register. When this bit is '1', it indicates that the state of ECO EX is locked. This is illustrated in Figure 9-2.

As shown in Figure 10-2, once the SLEEP bit is deasserted, the wakeup is initiated. The sequence is shown in the following timing diagram. The taps used in this wakeup sequence are generated based upon user configuration settings in the SLP_CFG3 register."

Figure 10-2. Wakeup Sequence for the Device[123]



1. The duration of Power Good is 3 ILO Cycles.
2. The timing of T0 – T4 is based on the IMO frequency and the settings in the SLP_CFG3 register. For additional information, refer to the SLP_CFG3 Register on page 78.
3. The maximum worst-case duration of the wakeup sequence is 263 µs, based on the minimum specified ILO frequency of 19 kHz, the minimum specified IMO frequency, and the default settings of the SLP_CFG3 register.

14.2 Register Definitions

The following registers are associated with the Digital Clocks and are listed in address order. Each register description has an associated register table showing the bit structure for that register. The bits in the tables that are grayed out throughout this manual are reserved bits and are not detailed in the register descriptions that follow. Always write reserved bits with a value of '0'. For a complete table of digital clock registers, refer to the "System Resources Register Summary" on page 106.

14.2.1 USB_MISC_CR Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
1,BDh	USB_MISC_CR						USB_SE_EN	USB_ON	USB_CLK_ON	RW : 0

The USB Miscellaneous Control Register controls the clocks to the USB block, to make the IMO work with better accuracy for the USB part and to disable the single ended input of the USBIO in the case of a non-USB part.

Bit 2: USB_SE_EN. The single ended outputs of USBIO is enabled or disabled based upon this bit setting. Set this bit to '1' when using this part as a USB part for USB transactions to occur. Set this bit to '0' to disable single ended outputs of USBIO. The DPO, DMO are held at logic high state and RSE0 is held at a low state.

Note Bit [1:0] of the USBIO_CR1 register is also affected depending on this register setting. When this bit is '0' (default), regardless of the DP and DM state, the DPO and DMO bits of USBIO_CR1 are '11b'.

Bit 1: USB_ON. This bit is used by the IMO DAC block to either work with better DNL consuming higher power, or with sacrificed DNL consuming lower power. Set this bit to '1' when the part is used as a USB part. A '0' runs the IMO with sacrificed DNL by consuming less power. A '1' runs the IMO with better DNL by consuming more power.

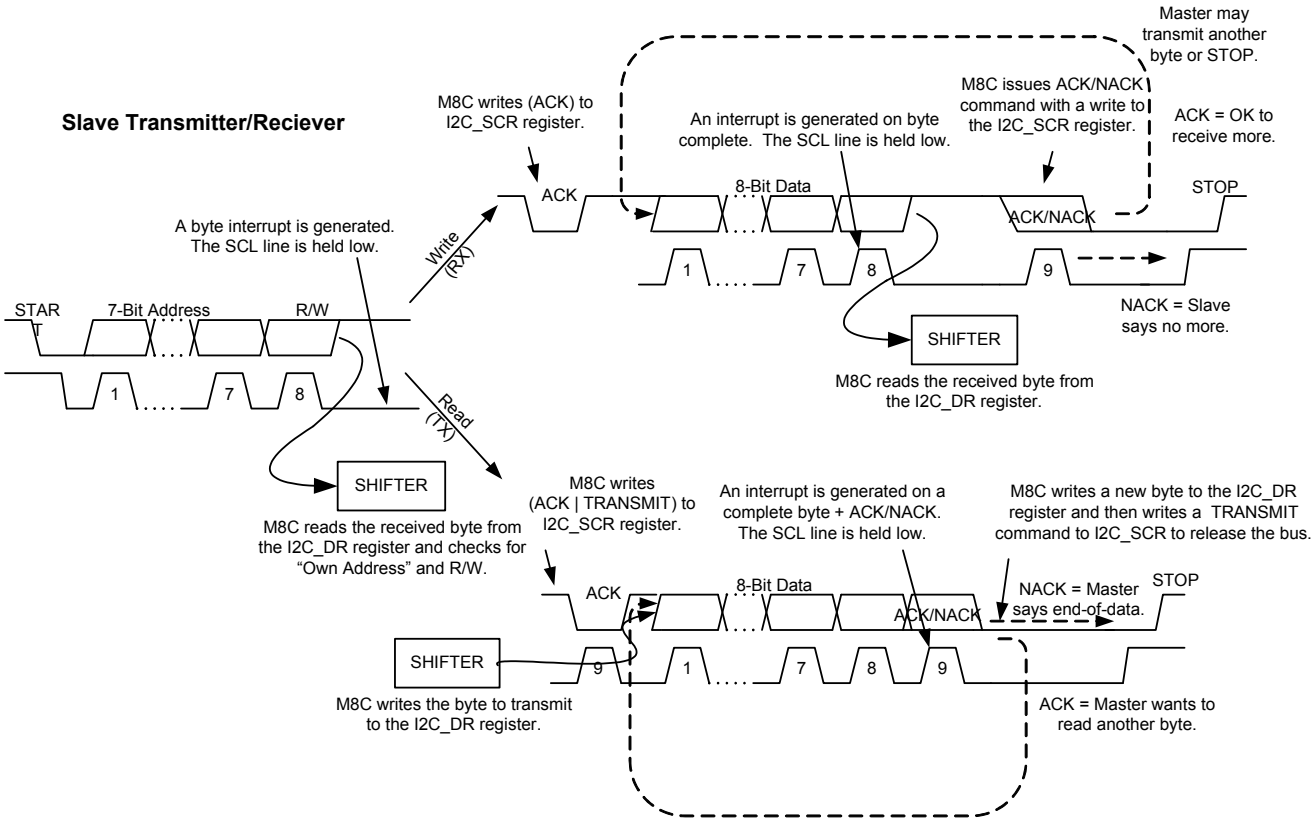
Bit 0: USB_CLK_ON. This bit either enables or disables the clocks to the USB block. It is used to save power in cases when the device need not respond to USB traffic. Set this bit to '1' when the device is used as a USB part.

When this bit is a '0', all clocks to the USB block are driven. The device does not respond to USB traffic and none of the USB registers, except IMO_TR, IMO_TR1 and USBIO_CR1, listed in the Register Definitions on page 171 are writable.

When this bit is a '1', clocks are not blocked to the USB block. The device responds to USB traffic depending on the other register settings mentioned under Register Definitions in the Full-Speed USB chapter on page 165.

For additional information, refer to the USB_MISC_CR register on page 267.

Figure 15-3. Slave Operation



15.2.2 EZI2C Mode

When EZI2C mode is configured with the Buffer Mode bit of the I2C_XCFG register, both the master and slave must use a predefined communication protocol. In this protocol, the I^2C slave operates as a RAM buffer interface, in which the external master controls the read and write addresses to the 32-byte RAM buffer. The external master reads and writes to the RAM buffer are independent of the CPU reads and writes to the RAM buffer, so conceptually, it appears as a dual port RAM buffer. Higher level protocols, such as polling or semaphore methods, are used to ensure data synchronization and integrity. Conceptually, data is always available in this mode so the bus never stalls except in I^2C sleep mode.

The RAM buffer interfaces contain separate address pointers, I2C_BP and I2C_CP, that are set with the first data byte of a write operation. When the external master writes one or more bytes, the first data byte is always the base address pointer value. This value gets written to both the base address pointer, I2C_BP, and the current address pointer, I2C_CP. The byte after the base address pointer is written into the location pointed to by the current address pointer value contained in I2C_CP.

The third byte (second data byte) is written to the current address pointer value, I2C_CP, plus one and so on. This current address pointer, I2C_CP, increments for each byte read or written, but is reset to the base address pointer value at the beginning of each new write or read operation (following a start command).

For example, if the base address pointer, I2C_BP, is set to 4, a read operation begins to read data at location 4 and continues sequentially until the host completes the read operation. So, if the base address pointer is set to 4, each read operation resets the data pointer to 4 and reads sequentially from that location. This is true whether single or multiple read operations are performed. The base address pointer is not changed until a new write operation initiates.

If the I^2C master attempts to write data past the RAM address boundary, 32 bytes, the data is discarded and does not affect any RAM inside or outside the designated RAM area. You cannot read data outside the allowed range. Any read requests by the master outside the allowed range result in the return of invalid data. If the RAM address is sent

15.3.2 I2C_XSTAT Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
0,C9h	I2C_XSTAT							Dir	Slave Busy	R : 0

The I²C Extended Status Register (I2C_XSTAT) reads enhanced feature status. All bits are read only. When the bits of I2C_XCFG are left in their reset state, the block is in compatibility mode, and this register is not in use.

Bit 1: Dir. This bit indicates the direction of the current transfer. A '1' indicates a master read, and a '0' indicates a master write. It is only valid when the Slave Busy bit (bit 0) is set to '1'.

Bits 0: Slave Busy. This bit is set upon a hardware address compare and is reset upon the following stop signal. Poll this bit to determine when the slave is busy and the buffer module is being accessed.

For additional information, refer to the I2C_XSTAT register on page 227.

15.3.3 I2C_ADDR Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
0,CAh	I2C_ADDR		Slave Address[6:0]							RW : 00

The I²C Slave Address Register (I2C_ADDR) holds the slave's 7-bit address. All bits are RW.

Note When hardware address compare mode is not enabled in the I2C_XCFG register, this register is not in use.

Bits 6 to 0: Slave Address[6:0]. These 7 bits hold the slave's own device address.

For additional information, refer to the I2C_ADDR register on page 228.

15.3.4 I2C_BP Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
0,CBh	I2C_BP				I2C Base Pointer[4:0]					R : 00

The I²C Base Address Pointer Register (I2C_BP) contains the base address value of the RAM data buffer.

Note When in compatibility mode, this register is not in use.

Bits 4 to 0: I2C Base Pointer[4:0]. In the EZI2C protocol, the first data byte after the slave address transaction in write mode is the base address for subsequent reads and writes and it is transferred directly into this register. If the desired transaction is a master write to the slave, subsequent bytes are written to the RAM buffer starting with this address and auto incremented (see I2C_CP Register).

In case of a read, a Start or Restart must be issued and the read location starts with this address and again subsequent read addresses are auto incremented as pointed to by the I2C_CP register value. The value of this register is modified only at the beginning of every I²C write transaction. The I²C master must always supply a value for this register in the first byte of data after the slave's address in a given write transaction. If performing reads, the master need not set the value of this register. The current value of this register is also used directly for reads.

For additional information, refer to the I2C_BP register on page 229.

17. POR and LVD

This chapter briefly discusses the Power on Reset (POR) and Low Voltage Detect (LVD) circuits and their associated registers. For a complete table of the POR registers, refer to the Summary Table of the System Resource Registers on page 106. For a quick reference of all PSoC registers in address order, refer to the Register Reference chapter on page 187.

17.1 Architectural Description

The Power on Reset (POR) and Low Voltage Detect (LVD) circuits provide protection against low voltage conditions. The POR function senses Vcc and Vcore (regulated voltage) holding the system in reset until the magnitude of Vcc and Vcore supports operation to specification. The LVD function senses Vcc and provides an interrupt to the system when Vcc falls below a selected threshold. Other outputs and status bits are provided to indicate important voltage trip levels. Refer to Section 16.2 Pin Behavior During Reset for a description of GPIO pin behavior during power up.

20. Full-Speed USB

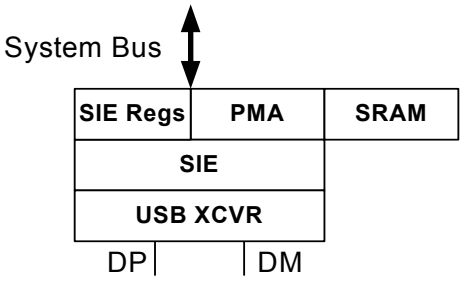
This chapter explains the Full-Speed USB (Universal Serial Bus) resource and its associated registers. For a quick reference of all PSoC registers in address order, refer to the Register Reference chapter on page 187.

20.1 Architectural Description

The PSoC USB system resource adheres to the USB 2.0 Specification for full-speed devices operating at 12 Mbps with one upstream port and one USB address. PSoC USB consists of these components:

- Serial Interface Engine (SIE) block
- PSoC Memory Arbiter (PMA) block
- 512 bytes of dedicated SRAM
- A Full-Speed USB Transceiver with internal regulator and two dedicated USB pins

Figure 20-1. USB Block Diagram



At the PSoC system level, the full-speed USB system resource interfaces to the rest of the PSoC by way of the M8C's register access instructions and to the outside world by way of the two USB pins. The SIE supports nine endpoints including a bidirectional control endpoint (endpoint 0) and eight uni-directional data endpoints (endpoints 1 to 8). The uni-directional data endpoints are individually configurable as either IN or OUT.

20.2 Application Description

The individual components and issues of the USB system are described in detail in the following sections.

20.2.1 USB SIE

The USB Serial Interface Engine (SIE) allows the PSoC device to communicate with the USB host at full-speed data rates (12 Mbps). The SIE simplifies the interface to USB traffic by automatically handling the following USB processing tasks without firmware intervention:

- Translates the encoded received data and formats the data to be transmitted on the bus.
- Generates and checks CRCs. Incoming packets failing checksum verification are ignored.
- Checks addresses. Ignores all transactions not addressed to the device.
- Sends appropriate ACK/NAK/Stall handshakes.
- Identifies token type (SETUP, IN, OUT) and sets the appropriate token bit once a valid token in received.
- Identifies Start-of-Frame (SOF) and saves the frame count.
- Sends data to or retrieves data from the USB SRAM, by way of the PSoC Memory Arbiter (PMA).

Firmware is required to handle various parts of the USB interface. The SIE issues interrupts after key USB events to direct firmware to appropriate tasks:

- Fill and empty the USB data buffers in USB SRAM.
- Enable PMA channels appropriately.
- Coordinate enumeration by decoding USB device requests.
- Suspend and resume coordination.
- Verify and select data toggle values.

20.3.10 EPx_CR0 Register

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Access
1,54h	EP1_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,55h	EP2_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,56h	EP3_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,57h	EP4_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,58h	EP5_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,59h	EP6_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,5Ah	EP7_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00
1,5Bh	EP8_CR0	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]				# : 00

The Endpoint Control Register 0 (EPx_CR0) is used for status and configuration of the non-control endpoints 1 to 8.

Bit 7: Stall. When this bit is set, the SIE stalls an OUT packet if the mode bits are set to ACK-OUT. The SIE stalls an IN packet if the mode bits are set to ACK-IN. This bit must be cleared for all other modes. '0' is do not issue a stall. '1' is stall an OUT packet if mode bits are set to ACK-OUT, or stall an IN packet if mode bits are set to ACK-IN.

Bit 5: NAK Int Enable. When set, this bit causes an endpoint interrupt to be generated even when a transfer completes with a NAK. '0' is do not issue an interrupt after completing the transaction by sending NAK. '1' is interrupt after transaction is complete by sending NAK.

Bit 4: ACK'ed Transaction. The ACK'ed Transaction bit is set whenever the SIE engages in a transaction to the register's endpoint that completes with an ACK packet. This bit is cleared by any writes to the register. '0' is no ACK'ed transactions since bit was last cleared. '1' indicates a transaction ended with an ACK.

Bits 3 to 0: Mode[3:0]. The mode controls how the USB SIE responds to traffic and how the USB SIE changes the mode of that endpoint as a result of host packets to the endpoint. Refer to "Mode Encoding for Control and Non-Control Endpoints" on page 166.

For additional information, refer to the EPx_CR0 register on page 265.

Section E: Registers

The Registers section discusses the registers of the PSoC device. It lists all the registers in mapping tables, in address order. For easy reference, each register is linked to the page of a detailed description located in the next chapter. This section encompasses the following chapter:

- Register Reference chapter on page 187.

Register General Conventions

The register conventions specific to this section and the Register Reference chapter are listed in the following table:

Register Conventions

Convention	Description
Empty, grayed-out table cell	Illustrates a reserved bit or group of bits.
‘x’ before the comma in an address	Indicates the register exists in register bank 1 and register bank 2.
‘x’ in a register name	Indicates that there are multiple instances/address ranges of the same register.
R	Read register or bit(s).
W	Write register or bit(s).
O	Only a read/write register or bit(s).
L	Logical register or bit(s).
C	Clearable register or bit(s).
#	Access is bit specific.

Register Mapping Tables

The PSoC device has a total register address space of 512 bytes. The register space is also referred to as I/O space and is broken into two parts: Bank 0 (user space) and Bank 1 (configuration space). The XIO bit in the Flag register (CPU_F) determines which bank the user is currently in. When the XIO bit is set, the user is said to be in the “extended” address space or the “configuration” registers.

Refer to the individual PSoC device data sheets for device-specific register mapping information.

21.3.31 CS_SLEW

TrueTouch Slew Control Register

Individual Register Names and Addresses:

CS_SLEW : 0,A8h

	7	6	5	4	3	2	1	0
Access : POR	RW : 00							RW : 0
Bit Name	FastSlew[6:0]							FS_EN

This register enables and controls a fast slewing mode for the relaxation oscillator.

For additional information, refer to the Register Definitions on page 92 in the TrueTouch Module chapter.

Bit	Name	Description
7:1	**FastSlew[6:0]**	This 7-bit value sets a counter, clocked at the IMO frequency. While the counter is counting down from this value, the relaxation oscillator edge slews at the maximum gain setting. During this interval, the IRANGE bits in the CS_CR2 register are internally set to maximum (11b). At the end of the interval, the user-defined IRANGE level is restored so that the relaxation oscillator continues slewing with a slower edge rate to the target voltage threshold. If the FS_EN bit is low, the FastSlew setting has no effect. After each edge of the relaxation oscillator, the counter is re-loaded and the fast slewing interval re-occurs, followed by the slower edge rate at the end of the count down. Note that the IRANGE bits in the CS_CR2 register always read the user-defined setting. Because the IRANGE value is forced to maximum during this interval, the increase in the edge rate can be 1X, 2X, 4X, or 8X, depending on the programmed value of the IRANGE bits. 0000000b No fast edge rate interval. 0000001b Minimum fast edge rate interval (1 IMO period). … 1111111b Maximum fast edge rate interval (127 IMO period).
0	**FS_EN**	Enable bit for the Fast Slew mode. 0 Fast slew mode disabled. 1 Fast slew mode enabled. After each relaxation oscillator transition, the relaxation oscillator runs with a higher current for a time controlled by the FastSlew bits.

21.3.51 I2C_CFG

I²C Configuration Register

Individual Register Names and Addresses:

I2C_CFG : 0,D6h

	7	6	5	4	3	2	1	0
Access : POR		RW : 0		RW : 0	RW : 0			RW : 0
Bit Name		PSelect		Stop IE	Clock Rate[1:0]			Enable

This register is used to set the basic operating modes, baud rate, and interrupt selection.

In the table above, note that reserved bits are grayed table cells and are not described in the bit description section below. Always write reserved bits with a value of '0'. For additional information, refer to the Register Definitions on page 122 in the I2C Slave chapter.

Bit	Name	Description
6	**P Select**	I2C Pin Select. 0 P1[5] and P1[7]. 1 P1[0] and P1[1]. **Note** Read the I2C Slave chapter for a discussion of the side effects of choosing the P1[0] and P1[1] pair of pins.
4	**Stop IE**	Stop Interrupt Enable. 0 Disabled. 1 Enabled. An interrupt is generated on the detection of a Stop condition.
3:2	**Clock Rate[1:0]**	00b 100K Standard Mode. 01b 400K Fast Mode. 10b 50K Standard Mode. 11b Reserved..
0	**Enable**	0 Disabled. 1 Enabled.

21.4.7 EPx_CR0

Endpoint Control Registers 0

Individual Register Names and Addresses:

EP1_CR0 : 1,54h
EP2_CR0 : 1,55h
EP3_CR0 : 1,56h
EP4_CR0 : 1,57h
EP5_CR0 : 1,58h
EP6_CR0 : 1,59h
EP7_CR0 : 1,5Ah
EP7_CR0 : 1,5Bh

	7	6	5	4	3	2	1	0
Access : POR	RW : 0		RW : 0	RC : 0	RW : 0			
Bit Name	Stall		NAK_INT_EN	ACK'ed Tx	Mode[3:0]			

These registers endpoint control registers.

In the table above, note that the reserved bit is a grayed table cell and is not described in the bit description section below. Reserved bits must always be written with a value of '0'. For additional information, refer to the Register Definitions on page 171 in the Full-Speed USB chapter.

Bit	Name	Description
7	**Stall**	When this bit is set, the SIE stalls an OUT packet if the Mode bits are set to ACK-OUT. The SIE stalls an IN packet if the Mode bits are set to ACK-IN. This bit must be clear for all other modes.
5	**NAK_INT_EN**	When set, this bit causes an endpoint interrupt to be generated even when a transfer completes with a NAK.
4	**ACKed Tx**	The ACK'ed transaction bit is set whenever the SIE engages in a transaction to the register's end-point that completes with an ACK packet.
3:0	**Mode[3:0]**	The mode controls how the USB SIE responds to traffic and how the USB SIE changes the mode of that endpoint as a result of host packets to the endpoint.

in MVR_PP register 237
in MVW_PP register 238
in STK_PP register 235
Pending Interrupt bits 252
PgMode bits 254
pin behavior during reset 135
pin information, *See* pinouts
pinouts
16-pin part 19
24-pin part 20
32-pin part 21
48-pin part 22
PMAx_CUR register 180
POR and LVD 143
architecture 143
register definitions 144
PORLEV bits 279
PORS bit 258
power modes
system resets 141
power on reset (POR)
See POR and LVD
power on reset in system resets 139
PPM bit 217
PPS bit 217
product upgrades 16
programmable timer 161
architecture 161
register definitions 163
ProtectBlock function in SROM 36
protocol function for SPI 145
PRTxDM0 register 60, 259
PRTxDM1 register 60, 260
PRTxDR register 59, 188
PRTxIE register 59, 189
PSelect bit 239
PSoC core
architecture 23
overview 14
register summary 24
See also CPU core
PSSDC bits 283, 284, 285
PT_CFG register 163, 164, 221
PT_DATA1 register 164
PTx_DATA0 register 164, 223
PTx_DATA1 register 164, 222
PXD_EN bit 213

R

RAM paging 39
architecture 39
basic paging 39
current page pointer 40
index memory page pointer 41
interrupts 40
MVI instructions 40
register definitions 42
stack operations 40
ReadBlock function in SROM 35
reference of all registers 187
register conventions 17, 187
register definitions
comparators 103
CPU core 32
digital clocks 112
general purpose IO 59
I2C slave 122
internal low speed oscillator 68
internal main oscillator 64
interrupt controller 48
IO analog multiplexer 100
POR and LVD 144
programmable timer 163
RAM paging 42
sleep and watchdog 77
SPI 147
supervisory ROM 38
system resets 137
TrueTouch module 92
USB, full-speed 171
register mapping tables
bank 0 registers 184
bank 1 registers 185
registers
bank 0 registers 188
bank 1 registers 259
core register summary 24
internal M8C registers 27
maneuvering around 187
mapping tables 183
reference of all registers 187
system resources register summary 106
TrueTouch register summary 84
regulated IO
application overview 70
architecture 69
RES_WDT register 77, 253
RLOCLK bit 212
RX Reg Full bit 192

S

serial peripheral interconnect, *See* SPI
Slave bit 261
slave function for SPI 146
slave operation, I2C 118
sleep and watchdog 73
application overview 76
architecture 73
bandgap refresh 80
register definitions 77
sleep sequence 79
sleep timer 76
timing diagrams 79
wake up sequence 80
watchdog timer 81
Sleep bit
in CPU_SCR0 register 258
in INT_CLR0 register 242
in INT_MSK0 register 250