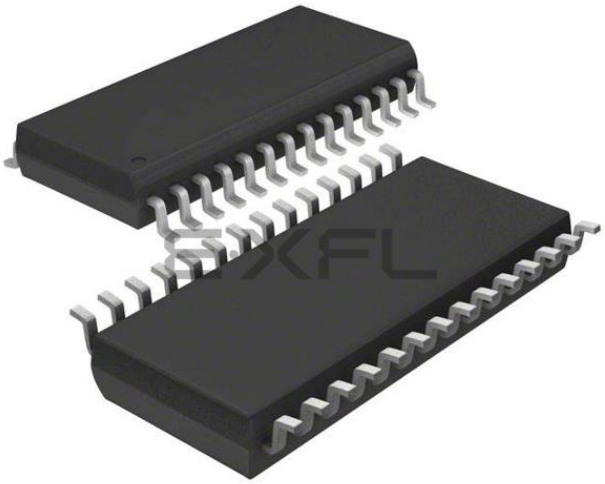

Welcome to E-XFL.COM

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	24
Program Memory Size	16KB (16K 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	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy8c26443-24pvit

2.2 CPU Registers

2.2.1 Flags Register

The Flags Register can only be set or reset with logical instruction.

Table 8: Flags Register

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	1	0
Read/ Write	--	--	--	RW	R	RW	RW	RW
Bit Name	Reserved	Reserved	Reserved	XIO	Super	Carry	Zero	Global IE

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

Bit 4: **XIO** Set by the user to select between the register banks
0 = Bank 0
1 = Bank 1

Bit 3: **Super** Indicates whether the CPU is executing user code or Supervisor Code. (This code cannot be accessed directly by the user and is not displayed in the ICE debugger.)
0 = User Code
1 = Supervisor Code

Bit 2: **Carry** Set by CPU to indicate whether there has been a carry in the previous logical/arithmetic operation
0 = No Carry
1 = Carry

Bit 1: **Zero** Set by CPU to indicate whether there has been a zero result in the previous logical/arithmetic operation
0 = Not Equal to Zero
1 = Equal to Zero

Bit 0: **Global IE** Determines whether all interrupts are enabled or disabled
0 = Disabled
1 = Enabled

2.2.2 Accumulator Register

Table 9: Accumulator Register (CPU_A)

Bit #	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0
Read/Write	System[1]	System[1]	System[1]	System[1]	System[1]	System[1]	System[1]	System[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]** 8-bit data value holds the result of any logical/arithmetic instruction that uses a source addressing mode

1. System - not directly accessible by the user

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.

Table 40: Oscillator Control 0 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	32k Select	PLL Mode	Reserved	Sleep [1]	Sleep [0]	CPU [2]	CPU [1]	CPU [0]

Bit 7: **32k Select**
0 = Internal low precision 32 kHz oscillator
1 = External Crystal Oscillator

Bit 6: **PLL Mode**
0 = Disabled
1 = Enabled, Internal Main Oscillator is locked to External Crystal Oscillator

Bit 5: **Reserved**

Bit [4:3]: **Sleep [1:0]**
0 0 = 512 Hz or 1.95 ms period
0 1 = 64 Hz or 15.6 ms period
1 0 = 8 Hz or 125 ms period
1 1 = 1 Hz or 1 s period

Bit [2:0]: **CPU [2:0]**
0 0 0 = 3 MHz
0 0 1 = 6 MHz
0 1 0 = 12 MHz
0 1 1 = 24 MHz
1 0 0 = 1.5 MHz
1 0 1 = 750 kHz
1 1 0 = 187.5 kHz
1 1 1 = 93.7 kHz

Oscillator Control 0 Register (OSC_CR0, Address = Bank 1, E0h)

Table 41: Oscillator Control 1 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	24V1 [3]	24V1 [2]	24V1 [1]	24V1 [0]	24V2 [3]	24V2 [2]	24V2 [1]	24V2 [0]

Bit [7:4]: **24V1 [3:0]** 4-bit data value determines the divider value for the **24V1** system-clocking signal. Note that the 4-bit data value equals n-1, where n is the desired divider value, as illustrated in PSoC MCU Clock Tree of Signals. See Table 42 on page 41.

Bit [3:0]: **24V2 [3:0]** 4-bit data value determines the divider value for the **24V2** system-clocking signal. Note that the 4-bit data value equals n-1, where n is the desired divider value, as illustrated in the PSoC MCU Clock Tree of Signals. See Table 42 on page 41.

Oscillator Control 1 Register (OSC_CR1, Address = Bank 1, E1h)

Table 45: Digital PSoC Block Interrupt Mask 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	DCA07	DCA06	DCA05	DCA04	DBA03	DBA02	DBA01	DBA00

Bit 7: **DCA07 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 6: **DCA06 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 5: **DCA05 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 4: **DCA04 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 3: **DBA03 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 2: **DBA02 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 1: **DBA01 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Bit 0: **DBA00 Interrupt Enable Bit**
0 = Disabled
1 = Enabled

Digital PSoC Block Interrupt Mask Register (INT_MSK1, Address = Bank 0, E1h)

8.5 Interrupt Vector Register

Table 46: Interrupt Vector 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	Data[7]	Data[6]	Data[5]	Data[4]	Data[3]	Data[2]	Data[1]	Data[0]

Bit [7:0]: **Data [7:0]**
8-bit data value holds the interrupt vector for the highest priority pending interrupt. Writing to this register will clear all pending interrupts

Interrupt Vector Register (INT_VC, Address = Bank 0, E2h)

2. Enabling

   The data input to the Dead-Band function is hardware to the primary output of the previous block, which is typically programmed to be a PWM. The proper order for enabling these blocks (writing the Control Register 0) is PWM first, then Dead-Band.

3. Disabled State

   When the Control Register Enable bit is set to '0', the internal block clock is turned off. A write to Data Register 1 (Period) is loaded directly into Data Register 0 (Counter) to initialize or reset the dead-band time. All outputs are low and the block interrupt is held low.

4. Asserting the Kill Signal

   When the Kill signal is asserted high, both outputs FO and F1 are held low. When the Kill signal is selected from an external source through a Global Input, it is synchronized to the 24 MHz clock and therefore has up to 42 ns of latency.

5. Negating the Kill Signal

   The Kill signal may be negated at any time. However, the output may be enabled at an arbitrary time with respect to the F0 and F1 generation. If exact timing is required when re-enabling the F0 and F1 outputs, the following procedure is recommended:

   1.Kill is asserted.

   2.Write to Control Register 0 to disable the block.

   3.Write to Data Register 1 (Deadband time) to initialize the period.

   4.Kill is eventually negated.

   5.Write to Control Register 0 to enable the block.

9.5.4 PRS - Pseudo-Random Sequence Generator

9.5.4.1 Summary

The PRS function generates an output waveform corresponding to a sequence of pseudo-random numbers. A linear-feedback shift register generates the sequence according to a user-specified polynomial. The width of the numbers in the sequence is variable and the initial value is determined by a user-defined "seed" value. PRS PSoC blocks can be chained to increase the width of the numbers and, hence, the length of the sequence. A chain of N PSoC blocks can generate numbers from 2- to 8N-bits wide and sequences of up to 2^{8N}-1 distinct values.

9.5.4.2 Registers

Data Register 0 implements a linear-feedback shift register. Data Register 2 holds the "seed" value and when the block is disabled, a write to Data Register 2 is loaded directly into Data Register 0 (The block must be disabled when writing this value). Data Register 1 specifies the polynomial and width of the numbers in the sequence (see 9.5.4.6).

9.5.4.3 Inputs

The clock input determines the rate at which the output sequence is produced. The data input must be set to low for the block to function as a PRS. The multiplexer for selecting these inputs is controlled by the PSoC block Input Register (DBA00IN-DCA07IN).

9.5.4.4 Outputs

The PRS function drives the output serial data stream synchronous with the input clock. The output bits change on the rising edge of the input clock. The output may be driven on the Global Output bus or to the subsequent digital PSoC block. The PSoC block Output Register (DBA00OU-DCA07OU) controls output options.

9.5.4.5 Interrupts

The PRS function provides an interrupt based on the Compare signal between Data Register 0 and Data Register 2. Data Register 2 is initially loaded with the "seed" value, and therefore a periodic interrupt will be generated when the PRS count matches the seed value.

10.4 Analog Reference Control

The reference generator establishes a set of three internally fixed reference voltages for the whole chip, AGND, RefHi and RefLo. The 8C26xxx is a single supply part, with no negative voltage available or applicable. Analog ground (AGND) is constructed near mid-supply. This ground is routed to all analog blocks and separately buffered within each block. There may be a small offset voltage between buffered analog grounds, as indicated in the AC/DC Characteristics section. RefHi and RefLo signals are generated, buffered and routed to the analog blocks. RefHi is used to set the conversion range (i.e., span) of analog to digital (ADC) and digital to analog (DAC) converters. RefHi and RefLo can be used to set thresholds in comparators.

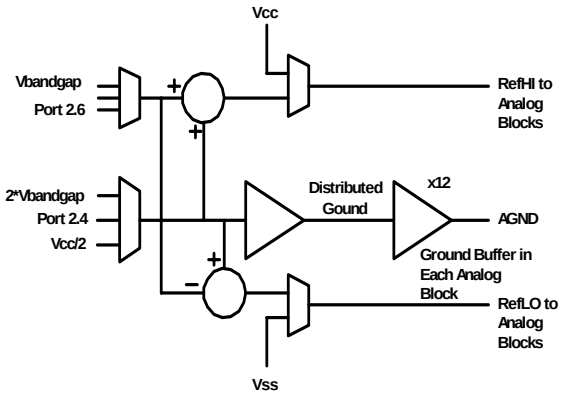

Figure 17: Analog Reference Control Schematic

10.4.1 Bandgap Test

BGT Bandgap Test is used for factory testing of the internal reference voltage testing.

10.4.2 Bias Level

HBE Controls the bias level for all analog functions. It operates with the power setting in each block to set the parameters of that block. Most applications will benefit most from the low bias level. At high bias, the analog block op-amps have faster slew rate but slightly less voltage swing and higher noise.

10.4.3 AGND, RefHI, RefLO

REF Sets Analog Array Reference Control, selecting specific combinations of voltage for analog ground and references. Many of these reference voltages are based on the precision internal reference, a Silicon band gap operating at 1.300 Volts. This reference has good thermal stability and power supply rejection.

Alternatively, the power supply can be scaled to provide analog ground and references; this is particularly useful for signals, which are ratiometric to the power supply voltage.

User supplied external precision references can be connected to Port 2 inputs (available on 28 pin and larger parts). This is useful in setting reference for specific customer applications such as a +/-1.000 V (from AGND) ADC scale. References derived from Port 2 inputs are limited to the same output voltage range as the op-amps in the analog blocks.

Table 63: Analog Reference 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	BGT	HBE	REF[2]	REF[1]	REF[0]	PWR[2]	PWR[1]	PWR[0]

Bit 7: **BGT** Bandgap Test used for internal reference voltage testing (customer should not alter; must be written as 0)

Bit 6: **HBE** Bias level control for op-amps
0 = Low bias mode for analog array
1 = High bias mode for analog array

Bit [5:3]: **REF [2:0]** Analog Array Reference Control

AGND	High/Low
0 0 0 = Vcc/2	± Bandgap
0 0 1 = P2[4]	± P2[6]
0 1 0 = Vcc/2	± Vcc/2
0 1 1 = 2 Bandgap	± Bandgap
1 0 0 = 2 Bandgap	± P2[6]
1 0 1 = P2[4]	± Bandgap
1 1 0 = Reserved	
1 1 1 = Reserved	

Bit [2:0]: **PWR [2:0]** Analog Array Power Control
0 0 0 = All Analog Off
0 0 1 = SC Off, REFPWR Low
0 1 0 = SC Off, REFPWR Med
0 1 1 = SC Off, REFPWR High
1 0 0 = All Analog Off
1 0 1 = SC On, REFPWR Low
1 1 0 = SC On, REFPWR Med
1 1 1 = SC On, REFPWR High

Analog Reference Control Register (ARF_CR, Address = Bank 0, 63h)

10.5 Analog PSoC Block Clocking Options

All analog PSoC blocks in a particular Analog Column share the same clock signal. Choosing the clocking for an analog PSoC block is a two-step process.

1. First, if the user wants to use the **ACLK0** and **ACLK1** system-clocking signals, the digital PSoC blocks that serve as the source for these signals must be selected. This selection is made in the Analog Clock Select Register (CLK_CR1).
2. Next, the user must select the source for the **Acolumn0**, **Acolumn1**, **Acolumn2**, and **Acolumn3** system-clocking signals. The user will choose the clock for Acolumnx[1:0] bits in the Analog Column Clock Select Register (CLK_CR0). Each analog PSoC block in a particular Analog Column is clocked from the **Acolumn[x]** system-clocking signal for that column. (Note that the Acolumn[x] signals have a 1:4 divider on them.)

10.5.1 Analog Column Clock Select Register

Table 64: Analog Column 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	Acolumn3 [1]	Acolumn3 [0]	Acolumn2 [1]	Acolumn2 [0]	Acolumn1 [1]	Acolumn1 [0]	Acolumn0 [1]	Acolumn0 [0]

Bit [7:6]: **Acolumn3 [1:0]**
0 0 = 24V1
0 1 = 24V2
1 0 = ACLK0
1 1 = ACLK1

Bit [5:4]: **Acolumn2 [1:0]**
0 0 = 24V1
0 1 = 24V2
1 0 = ACLK0
1 1 = ACLK1

Bit [3:2]: **Acolumn1 [1:0]**
0 0 = 24V1
0 1 = 24V2
1 0 = ACLK0
1 1 = ACLK1

Bit [1:0]: **Acolumn0 [1:0]**
0 0 = 24V1
0 1 = 24V2
1 0 = ACLK0
1 1 = ACLK1

Analog Column Clock Select Register (CLK_CR0, Address = Bank 1, 60h)

10.7.2.2 Analog Continuous Time Block xx Control 1 Register

The PMux bits control the multiplexing of inputs to the non-inverting input of the op-amp. There are physically only 7 inputs.

The 8th code (111) will leave the input floating. This is not desirable, and should be avoided.

The NMux bits control the multiplexing of inputs to the inverting input of the op-amp. There are physically only 7 inputs.

CompBus controls a tri-state buffer that drives the comparator logic. If no PSoC block in the analog column is driving the comparator bus, it will be driven low externally to the blocks.

AnalogBus controls the analog output bus. A CMOS switch connects the op-amp output to the analog bus.

Table 67: Analog Continuous Time Block xx Control 1 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	AnalogBus	CompBus	NMux2	NMux1	NMux0	PMux2	PMux1	PMux0

Bit 7: **AnalogBus** Enable output to the analog bus
0 = Disable analog bus driven by this block
1 = Enable analog bus driven by this block
Bit 6: **CompBus** Enable output to the comparator bus
0 = Disable comparator bus driven by this block
1 = Enable comparator bus driven by this block

Bit [5:3]: **NMux [2:0]** Encoding for negative input select

	ACA00	ACA01	ACA02	ACA03
0 0 0 =	ACA01	ACA00	ACA03	ACA02
0 0 1 =	AGND	AGND	AGND	AGND
0 1 0 =	REFLO	REFLO	REFLO	REFLO
0 1 1 =	REFHI	REFHI	REFHI	REFHI
1 0 0[1] =	ACA00	ACA01	ACA02	ACA03
1 0 1 =	ASA10	ASB11	ASA12	ASB13
1 1 0 =	ASB11	ASA10	ASB13	ASA12
1 1 1 =	Reserved	Reserved	Reserved	Reserved

Bit [2:0]: **PMux [2:0]** Encoding for positive input select

	ACA00	ACA01	*ACA02*	ACA03
0 0 0 =	REFLO	ACA02	ACA01	REFLO
0 0 1 =	Port Inputs	Port Inputs	Port Inputs	Port Inputs
0 1 0 =	ACA01	ACA00	ACA03	ACA02
0 1 1 =	AGND	AGND	AGND	AGND
1 0 0 =	ASA10	ASB11	ASA12	ASB13
1 0 1 =	ASB11	ASA10	ASB13	ASA12
1 1 0 =	ABUS0	ABUS1	ABUS2	ABUS3
1 1 1 =	Reserved	Reserved	Reserved	Reserved

1. This in fact is the feedback input of the MUX.

Analog Continuous Time Block 00 Control 1 Register (ACA00CR1, Address = Bank 0/1, 72h)
Analog Continuous Time Block 01 Control 1 Register (ACA01CR1, Address = Bank 0/1, 76h)
Analog Continuous Time Block 02 Control 1 Register (ACA02CR1, Address = Bank 0/1, 7Ah)
Analog Continuous Time Block 03 Control 1 Register (ACA03CR1, Address = Bank 0/1, 7Eh)

11.0 Special Features of the CPU

11.1 Multiplier/Accumulator

A fast, on-chip signed 2's complement MAC (Multiply/Accumulate) function is provided to assist the main CPU with digital signal processing applications. Multiply results, as well as the lower 2 bytes of the Accumulator, are available immediately after the input registers are written. The upper 2 bytes require a single instruction delay before reading. The MAC function is tied directly on the internal data bus, and is mapped into the register space. The following MAC block diagram provides data flow information. The user has the choice to either cause a multiply/accumulate function to take place, or a multiply only function. The user selects which operation is performed by the choice of input register. The multiply function occurs immediately whenever the MUL_X or the MUL_Y multiplier input registers are written, and the result is available in the MUL_DH and MUL_DL multiplier result registers. The Multiply/Accumulate function is executed whenever there is a write to the MAC_X or the MAC_Y Multiply/Accumulate input registers, and the result is available in the ACC_DR3, ACC_DR2, ACC_DR1, and ACC_DR0 accumulator result registers. A write to MUL_X or MAC_X is input as the X value to both the multiply and Multiply/Accumulate functions. A write to MUL_Y or MAC_Y is input as the Y value to both the multiply and Multiply/Accumulate functions. A write to the MAC_CL0 or MAC_CL1 registers will clear the value in the four accumulate registers.

Operation of the Multiply/Accumulate function relies on proper multiplicand input. The first value of each multiplicand must be placed into MUL_X (or MUL_Y) register to avoid causing a Multiply/Accumulate to occur. The second multiplicand must be placed into MAC_Y (or MAC_X) thereby triggering the Multiply/Accumulate function.

MUL_X, MUL_Y, MAC_X, and MAC_Y are 8-bit signed input registers. MUL_DL and MUL_DH form a 16-bit signed output. ACC_DR0, ACC_DR1, ACC_DR2 and ACC_DR3 form a 32-bit signed output.

An extra instruction must be inserted between the following sequences of MAC operations to provide extra delay. If this is not done, the Accumulator results will be inaccurate.

a. Two MAC instructions in succession:

```
mov reg[MAC_X],a
nop //add nop or any other instruction
mov reg[MAC_X],a
```

For sequence a., there is no workaround, the nop or other instruction must be inserted.

b. A MAC instruction followed by a read of the most significant Accumulator bytes:

```
mov reg[MAC_X],a
nop //add nop or any other instruction
mov a,[ACC_DR2] // or ACC_DR3
```

For sequence b., the least significant Accumulator bytes (ACC_DR0, ACC_DR1) may be reliably read directly after the MAC instruction.

Writing to the multiplier registers (MUL_X, MUL_Y), and reading the result back from the multiplier product registers (MUL_DH, MUL_DL), is not affected by this problem and does not have any restrictions.

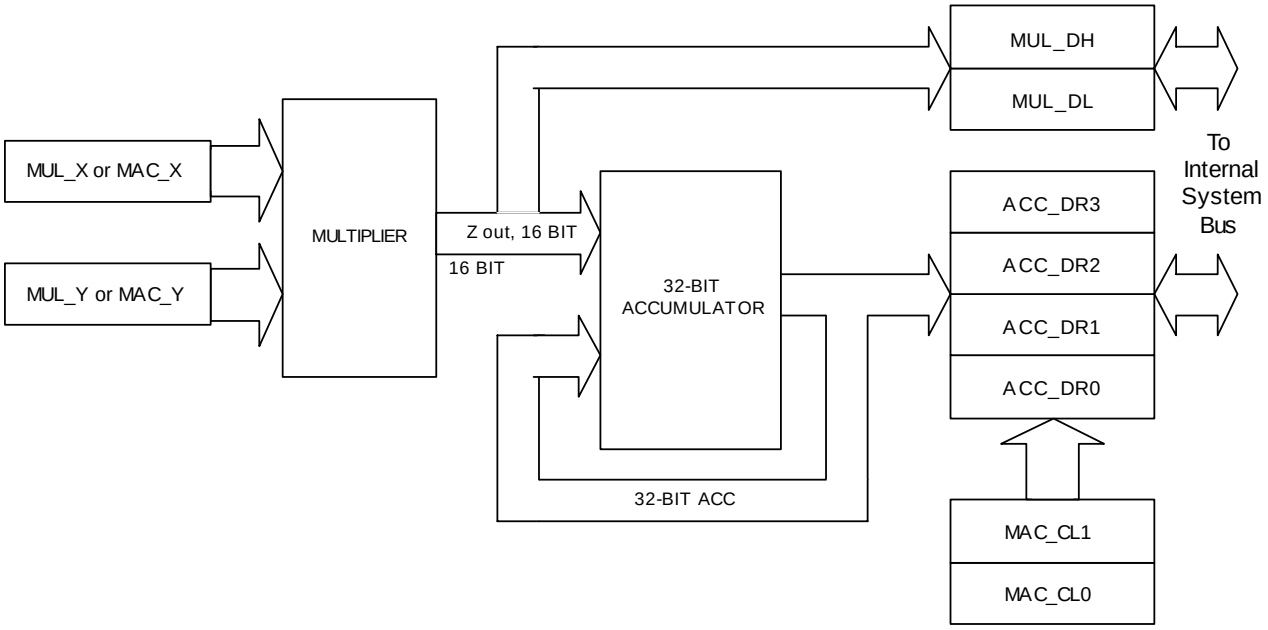

Figure 29: Multiply/Accumulate Block Diagram

Table 83: Multiply Input X Register

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
Name	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]
Bit [7:0]: **Data [7:0]** 8-bit data is the input value for X multiplier								

Multiply Input X Register (MUL_X, Address = Bank 0, E8h)

Table 84: Multiply Input Y Register

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	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]
Bit [7:0]: **Data [7:0]** 8-bit data is the input value for Y multiplier								

Multiply Input Y Register (MUL_Y, Address = Bank 0, E9h)

Table 89: Accumulator Result 3 / Multiply/Accumulator Clear 0 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	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]

Bit [7:0]: **Data [7:0]**
8-bit data value when read is the highest order result of the multiply/accumulate function
Any 8-bit data value when written will cause all four Accumulator result registers to clear

Accumulator Result 3 / Multiply/Accumulator Clear 0 Register (ACC_DR3 / MAC_CL0, Address = Bank 0, EEh)

Table 90: Accumulator Result 2 / Multiply/Accumulator Clear 1 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	Data [7]	Data [6]	Data [5]	Data [4]	Data [3]	Data [2]	Data [1]	Data [0]

Bit [7:0]: **Data [7:0]**
8-bit data value when read is next to highest order result of the multiply/accumulate function
Any 8-bit data value when written will cause all four Accumulator result registers to clear

Accumulator Result 2 / Multiply/Accumulator Clear 1 Register (ACC_DR2 / MAC_CL1, Address = Bank 0, EFh)

11.2 Decimator

The output of a ΔΣ modulator is a high-speed, single bit A/D converter. A single bit A/D converter is of little use to anyone and must be converted to a lower speed multiple bit output. Converting this high-speed single bit data stream to a lower speed multiple bit data stream requires a data decimator.

A “divide by n” decimator is a digital filter that takes the single bit data at a fast rate and outputs multiple bits at one n^{th} the speed. For a single stage ΔΣ converter, the optimal filter has a $sinc^2$ response. This filter can be implemented as a finite impulse response (FIR) filter and for a “divide by n” implementation should have the following coefficients:

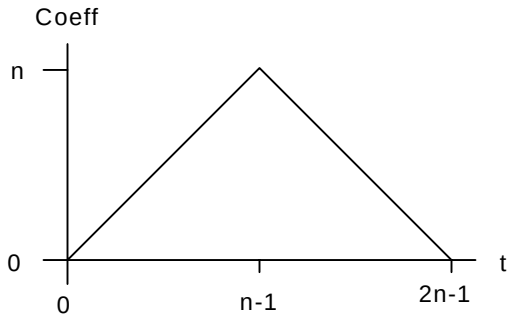

Figure 30: Decimator Coefficients

interrupt will wake the part from sleep. The Stop bit in the Status and Control Register (CPU_SCR) must be cleared for a part to resume out of sleep.

Any digital PSoC block that is clocked by a System Clock other than the **32K** system-clocking signal or external pins will be stopped, as these clocks do not run in sleep mode.

The Internal Main Oscillator restarts immediately on exiting either the Full Sleep or CPU Sleep modes. Analog functions must be re-enabled by firmware. If the External Crystal Oscillator is used and the internal PLL is enabled, the PLL will take many cycles to change from its initial 2.5% accuracy to track that of the External Crystal Oscillator. If the PLL is enabled, there will be a 30µs (one full **32K** cycle) delay hold-off time for the CPU to let the VCO and PLL stabilize. If the PLL is not enabled, the hold-off time is one half of the **32K** cycle. For further details on PLL, see 7.0.

The Sleep interrupt allows the microcontroller to wake up periodically and poll system components while maintaining very low average power consumption. The sleep interrupt may also be used to provide periodic interrupts during non-sleep modes.

In System Sleep State, GPIO Pins P2[4] and P2[6] should be held to a logic low or a false Low Voltage Detect interrupt may be triggered. The cause is in the System Sleep State, the internal Bandgap reference generator is turned off and the reference voltage is maintained on a capacitor.

The circumstances are that during sleep, the reference voltage on the capacitor is refreshed periodically at the sleep system duty cycle. Between refresh cycles, this voltage may leak slightly to either the positive supply or ground. If pins P2[4] or P2[6] are in a high state, the leakage to the positive supply is accelerated (especially at high temperature). Since the reference voltage is compared to the supply to detect a low voltage condition, this accelerated leakage to the positive supply voltage will cause that voltage to appear lower than it actually is, leading to the generation of a false Low Voltage Detect interrupt.

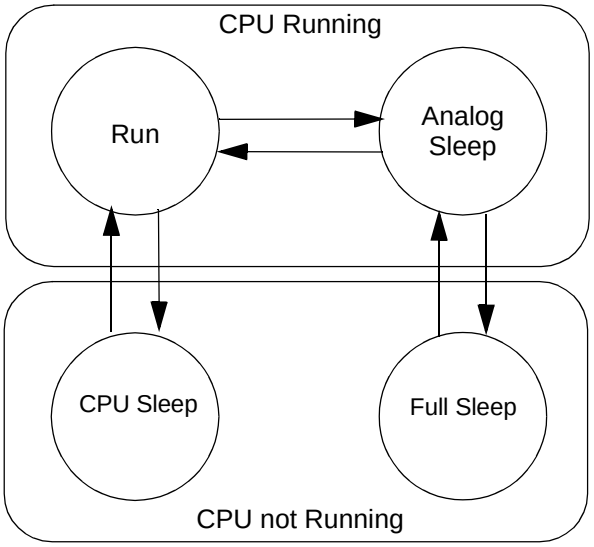

Figure 32: Three Sleep States

11.8.1 Additional Function for Table Read Supervisory Call

The Table Read supervisory operation will return the Version ID in the Accumulator. The value in the Accumulator is divided into a high and low nibble, indicating major and minor revisions, respectively. **Note**: The value in the X register is modified during the Table Read Supervisory Call, and must be saved and restored if needed after the call completes.

- A[7:4]: Major silicon revisions.
- A[3:0]: Minor silicon revisions.

Table 99: Table Read for Supervisory Call Functions

Table ID	Function	TV(0)	TV(1)	TV(2)	TV(3)	TV(4)	TV(5)	TV(6)	TV(7)
00[1]	Production Silicon ID	Silicon ID 1	Silicon ID 0	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
01	Provides trim value for Internal Main Oscillator and Internal Voltage Reference	Internal Voltage Reference trim value for 3.3V	Internal Main Oscillator trim value for 3.3V	Reserved	Reserved	Internal Voltage Reference trim value for 5.0V	Internal Main Oscillator trim value for 5.0V	Reserved	Reserved

1. Determines silicon revision values in Accumulator and X registers.

11.9 Flash Program Memory Protection

The user has the option to define the access to the Flash memory. A flexible system allows the user to select one of four protection modes for each 64-byte block within the Flash, based on the particular application. The protection mechanism is implemented by a device programmer using the System Supervisor Call. When this command is executed, two bits within the data programmed into the Flash will select the protection mode. It is not intended that the protection byte will be modified by the user's code. The following table lists the available protection options:

Table 100: Flash Program Memory Protection

Mode Bits	Mode Name	External Read	External Write	Internal Write
00	Unprotected	Enabled	Enabled	Enabled
01	Factory Upgrade	Disabled	Enabled	Enabled
10	Field Upgrade	Disabled	Disabled	Enabled
11	Full Protection	Disabled	Disabled	Disabled

Note: Mode 10 is the default.

11.10 Programming Requirements and Step Descriptions

The pins in the following table are critical for the programmer:

Table 101: Programmer Requirements

Pin Name	Function	Programmer HW Pin Requirements
SDATA	Serial Data In/Out	Drive TTL Levels, Read TTL, High Z
SCLK	Serial Clock	Drive TTL levEl Clock Signal
V_{ss}	Power Supply Ground Connection	Low Resistance Ground Connection
V_{cc}	Power Supply Positive Voltage	0V, 3.0V, 5V, & 5.4V. 0.1V Accuracy. 20mA Current Capability

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.

13.2.5 Switch Mode Pump Specifications

Table 111: DC Switch Mode Pump Specifications

Symbol	DC Switch Mode Pump Specifications	Minimum	Typical	Maximum	Unit
	Output Voltage[1]	3.07	-	5.15	V
	Available Output Current V_i = 1.5 V, V_o = 3.25 V V_i = 1.5 V, V_o = 5.0 V	 8[2] 5	 - -	 - -	 mA mA
	Short Circuit Current (V_i = 3.3 V)	-	12	-	mA
	Input Voltage Range (During sustained operation)	1.0	-	3.3	V
	Minimum Input Voltage to Start Pump	1.1	1.2	-	
	Output Voltage Tolerance (Over V_i Range)	-	5	-	%V_o
	Line Regulation (Over V_i Range)	-	5	-	%V_o
	Load Regulation	-	5	-	%V_o
	Output Voltage Ripple (Depends on capacitor and load)	-	25[3]	-	mV_{pp}
	Transient Response 50% Load Change to 5% error envelope V_o Over/Undershoot for 50% Load Change	 - -	 1 1	 - -	 µs %V_o
	Efficiency	35[4]	50	-	%
	Switching Frequency	-	1.3	-	MHz
	Switching Duty Cycle	-	50	-	%

1. Average, neglecting ripple.
2. For implementation, which includes 2 µH inductor, 1 µF capacitor, and Schottkey diode. Performance is significantly a function of external components. Specifications guaranteed for inductors with series resistance less than 0.1 W, with a current rating of > 250 mA, a capacitor with less than 1µA leakage at 5V, and Schottkey diode with less than 0.6V of drop at 50 mA.
3. Configuration of note 2. Load is 5 mA.
4. Configuration of note 2. Load is 5 mA. V_{out} is 3.25V.

13.3.2 AC Analog Output Buffer Specifications

The following table lists guaranteed maximum and minimum specifications for the voltage and temperature ranges, 5V +/- 5% and –40°C <= T_A <= 85°C. Typical parameters are provided for design guidance only. Typical parameters apply to 5V and 25°C. For 3.3V operation, see Table 120 on page 142.

Table 119: 5V AC Analog Output Buffer Specifications

Symbol	5V AC Analog Output Buffer Specifications	Minimum	Typical	Maximum	Unit
	Rising Settling Time to 0.1%, 1V Step, 100pF Load				
	Bias = Low	-	-	2.5	µs
	Bias = High	-	-	2.5	µs
	Falling Settling Time to 0.1%, 1V Step, 100pF Load				
	Bias = Low	-	-	2.2	µs
	Bias = High	-	-	2.2	µs
	Rising Slew Rate (20% to 80%), 1V Step, 100pF Load				
	Bias = Low	.9	-	-	V/µs
	Bias = High	.9	-	-	V/µs
	Falling Slew Rate (80% to 20%), 1V Step, 100pF Load				
	Bias = Low	.9	-	-	V/µs
	Bias = High	.9	-	-	V/µs
	Small Signal Bandwidth, 20mV_{pp}, 3dB BW, 100pF Load				
	Bias = Low	1.5	-	-	MHz
	Bias = High	1.5	-	-	MHz
	Large Signal Bandwidth, 1V_{pp}, 3dB BW, 100pF Load				
	Bias = Low	600	-	-	kHz
	Bias = High	600	-	-	kHz

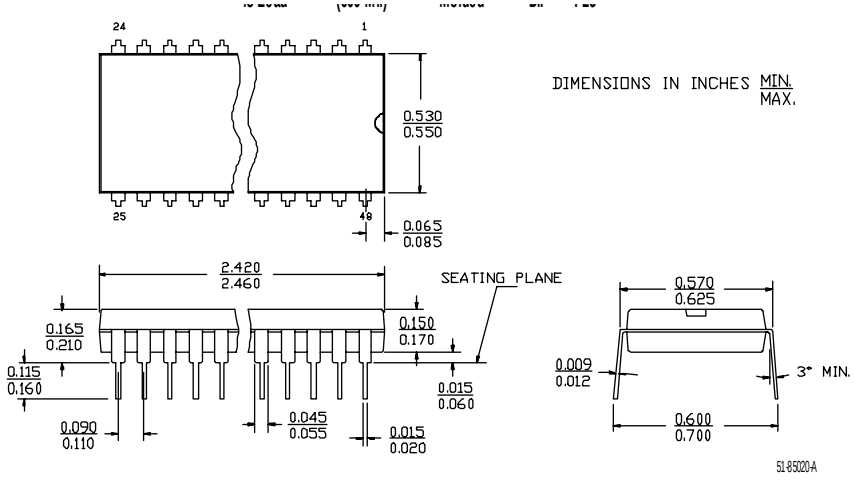

Figure 43: 48-Lead (600-Mil) Molded DIP P25

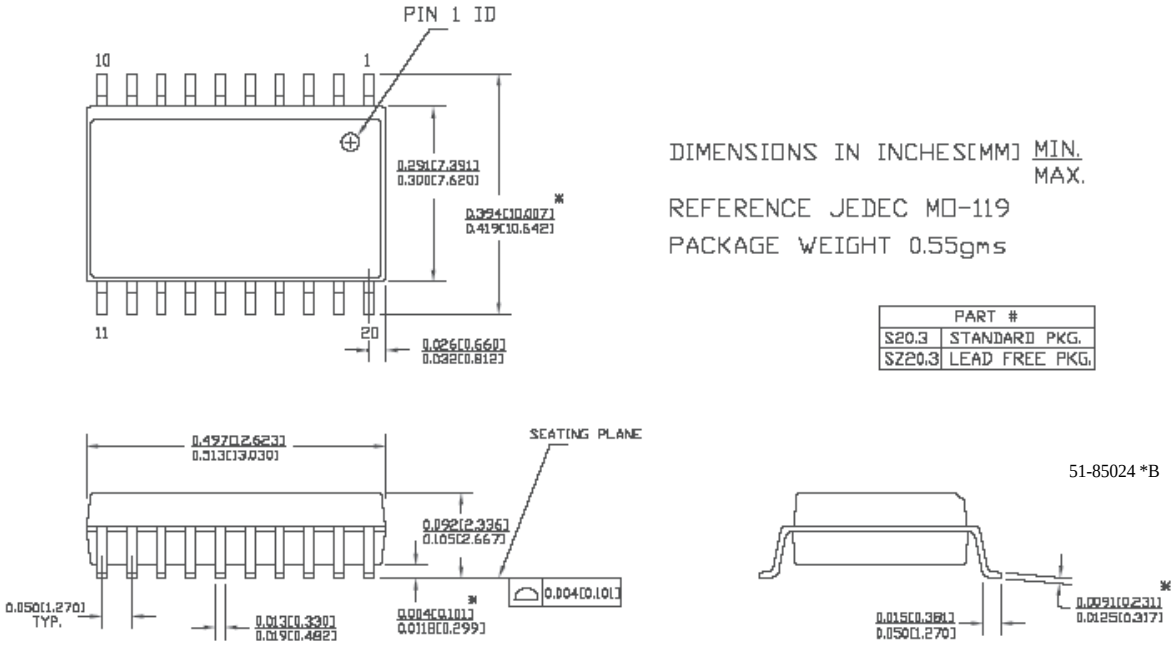

Figure 44: 20-Lead (300-Mil) Molded SOIC S5