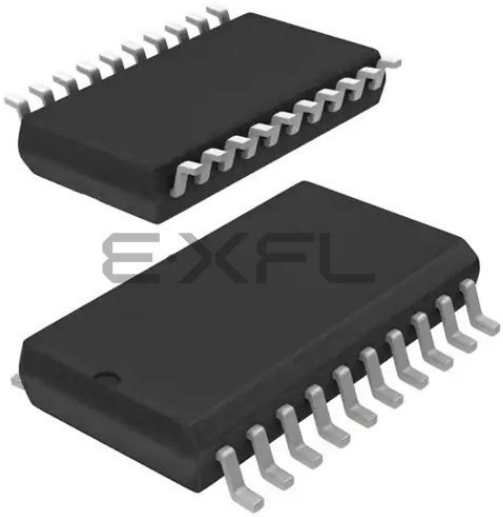

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	M8C
Core Size	8-Bit
Speed	24MHz
Connectivity	SPI, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	16
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.25V
Data Converters	A/D 1x8b, 1x11b, 1x12b; D/A 1x9b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy8c26233-24sit

6.0 I/O Registers

6.1 Port Data Registers

Table 28: Port Data Registers

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	RW	RW	RW	RW	RW	RW	RW	RW
Bit Name	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]
Bit [7:0]: **Data [7:0]** When written is the bits for output on port pins. When read is the state of the port pins								

Port 0 Data Register (PRT0DR, Address = Bank 0, 00h)
Port 1 Data Register (PRT1DR, Address = Bank 0, 04h)
Port 2 Data Register (PRT2DR, Address = Bank 0, 08h)
Port 3 Data Register (PRT3DR, Address = Bank 0, 0Ch)
Port 4 Data Register (PRT4DR, Address = Bank 0, 10h)
Port 5 Data Register (PRT5DR, Address = Bank 0, 14h) **Note**: Port 5 is 4-bits wide, Bit [3:0]

6.2 Port Interrupt Enable Registers

Table 29: Port Interrupt Enable Registers

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W
Bit Name	Int En [7]	Int En [6]	Int En [5]	Int En [4]	Int En [3]	Int En [2]	Int En [1]	Int En [0]
Bit [7:0]: **Int En [7:0]** When written sets the pin interrupt state 0 = Interrupt disabled for pin 1 = Interrupt enabled for pin								

Port 0 Interrupt Enable Register (PRT0IE, Address = Bank 0, 01h)
Port 1 Interrupt Enable Register (PRT1IE, Address = Bank 0, 05h)
Port 2 Interrupt Enable Register (PRT2IE, Address = Bank 0, 09h)
Port 3 Interrupt Enable Register (PRT3IE, Address = Bank 0, 0Dh)
Port 4 Interrupt Enable Register (PRT4IE, Address = Bank 0, 11h)
Port 5 Interrupt Enable Register (PRT5IE, Address = Bank 0, 15h) **Note**: Port 5 is 4-bits wide

6.3.2 Port Drive Mode 1 Registers

Table 32: Port Drive Mode 1 Registers

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W
Bit Name	DM1 [7]	DM1 [6]	DM1 [5]	DM1 [4]	DM1 [3]	DM1 [2]	DM1 [1]	DM1 [0]
Bit [7:0]: **DM1 [7:0]** See truth table for Port Drive Mode 0 Registers, above								

Port 0 Drive Mode 1 Register (PRT0DM1, Address = Bank 1, 01h)
Port 1 Drive Mode 1 Register (PRT1DM1, Address = Bank 1, 05h)
Port 2 Drive Mode 1 Register (PRT2DM1, Address = Bank 1, 09h)
Port 3 Drive Mode 1 Register (PRT3DM1, Address = Bank 1, 0Dh)
Port 4 Drive Mode 1 Register (PRT4DM1, Address = Bank 1, 11h)
Port 5 Drive Mode 1 Register (PRT5DM1, Address = Bank 1, 15h) **Note**: Port 5 is 4-bits wide

6.3.3 Port Interrupt Control 0 Registers

Table 33: Port Interrupt Control 0 Registers

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W
Bit Name	IC0 [7]	IC0 [6]	IC0 [5]	IC0 [4]	IC0 [3]	IC0 [2]	IC0 [1]	IC0 [0]
Bit [7:0]: **IC0 [7:0]** The two Interrupt Control bits that control a particular port pin are treated as a pair and are decoded as follows: IC1 [x], IC0 [x] = 0 0 = Disabled (Default) IC1 [x], IC0 [x] = 0 1 = Falling Edge (-) IC1 [x], IC0 [x] = 1 0 = Rising Edge (+) IC1 [x], IC0 [x] = 1 1 = Change from Last Direct Read								

Port 0 Interrupt Control 0 Register (PRT0IC0, Address = Bank 1, 02h)
Port 1 Interrupt Control 0 Register (PRT1IC0, Address = Bank 1, 06h)
Port 2 Interrupt Control 0 Register (PRT2IC0, Address = Bank 1, 0Ah)
Port 3 Interrupt Control 0 Register (PRT3IC0, Address = Bank 1, 0Eh)
Port 4 Interrupt Control 0 Register (PRT4IC0, Address = Bank 1, 12h)
Port 5 Interrupt Control 0 Register (PRT5IC0, Address = Bank 1, 16h) **Note**: Port 5 is 4-bits wide

6.3.4 Port Interrupt Control 1 Registers

Table 34: Port Interrupt Control 1 Registers

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/ Write	W	W	W	W	W	W	W	W
Bit Name	IC1 [7]	IC1 [6]	IC1 [5]	IC1 [4]	IC1 [3]	IC1 [2]	IC1 [1]	IC1 [0]
Bit [7:0]: **IC1 [7:0]** See truth table for Port Interrupt Control 0 Registers, above								

Port 0 Interrupt Control 1 Register (PRT0IC1, Address = Bank 1, 03h)
Port 1 Interrupt Control 1 Register (PRT1IC1, Address = Bank 1, 07h)
Port 2 Interrupt Control 1 Register (PRT2IC1, Address = Bank 1, 0Bh)
Port 3 Interrupt Control 1 Register (PRT3IC1, Address = Bank 1, 0Fh)
Port 4 Interrupt Control 1 Register (PRT4IC1, Address = Bank 1, 13h)
Port 5 Interrupt Control 1 Register (PRT5IC1, Address = Bank 1, 17h) **Note**: Port 5 is 4-bits wide

7.0 Clocking

7.1 Oscillator Options

7.1.1 Internal Main Oscillator

The internal main oscillator outputs two frequencies, 48 MHz and 24 MHz. In the absence of a high-precision input source from the external oscillator, the accuracy of this circuit is +/- 2.5% (between 0°C and +85°C). No external components are required to achieve this level of accuracy. The Internal Main Oscillator Trim Register (IMO_TR) is used to calibrate this oscillator into specified tolerance. Factory-programmed trim values are available for 5.0V and 3.3V operation. The 5.0V value is loaded in the IMO_TR register upon reset. This register must be adjusted when the operating voltage is outside the range for which factory calibration was set. The factory-programmed trim value is selected using the Table Read Supervisor Call, and is documented in 11.8.

There is an option to phase lock this oscillator to the External Crystal Oscillator. The choice of crystal and its inherent accuracy will determine the overall accuracy of the oscillator. The External Crystal Oscillator must be stable prior to locking the frequency of the Internal Main Oscillator to this reference source.

Table 35: Internal Main Oscillator Trim Register

Bit #	7	6	5	4	3	2	1	0
POR	FS[1]	FS[1]	FS[1]	FS[1]	FS[1]	FS[1]	FS[1]	FS[1]
Read/Write	W	W	W	W	W	W	W	W
Bit Name	IMO Trim [7]	IMO Trim [6]	IMO Trim [5]	IMO Trim [4]	IMO Trim [3]	IMO Trim [2]	IMO Trim [1]	IMO Trim [0]
Bit [7:0]: **IMO Trim [7:0]** Data value stored will alter the trimmed frequency of the Internal Main Oscillator. A larger value in this register will increase the speed of the Internal Main Oscillator								

1. FS = Factory set trim value

Internal Main Oscillator Trim Register (IMO_TR, Address = Bank 1, E8h)

7.1.2 Internal Low Speed Oscillator

An internal low speed oscillator of nominally 32 kHz is available to generate sleep wake-up interrupts and Watchdog resets if the user does not want to attach a 32.768 kHz watch crystal. This oscillator can also be used as a clocking source for the digital PSoC blocks.

The oscillator operates in two different modes. A trim value is written to the Internal Low Speed Oscillator Trim Register (ILO_TR), shown below, upon reset. See section 13.0 for accuracy information. When the IC is put into sleep mode this oscillator drops into an ultra low current state and the accuracy is reduced.

This register sets the adjustment for the Internal Low Speed Oscillator. The value placed in this register is based on factory testing. It is recommended that the user not alter this value.

7.1.5 Phase-Locked Loop (PLL) Operation

The Phase-Locked Loop (PLL) function generates the system clock with crystal accuracy. It is designed to provide a 23.986 MHz oscillator when utilized with an external 32.768 kHz crystal. Although the PLL provides crystal accuracy it requires time to lock onto the reference frequency when first starting. After the External Crystal Oscillator has been selected and enabled, the following procedure should be followed to enable the PLL and allow for proper frequency lock:

1. Select a CPU frequency of 3 MHz or less.
2. Enable the PLL.
3. Wait at least 10 ms.
4. Set CPU to a faster frequency, if desired. To do this, write the bits CPU[2:0] in the OSC_CR0 register.

   The CPU frequency will immediately change when these bits are set.

If the proper settings are selected in PSoC Designer, the above steps are automatically done in *boot.asm*.

7.2 System Clocking Signals

There are twelve system-clocking signals that are used throughout the device. Referenced frequencies are based on use of 32.768 kHz crystal. The names of these signals and their definitions are as follows:

Table 39: System Clocking Signals and Definitions

Signal	Definition
48M	The direct 48 MHz output from the Internal Main Oscillator.
24M	The direct 24 MHz output from the Internal Main Oscillator.
24V1	The 24 MHz output from the Internal Main Oscillator that has been passed through a user-selectable 1 to 16 divider {F = 24 MHz / (1 to 16) = 24 MHz to 1.5 MHz}. The divider value is found in the Oscillator Control 1 Register (OSC_CR1). Note that the divider will be N+1, based on a value of N written into the register bits.
24V2	The 24V1 signal that has been passed through an additional user-selectable 1 to 16 divider {F = 24 MHz / ((1 to 16) * (1 to 16)) = 24 MHz to 93.7 kHz}. The divider value is found in the Oscillator Control 1 Register (OSC_CR1). Note that the divider will be N+1, based on a value of N written into the register bits.
32K	The multiplexed output of either the Internal Low Speed Oscillator or the External Crystal Oscillator.
CPU	The output from the Internal Main Oscillator that has been passed through a divider that has 8 user selectable ratios ranging from 1:1 to 1:256, yielding frequencies ranging from 24 MHz to 93.7 kHz.
SLP	The **32K** system-clocking signal that has been passed through a divider that has 4 user selectable ratios ranging from $1:2^6$ to $1:2^{15}$, yielding frequencies ranging from 512 Hz to 1 Hz. This signal is used to clock the sleep timer period.

7.2.3 Digital PSoC Block Clocking Options

All digital PSoC block clocks are a user-selectable choice of **48M**, **24V1**, **24V2**, or **32K**, as well as clocking signals from other digital PSoC blocks or general purpose I/O pins. There are a total of 16 possible clock options for each digital PSoC block. See the **Digital PSoC Block** section for details.

8.0 Interrupts

8.1 Overview

Interrupts can be generated by the General Purpose I/O lines, the Power monitor, the internal Sleep Timer, the eight Digital PSoC blocks, and the four analog columns. Every interrupt has a separate enable bit, which is contained in the General Interrupt Mask Register (INT_MSK0) and the Digital PSoC Block Interrupt Mask Register (INT_MSK1). When the user writes a “1” to a particular bit position, this enables the interrupt associated with that position. There is a single Global Interrupt Enable bit in the Flags Register (CPU_F), which can disable all interrupts, or enable those interrupts that also have their individual interrupt bit enabled. During a reset, the enable bits in the General Interrupt Mask Register (INT_MASK0), the enable bits in the Digital PSoC Block Interrupt Mask Register (INT_MSK1) and the Global Interrupt Enable bit in the Flags Register (CPU_F) are all cleared. The Interrupt Vector Register (INT_VC) holds the interrupt vector for the highest priority pending interrupt when read, and when written will clear all pending interrupts.

If there is only one interrupt pending and an instruction is executed that would mask that pending interrupt (by clearing the corresponding bit in either of the interrupt mask registers at address E0h or E1h in Bank 0), the CPU will take that interrupt. Since the pending interrupt has been cleared and there are no others, the resulting interrupt vector is 0000h and the CPU will jump to the user code at the beginning of Flash. To address this issue, use the macro defined in *m8c.inc* called "M8C_DisableIntMask" in PSoC Designer. This macro brackets the register write with a disable then an enable of global interrupts.

9.0 Digital PSoC Blocks

9.1 Introduction

PSoC blocks are user configurable system resources. On-chip digital PSoC blocks reduce the need for many MCU part types and external peripheral components. Digital PSoC blocks can be configured to provide a wide variety of peripheral functions. PSoC Designer Software Integrated Development Environment provides automated configuration of PSoC blocks by simply selecting the desired functions. PSoC Designer then generates the proper configuration information and can print a device data sheet unique to that configuration.

Digital PSoC blocks provide up to eight, 8-bit multipurpose timers/counters supporting multiple event timers, real-time clocks, Pulse Width Modulators (PWM), and CRCs. In addition to all PSoC block functions, communication PSoC blocks support full-duplex UARTs and SPI master or slave functions.

As shown in Figure 12:, there are a total of eight 8-bit digital PSoC blocks in this device family configured as a linear array. Four of these are the Digital Basic Type A blocks and four are the Digital Communications Type A blocks. Each of these digital PSoC blocks can be configured independently, or used in combination.

Each digital PSoC block has a unique Interrupt Vector and Interrupt Enable bit. Functions can be stopped or started with a user-accessible Enable bit.

The Timer/Counter/CRC/PRS/Deadband functions are available on the Digital Basic Type A blocks and also the Digital Communications Type A blocks. The UART and SPI communications functions are only available on the Digital Communications Type A blocks.

There are three configuration registers: the Function Register (DBA00FN-DCA07FN) to select the block function and mode, the Input Register (DBA00IN-DCA07IN) to select data input and clock selection, and the Output Register (DBA00OU-DCA07OU) to select and enable function outputs.

The three data registers are designated Data 0 (DBA00DR0-DCA07DR0), Data 1 (DBA00DR1-DCA07DR1), and Data 2 (DBA00DR2-DCA07DR2). The function of these registers and their bit mapping is dependent on the overall block function selected by the user.

The one Control Register (DBA00CR0-DCA07CR0) is designated Control 0. The function of this register and its bit mapping is dependent on the overall block function selected by the user.

If the CPU frequency is 24 MHz and a PSoC timer/counter of 24-bits or longer is operating at 48 MHz, a write to the block Control Register to enable it (for example, a call to Timer_1_Start) may not start the block properly. In the failure case, the first count will typically be indeterminate as the upper bytes fail to make the first count correctly. However, on the first terminal count, the correct period will be loaded and counted thereafter.

Table 51: Digital Function Outputs

Function	Primary Output	Auxiliary Output	Auxiliary Input
Timer	Terminal Count	Compare True	N/A
Counter	Compare True	Terminal Count	N/A
CRC	N/A	Compare True	N/A
PRS	Serial Data	Compare True	N/A
Deadband	F0	F1	N/A
TX UART	TX Data Out	N/A	N/A
RX UART	N/A	N/A	N/A
SPI Master	MOSI	SCLK	N/A
SPI Slave	MISO	N/A	SS_

9.3 Digital PSoC Block Bank 0 Registers

There are four user registers within each digital PSoC block: three data registers, and one status/control register. The three data registers are DR0, which is a shifter/ counter, and DR1 and DR2 registers, which contain data used during the operation. The status/control register (CR0) contains an enable bit that is used for all configurations. In addition, it contains function-specific status and control, which is outlined below.

9.3.1 Digital Basic Type A / Communications Type A Block xx Data Register 0,1,2

Table 52: Digital Basic Type A / Communications Type A Block xx Data Register 0,1,2

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]
Bit Name	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]
Bit [7:0]: **Data [7:0]**								

1. Varies by function/User Module selection. (See Table 53 on page 55.)

Digital Basic Type A Block 00 Data Register 0 (DBA00DR0, Address = Bank 0, 20h)
Digital Basic Type A Block 00 Data Register 1 (DBA00DR1, Address = Bank 0, 21h)
Digital Basic Type A Block 00 Data Register 2 (DBA00DR2, Address = Bank 0, 22h)
Digital Basic Type A Block 01 Data Register 0 (DBA01DR0, Address = Bank 0, 24h)
Digital Basic Type A Block 01 Data Register 1 (DBA01DR1, Address = Bank 0, 25h)
Digital Basic Type A Block 01 Data Register 2 (DBA01DR2, Address = Bank 0, 26h)
Digital Basic Type A Block 02 Data Register 0 (DBA02DR0, Address = Bank 0, 28h)
Digital Basic Type A Block 02 Data Register 1 (DBA02DR1, Address = Bank 0, 29h)
Digital Basic Type A Block 02 Data Register 2 (DBA02DR2, Address = Bank 0, 2Ah)
Digital Basic Type A Block 03 Data Register 0 (DBA03DR0, Address = Bank 0, 2Ch)
Digital Basic Type A Block 03 Data Register 1 (DBA03DR1, Address = Bank 0, 2Dh)
Digital Basic Type A Block 03 Data Register 2 (DBA03DR2, Address = Bank 0, 2Eh)
Digital Communications Type A Block 04 Data Register 0 (DCA04DR0, Address = Bank 0, 30h)
Digital Communications Type A Block 04 Data Register 1 (DCA04DR1, Address = Bank 0, 31h)
Digital Communications Type A Block 04 Data Register 2 (DCA04DR2, Address = Bank 0, 32h)
Digital Communications Type A Block 05 Data Register 0 (DCA05DR0, Address = Bank 0, 34h)
Digital Communications Type A Block 05 Data Register 1 (DCA05DR1, Address = Bank 0, 35h)
Digital Communications Type A Block 05 Data Register 2 (DCA05DR2, Address = Bank 0, 36h)
Digital Communications Type A Block 06 Data Register 0 (DCA06DR0, Address = Bank 0, 38h)
Digital Communications Type A Block 06 Data Register 1 (DCA06DR1, Address = Bank 0, 39h)

Digital Communications Type A Block 06 Data Register 2 (DCA06DR2, Address = Bank 0, 3Ah)
Digital Communications Type A Block 07 Data Register 0 (DCA07DR0, Address = Bank 0, 3Ch)
Digital Communications Type A Block 07 Data Register 1 (DCA07DR1, Address = Bank 0, 3Dh)
Digital Communications Type A Block 07 Data Register 2 (DCA07DR2, Address = Bank 0, 3Eh)

Table 53: R/W Variations per User Module Selection

Function	DR0	R/W	DR1	R/W	DR2	R/W
Timer	Count	R[1]	Period Value	W	Capture Value	RW
Counter	Count	R[1]	Period Value	W	Compare Value	RW
CRC	Current Value/CRC Residue	R[1]	Polynomial Mask Value	W	Seed Value	RW
PRS	Current Value	R[1]	Polynomial Mask Value	W	Seed Value	RW
Deadband	Count	R[1]	Period Value	W	Not Used	RW
RX UART	Shifter	NA	Not Used	NA	Data Register	R
TX UART	Shifter	NA	Data Register	W	Not Used	NA
SPI	Shifter	NA	TX Data Register		RX Data Register	R

1. Each time the register is read, its value is written to the DR2 register.

9.3.2 Digital Basic Type A / Communications Type A Block xx Control Register 0

Table 54: Digital Basic Type A / Communications Type A Block xx Control Register 0

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]	VF[1]
Bit Name	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]
Bit [7:0]: **Data [7:0]**								

1. Varies by function.

Digital Basic Type A Block 00 Control Register 0 (DBA00CR0, Address = Bank 0, 23h)
Digital Basic Type A Block 01 Control Register 0 (DBA01CR0, Address = Bank 0, 27h)
Digital Basic Type A Block 02 Control Register 0 (DBA02CR0, Address = Bank 0, 2Bh)
Digital Basic Type A Block 03 Control Register 0 (DBA03CR0, Address = Bank 0, 2Fh)
Digital Communications Type A Block 04 Control Register 0 (DCA04CR0, Address = Bank 0, 33h)
Digital Communications Type A Block 05 Control Register 0 (DCA05CR0, Address = Bank 0, 37h)
Digital Communications Type A Block 06 Control Register 0 (DCA06CR0, Address = Bank 0, 3Bh)
Digital Communications Type A Block 07 Control Register 0 (DCA07CR0, Address = Bank 0, 3Fh)

9.3.4 Digital Communications Type A Block xx Control Register 0 When Used as UART Transmitter

Table 56: Digital Communications Type A Block xx Control Register 0...

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/ Write	--	--	R	R	--	RW	RW	RW
Bit Name	Reserved	Reserved	TX Complete	TX Reg Empty	Reserved	Parity Type	Parity Enable	Enable

Bit 7: **Reserved**
Bit 6: **Reserved**

Bit 5: **TX Complete**
0 = Indicates that if a transmission has been initiated, it is still in progress
1 = Indicates that the current transmission is complete (including framing bits)
Optional interrupt source for TX UART. Reset when this register is read.

Bit 4: **TX Reg Empty**
0 = Indicates TX Data register is not available to accept another byte (writing to register will cause data to be lost)
1 = Indicates TX Data register is available to accept another byte
Note that the interrupt does not occur until at least 1 byte has been previously written to the TX Data Register
Default interrupt source for TX UART. Reset when the TX Data Register (Data Register 1) is written.

Bit 3: **Reserved**

Bit 2: **Parity Type**
0 = Even
1 = Odd

Bit 1: **Parity Enable**
0 = Parity Disabled
1 = Parity Enabled

Bit 0: **Enable**
0 = Function Disabled
1 = Function Enabled

Digital Communications Type A Block 04 Control Register 0 (DCA04CR0, Address = Bank 0, 33h)
Digital Communications Type A Block 05 Control Register 0 (DCA05CR0, Address = Bank 0, 37h)
Digital Communications Type A Block 06 Control Register 0 (DCA06CR0, Address = Bank 0, 3Bh)
Digital Communications Type A Block 07 Control Register 0 (DCA07CR0, Address = Bank 0, 3Fh)

9.4 Global Inputs and Outputs

Global Inputs and Outputs provide additional capability to route clock and data signals to the digital PSoC blocks. Digital PSoC blocks are connected to the global input and output lines by configuring the PSoC block Input and Output registers (DBA00IN-DCA07IN, DBA00OU-DCA07OU). These global input and output lines form an 8-bit global input bus and an 8-bit global output bus. Four Digital PSoC blocks have access to the upper half of these buses, while the other four access the lower half, per the configuration register. These global input/output buses may be connected to the I/O pins on a per-pin basis using the pin configuration registers. This allows digital PSoC blocks to route their inputs and outputs to pins using the global I/O buses.

9.4.1 Input Assignments

The PSoC block Input Register defines the selection of Global Inputs to digital PSoC blocks. Only 4 of the Global Inputs bus lines are available as selections to a given digital PSoC block as shown in the table below. Once the Global Input has been selected using the PSoC block Input Register selection bits, a GPIO pin must be configured to drive the selected Global Input. This configuration may be set in the Port Global Select Register. The GPIO direction must also be set to input mode by configuring the Port Drive Mode Registers to select High Z.

Table 59: Global Input Assignments

Global Input [7]	Global Input [6]	Global Input [5]	Global Input [4]	Global Input [3]	Global Input [2]	Global Input [1]	Global Input [0]
Port x[7]	Port x[6]	Port x[5]	Port x[4]	Port x[3]	Port x[2]	Port x[1]	Port x[0]
PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03

9.4.2 Output Assignments

The PSoC block Output Register defines the selection of the Global Output bus line to be driven by the digital PSoC blocks. Only 4 of the Global Output bus lines are available as selections to a given digital PSoC block as shown in the table below. The Global Output bus has two functions. Since Global Outputs are also selectable as inputs to digital PSoC blocks, signals can be routed between blocks using this bus. In addition, Global Outputs may drive out to GPIO pins. In this case, once the Global Output has been selected using the PSoC block Output Register selection bits, a GPIO pin must be configured to select the Global Output to drive to the pin. This configuration may be set in the Port Global Select Register. The GPIO direction must also be set to output mode (which is the default) by configuring the Port Drive Mode Registers to one of the available driving strengths.

Table 60: Global Output Assignments

Global Output [7]	Global Output [6]	Global Output [5]	Global Output [4]	Global Output [3]	Global Output [2]	Global Output [1]	Global Output [0]
Port x[7]	Port x[6]	Port x[5]	Port x[4]	Port x[3]	Port x[2]	Port x[1]	Port x[0]
PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 04 PSoC Block 05 PSoC Block 06 PSoC Block 07	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03	PSoC Block 00 PSoC Block 01 PSoC Block 02 PSoC Block 03

9.5 Available Programmed Digital Functionality

9.5.1 Timer with Optional Capture

9.5.1.1 Summary

The timer function continuously measures the amount of time in “ticks” between two events, and provides a rate generator. A down counter lies at the heart of the timer functions. Rate generators divide their clock source by an integer value. Hardware or software generated events

If the SPI Master block is being used to receive data, “dummy” bytes must be written to the TX Data Register in order to initiate transmission/reception of each byte.

9.5.8.3 Inputs

MISO (master-in, slave-out) is selected by the input multiplexer. The clock input multiplexer selects a clock that runs at twice the desired data rate. The SPIM function divides the input clock by 2 to obtain the 50% duty-cycle required for proper timing. The input multiplexer is controlled by the PSoC block Input Register (DCA04IN-DCA07IN).

9.5.8.4 Outputs

There are two outputs, both of which can be enabled onto the Global Output bus. The MOSI (master-out, slave-in) data line provides the output serial data. The second output is the bit-clock derived by dividing the input clock by 2 to ensure a 50% duty-cycle. The PSoC block Output Register (DCA04OU-DCA07OU) controls output options.

Note: The SPIM function does not provide the SS_ signal that may be used by a corresponding SPI Slave. However, this can be implemented with a GPIO pin and supporting firmware if desired.

9.5.8.5 Interrupts

When enabled, the function generates an interrupt on TX Reg Empty status (Data Register 1 empty). If Mode[1] in the Function Register is set, the SPI Master will generate an interrupt on SPI Complete.

9.5.8.6 Usage Notes

1. Reading the Status

   Reading Control Register 0, which contains the status bits, automatically resets the status bits to 0 with the exception of TX Reg Empty, which is cleared when a byte is written to the TX Data Register (Data Register 1), and the RX Reg Full, which is cleared when a byte is read from the RX Data Register (Data Register 2).

2. Using Interrupts

   TX Reg Empty status or optionally SPI Complete status generates the block interrupt. Executing the interrupt routine does not automatically clear status. If SPI Complete is selected as the interrupt source, Control Register 0 (status) must be read in the interrupt routine to clear the status. If TX Reg Empty status is selected, a byte must be written to the TX Data Register (Data Register 1) to clear the status. If the interrupting status is not cleared further interrupts will be suppressed.

9.5.9 SPI Slave - Serial Peripheral Interface (SPIS)

9.5.9.1 Summary

The SPI Slave function provides a full-duplex bi-directional synchronous data transceiver that requires an externally provided bit clock for the data. This function requires a Digital Communications Type PSoC block. It cannot be chained for longer data words. This Digital Communications Type PSoC block supports SPI modes for 0, 1, 2, and 3. See Figure 15: for waveforms of the supported modes.

9.5.9.2 Registers

Data Register 0 provides a shift register for both incoming and outgoing data. Output data is written to Data Register 1 (TX Data Register). Input data is read from Data Register 2 (RX Data Register). When Data Register 0 is empty, its value is updated from Data Register 1. As new data bits are shifted in, the transmit bits are shifted out. After the 8 bits are transmitted and received by Data Register 0, the received byte is transferred into Data Register 2 from which it can be read. Simultaneously, the next byte to transmit, if available, is transferred from Data Register 1 into Data Register 0. Control Register 0 (DCA04CR0-DCA07CR0) provides status information and configures the function for one of the four standard modes, which configure the interface based on clock polarity and phase with respect to data.

9.5.9.3 Inputs

The SPIS function has three inputs. The Input Register (DCA04IN-DCA07IN) controls the input multiplexer, which selects the MOSI data stream. It also controls the clock selection multiplexer from which the function obtains the master’s bit clock. The AUX-IO bits of the Output Register (DCA04OU-DCA07OU) select a Global Input signal from which the SS_ (Slave Select) signal is obtained. It is important to note that the SS_ signal can

only be input from GPIO input pins (Global Input Bus). There is no way to enable the SS_internally. In SPI modes 2 & 3, where SS is not required between each byte, the external pin may be grounded.

Important: The AUX Out Enable bit (bit 5) of the Output Register (DCA04OU-DCA07OU) must be set to 0 to disable it.

9.5.9.4 Outputs

The function output is the MISO (master-in, slave-out) signal, which may be driven on the Global Output bus and is selected by Output Register (DCA04OU-DCA07OU).

9.5.9.5 Interrupts

When enabled, the function generates an interrupt on RX Reg Full status (Data Register 2 full). If Mode[1] of the Function Register is set, the interrupt will be generated on SPI Complete status.

9.5.9.6 Usage Notes

1. Reading the Status

   Reading Control Register 0, which contains the status bits, automatically resets the status bits to 0 with the exception of TX Reg Empty, which is cleared when a byte is written to the TX Data Register (Data Register 1), and the RX Reg Full, which is cleared when a byte is read from the RX Data Register (Data Register 2).

2. Multi-Slave Environment

   The SS_ signal does not have any affect on the output from the slave. The output of the slave at the end of a reception/transmission is always the first bit sent (the MSB, unless LSBF option is selected, then it's the LSB). To implement a multi-slave environment, a GPIO interrupt may be configured on the SS_ input, and the Slave output strength may be toggled between driving and High Z in firmware.

3. Using Interrupts

   RX Reg Full status or SPI Complete status generates an interrupt. Executing the interrupt routine does not automatically clear status. If SPI Complete is selected as the interrupt source, Control Register 0 (status) must be read in the interrupt routine to clear the status. If RX Reg Full status is selected, a byte must be read from the RX Data Register (Data Register 2) to clear the status. If the interrupting status is not cleared further interrupts will be suppressed.

4. Synchronization of CPU Interaction

   Because the SPI Slave is clocked asynchronously by the master SCLK, transfer of data between the TX Register to shifter and shifter to RX Register occurs asynchronously.

   Either polling or interrupts can be used to detect that a byte has been received and is ready to read. However, on the TX side, the user is responsible for implementing a protocol that ensures there is enough set-up time from the TX Data Register write to the first clock (mode 2, 3) or SS_ (mode 0, 1) from the master.

10.6 Analog Clock Select Register

Table 65: Analog Clock Select Register

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/ Write	RW	RW	RW	RW	RW	RW	RW	RW
Bit Name	Reserved	SHDIS	ACLK1 [2]	ACLK1 [1]	ACLK1 [0]	ACLK0 [2]	ACLK0 [1]	ACLK0 [0]

Bit 7: **Reserved**

Bit 6: **SHDIS** During normal operation of an SC block for the amplifier of a column enabled to drive the output bus, the connection is only made for the last half of PHI2 (during PHI1 and for the first half of PHI2, the output bus floats at the last voltage to which it was driven). This forms a sample and hold operation using the output bus and its associated capacitance. This design prevents the output bus from being perturbed by the intermediate states of the SC operation (often a reset state for PHI1 and settling to the valid state during PHI2)

Following are the exceptions: 1) If the ClockPhase bit in CR0 (for the SC block in question) is set to 1, then the output is enabled for the whole of PHI2. 2) If the SHDIS signal is set in bit 6 of the Analog Clock Select Register, then sample and hold operation is disabled for all columns and all enabled outputs of SC blocks are connected to their respective output busses for the entire period of their respective PHI2s

0 = Sample and hold function enabled
1 = Sample and hold function disabled

Bit [5:3]: **ACLK1 [2:0]**
0 0 0 = Digital Basic Type A Block 00
0 0 1 = Digital Basic Type A Block 01
0 1 0 = Digital Basic Type A Block 02
0 1 1 = Digital Basic Type A Block 03
1 0 0 = Digital Communications Type A Block 04
1 0 1 = Digital Communications Type A Block 05
1 1 0 = Digital Communications Type A Block 06
1 1 1 = Digital Communications Type A Block 07

Bit [2:0]: **ACLK0 [2:0]** Same configurations as ACLK1 [2:0]
0 0 0 = Digital Basic Type A Block 00
0 0 1 = Digital Basic Type A Block 01
0 1 0 = Digital Basic Type A Block 02
0 1 1 = Digital Basic Type A Block 03
1 0 0 = Digital Communications Type A Block 04
1 0 1 = Digital Communications Type A Block 05
1 1 0 = Digital Communications Type A Block 06
1 1 1 = Digital Communications Type A Block 07

Analog Clock Select Register (CLK_CR1, Address = Bank 1, 61h)

There are a total of twelve analog PSoC blocks implemented for each of the following types; Analog Continuous Time Type A (ACAxx), Analog Switch Cap Type A (ASAxx), and Analog Switch Cap Type B (ASBxx). These blocks are arranged in an array of three rows by four columns. Each column has one of each type of PSoC block, and the individual PSoC blocks are identified by the row and column in which they reside.

There are two primary types of analog PSoC blocks. Both types contain one op-amp but their principles of operation are quite different. Continuous-time PSoC blocks employ three configuration registers and use resistors to condition amplifier response. Switched capacitor blocks have one comparator and four configuration registers and operate as discrete-time sampling operators. In both types, the configuration registers are

Table 69: Analog Switch Cap Type A Block xx Control 0 Register, continued

Bit 7: **FCap** F Capacitor value selection bit
0 = 16 capacitor units
1 = 32 capacitor units

Bit 6: **ClockPhase** Clock phase select, will invert clocks internal to the blocks. During normal operation of an SC block for the amplifier of a column enabled to drive the output bus, the connection is only made for the last half of PHI2 (during PHI1 and for the first half of PHI2, the output bus floats at the last voltage to which it was driven). This forms a sample and hold operation using the output bus and its associated capacitance. This design prevents the output bus from being perturbed by the intermediate states of the SC operation (often a reset state for PHI1 and settling to the valid state during PHI2)

Following are the exceptions: 1) If the ClockPhase bit in CR0 (for the SC block in question) is set to 1, then the output is enabled for the whole of PHI2. 2) If the SHDIS signal is set in bit 6 of the Analog Clock Select Register, then sample and hold operation is disabled for all columns and all enabled outputs of SC blocks are connected to their respective output busses for the entire period of their respective PHI2s

0 = Internal PHI1 = External PHI1
1 = Internal PHI1 = External PHI2

This bit also affects the latching of the comparator output (CBUS). Both clock phases, PHI1 and PHI2, are involved in the output latching mechanism. The capture of the next value to be output from the latch (capture point event) happens during the falling edge of one clock phase, and the rising edge of the other clock phase will cause the value to come out (output point event). This bit determines which clock phase triggers the capture point event, and the other clock will trigger the output point event. The value output to the comparator bus will remain stable between output point events.

0 = Capture Point Event triggered by Falling PHI2, Output Point Event triggered by Rising PHI1
1 = Capture Point Event triggered by Falling PHI1, Output Point Event triggered by Rising PHI2

Bit 5: **ASign**
0 = Input sampled on Internal PHI1, Reference Input sampled on internal PHI2
1 = Input sampled on Internal PHI2, Reference Input sampled on internal PHI1

Bit [4:0]: **ACap [4:0]** Binary encoding for 32 possible capacitor sizes for A Capacitor:

0 0 0 0 0 = 0 Capacitor units in array	1 0 0 0 0 = 16 Capacitor units in array
0 0 0 0 1 = 1 Capacitor units in array	1 0 0 0 1 = 17 Capacitor units in array
0 0 0 1 0 = 2 Capacitor units in array	1 0 0 1 0 = 18 Capacitor units in array
0 0 0 1 1 = 3 Capacitor units in array	1 0 0 1 1 = 19 Capacitor units in array
0 0 1 0 0 = 4 Capacitor units in array	1 0 1 0 0 = 20 Capacitor units in array
0 0 1 0 1 = 5 Capacitor units in array	1 0 1 0 1 = 21 Capacitor units in array
0 0 1 1 0 = 6 Capacitor units in array	1 0 1 1 0 = 22 Capacitor units in array
0 0 1 1 1 = 7 Capacitor units in array	1 0 1 1 1 = 23 Capacitor units in array
0 1 0 0 0 = 8 Capacitor units in array	1 1 0 0 0 = 24 Capacitor units in array
0 1 0 0 1 = 9 Capacitor units in array	1 1 0 0 1 = 25 Capacitor units in array
0 1 0 1 0 = 10 Capacitor units in array	1 1 0 1 0 = 26 Capacitor units in array
0 1 0 1 1 = 11 Capacitor units in array	1 1 0 1 1 = 27 Capacitor units in array
0 1 1 0 0 = 12 Capacitor units in array	1 1 1 0 0 = 28 Capacitor units in array
0 1 1 0 1 = 13 Capacitor units in array	1 1 1 0 1 = 29 Capacitor units in array
0 1 1 1 0 = 14 Capacitor units in array	1 1 1 1 0 = 30 Capacitor units in array
0 1 1 1 1 = 15 Capacitor units in array	1 1 1 1 1 = 31 Capacitor units in array

Analog Switch Cap Type A Block 10 Control 0 Register (ASA10CR0, Address = Bank 0/1, 80h)
Analog Switch Cap Type A Block 12 Control 0 Register (ASA12CR0, Address = Bank 0/1, 88h)
Analog Switch Cap Type A Block 21 Control 0 Register (ASA21CR0, Address = Bank 0/1, 94h)
Analog Switch Cap Type A Block 23 Control 0 Register (ASA23CR0, Address = Bank 0/1, 9Ch)

10.12 Analog I/O

10.12.1 Analog Input Muxing

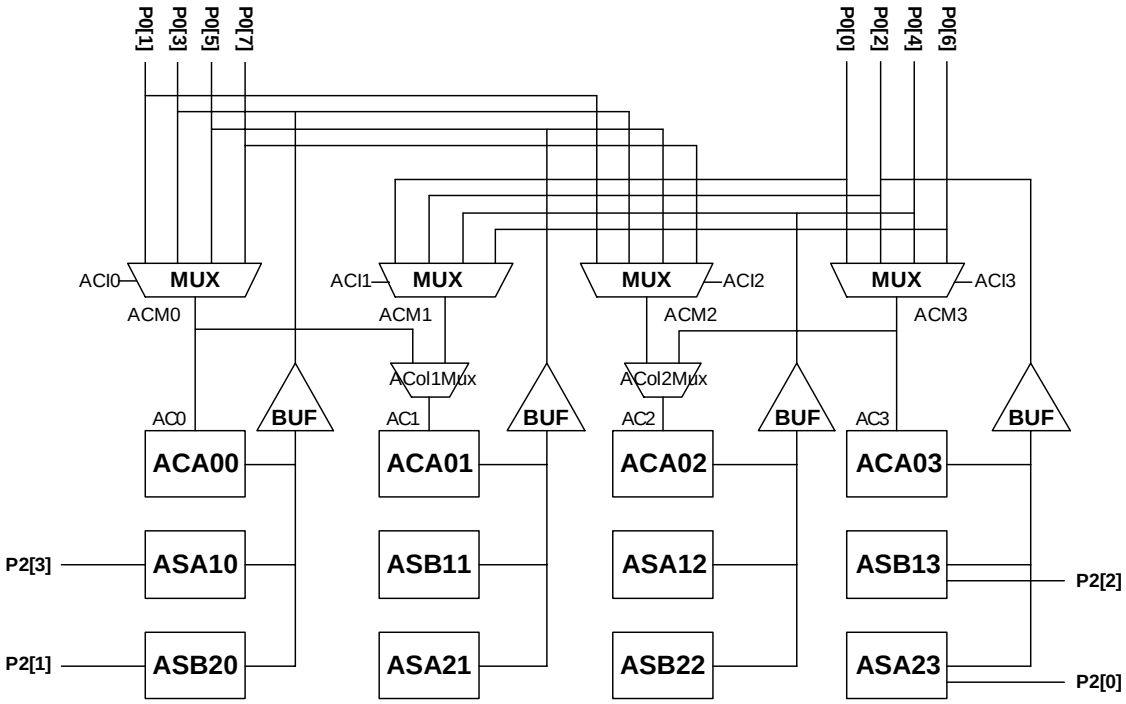

Figure 27: Analog Input Muxing

10.12.2 Analog Input Select Register

This register controls the analog muxes that feed signals in from port pins into each Analog Column. Each of the Analog Columns can have up to four port bits connected to its muxed input. Analog Columns 01 and 02 (ACI1 and ACI2) have additional muxes that allow selection between separate column multiplexers (see Analog Input Muxing diagram above). The AC1Mux and AC2Mux bit fields control the bits for those muxes and are located in the Analog Output Buffer Control Register (ABF_CR). There are four additional analog inputs that go directly into the Switch Capacitor PSoC blocks.

Table 82: Analog Modulator Control Register

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/ Write	RW	RW	RW	RW	RW	RW	RW	RW
Bit Name	Reserved	Reserved	Reserved	Reserved	AMOD2[1]	AMOD2[0]	AMOD0[1]	AMOD0[0]

Bit 7: **Reserved**
Bit 6: **Reserved**
Bit 5: **Reserved**
Bit 4: **Reserved**

Bit [3:2]: **AMOD2[1], AMOD2[0]** Selects the modulation signal for Analog Column 2
0 0 = No Modulation
0 1 = Global Output [0]
1 0 = Global Output [4]
1 1 = Digital Basic Type A Block 03

Bit [1:0]: **AMOD0[1], AMOD0[0]** Selects the modulation signal for Analog Column 0
0 0 = No Modulation
0 1 = Global Output [0]
1 0 = Global Output [4]
1 1 = Digital Basic Type A Block 03

Analog Modulator Control Register (AMD_CR, Address = Bank 1, 63h)

10.14 Analog PSoC Block Functionality

The analog PSoC blocks can be used to implement a wide range of functions, limited only by the designer's imagination. The following functions operate within the capability of the analog PSoC blocks using one analog PSoC block, multiple analog blocks, a combination of more than one *type* of analog block, or a combination of analog and digital PSoC blocks. Most of these functions are currently available as User Modules in PSoC Designer. Others will be added in the future.

- Delta-Sigma A/D Converters
- Successive Approximation A/D Converters
- Incremental A/D Converters
- Programmable Gain/Loss Stage
- Analog Comparators
- Zero-Crossing Detectors
- Low-Pass Filter
- Band-Pass Filter
- Notch Filter
- Amplitude Modulators
- Amplitude Demodulators
- Sine-Wave Generators
- Sine-Wave Detectors
- Sideband Detection
- Sideband Stripping
- Audio Output Drive
- DTMF Generator
- FSK Modulator

By modifying registers, as described in this Data Sheet, users can configure PSoC blocks to perform these functions and more.

12.2 Integrated Development Environment Subsystems

12.2.1 Online Help System

The online help system displays online, context-sensitive help for the user. Designed for procedural and quick reference, each functional subsystem has its own context-sensitive help. This system also provides tutorials and links to FAQs and an Online Support Forum to aid the designer in getting started.

12.2.2 Device Editor

PSoC Designer has several main functions. The Device Editor subsystem lets the user select different onboard analog and digital component configurations for the PSoC blocks. PSoC Designer sets up power-on initialization tables for selected PSoC block configurations and creates source code for an application framework. The framework contains software to operate the selected components and, if the project uses more than one operating configuration, contains routines to switch between different sets of PSoC block configurations at runtime. PSoC Designer can print out a configuration sheet for given project configuration for use during application programming in conjunction with the Device Data Sheet. Once the framework is generated, the user can add application-specific code to flesh out the framework. It's also possible to change the selected components and regenerate the framework.

12.2.3 Assembler

The included CYASM macro assembler supports the M8C microcontroller instruction set and generates a load file ready for device programming or system debugging using the ICE hardware.

12.2.4 C Language Software Development

A C language compiler supports Cypress MicroSystems' PSoC family devices. Even if you have never worked in the C language before, the product quickly allows you to create complete C programs for the PSoC family devices.

The embedded, optimizing C compiler provides all the features of C tailored to the PSoC architecture. It includes a built-in macro assembler allowing assembly code to be merged seamlessly with C code. The link libraries automatically use absolute addressing or can be compiled in relative mode, and linked with other software modules to get absolute addressing.

The compiler comes complete with embedded libraries providing port and bus operations, standard keypad and display support, and extended math functionality.

12.2.5 Debugger

The PSoC Designer Debugger subsystem provides hardware in-circuit emulation, allowing the designer to test the program in a physical system while providing an internal view of the PSoC device. Debugger commands allow the designer to read and write program and data memory, read and write I/O registers, read and write CPU registers, set and clear breakpoints, and provide program run, halt, and step control. The debugger also allows the designer to create a trace buffer of registers and memory locations of interest.

12.3 Hardware Tools

12.3.1 In-Circuit Emulator

A low cost, high functionality ICE is available for development support. This hardware has the capability to program single devices.

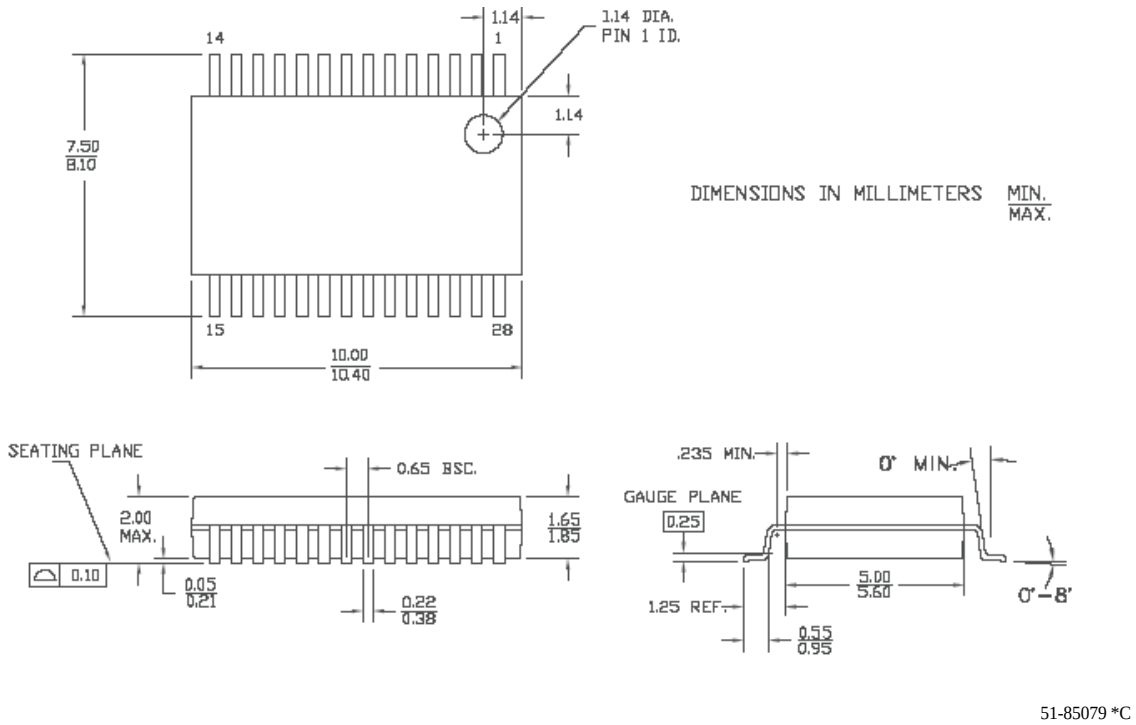

Figure 39: 28-Lead (210-Mil) Shrunk Small Outline Package O28

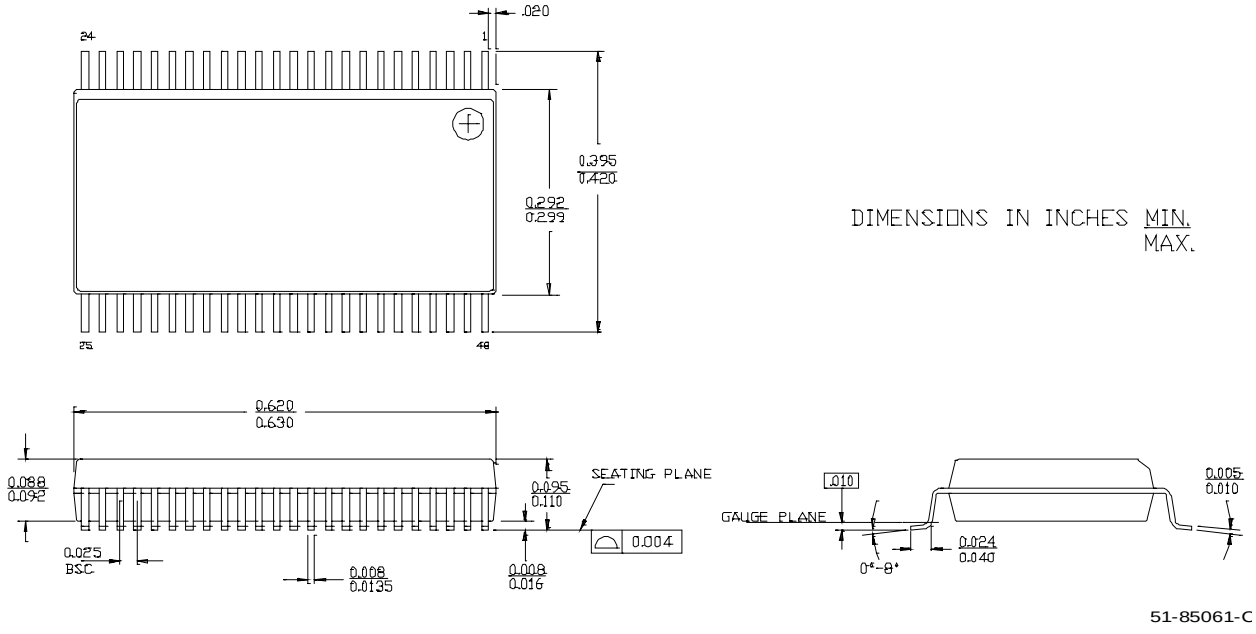

Figure 40: 48-Lead Shrunk Small Outline Package O48

15.0 Ordering Guide

Table 123: Ordering Guide (Leaded)[1]

Type	Ordering Code	Flash (KBytes)	RAM (Bytes)	SMP	Temperature Range
8 Pin (300 Mil) Molded DIP	CY8C25122-24PI	4	256	No	Ind. -40C to +85C
20 Pin (300 Mil) Molded DIP	CY8C26233-24PI	8	256	Yes	Ind. -40C to +85C
20 Pin (300 Mil) Molded SOIC	CY8C26233-24SI	8	256	Yes	Ind. -40C to +85C
20 Pin (210 Mil) SSOP	CY8C26233-24PVI	8	256	Yes	Ind. -40C to +85C
28 Pin (300 Mil) Molded DIP	CY8C26443-24PI	16	256	Yes	Ind. -40C to +85C
28 Pin (300 Mil) Molded SOIC	CY8C26443-24SI	16	256	Yes	Ind. -40C to +85C
28 Pin (210 Mil) SSOP	CY8C26443-24PVI	16	256	Yes	Ind. -40C to +85C
48 Pin (600 Mil) Molded DIP	CY8C26643-24PI[2]	16	256	Yes	Ind. -40C to +85C
48 Pin (300 Mil) SSOP	CY8C26643-24PVI	16	256	Yes	Ind. -40C to +85C
44 Pin Thin Plastic Quad Flatpack	CY8C26643-24AI	16	256	Yes	Ind. -40C to +85C

1. Orders for leaded devices will not be accepted after July 2005.
2. 48-PDIP package not offered Pb-Free.

Table 124: Ordering Guide (Pb-Free Denoted with an “X” in Ordering Code)

Type	Ordering Code	Flash (KBytes)	RAM (Bytes)	SMP	Temperature Range
8 Pin (300 Mil) Molded DIP	CY8C25122-24PXI	4	256	No	Ind. -40C to +85C
20 Pin (300 Mil) Molded DIP	CY8C26233-24PXI	8	256	Yes	Ind. -40C to +85C
20 Pin (300 Mil) Molded SOIC	CY8C26233-24SXI	8	256	Yes	Ind. -40C to +85C
20 Pin (300 Mil) Molded SOIC Tape and Reel	CY8C26233-24SXIT	8	256	Yes	Ind. -40C to +85C
20 Pin (210 Mil) SSOP	CY8C26233-24PVXI	8	256	Yes	Ind. -40C to +85C
20 Pin (210 Mil) SSOP Tape and Reel	CY8C26233-24PVXIT	8	256	Yes	Ind. -40C to +85C
28 Pin (300 Mil) Molded DIP	CY8C26443-24PXI	16	256	Yes	Ind. -40C to +85C
28 Pin (300 Mil) Molded SOIC	CY8C26443-24SXI	16	256	Yes	Ind. -40C to +85C
28 Pin (300 Mil) Molded SOIC Tape and Reel	CY8C26443-24SXIT	16	256	Yes	Ind. -40C to +85C
28 Pin (210 Mil) SSOP	CY8C26443-24PVXI	16	256	Yes	Ind. -40C to +85C
28 Pin (210 Mil) SSOP Tape and Reel	CY8C26443-24PVXIT	16	256	Yes	Ind. -40C to +85C
48 Pin (300 Mil) SSOP	CY8C26643-24PVXI	16	256	Yes	Ind. -40C to +85C
48 Pin (300 Mil) SSOP Tape and Reel	CY8C26643-24PVXIT	16	256	Yes	Ind. -40C to +85C
44 Pin Thin Plastic Quad Flatpack	CY8C26643-24AXI	16	256	Yes	Ind. -40C to +85C
44 Pin Thin Plastic Quad Flatpack Tape and Reel	CY8C26643-24AXIT	16	256	Yes	Ind. -40C to +85C