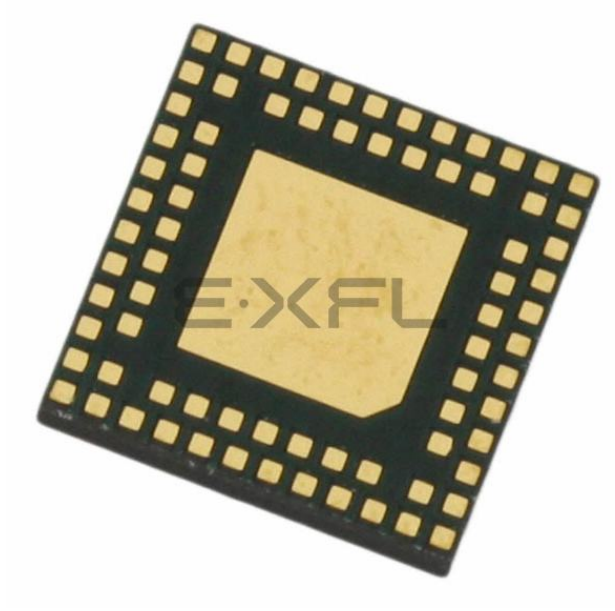

Welcome to E-XFL.COM

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	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	DMA, LCD, POR, PWM, WDT
Number of I/O	57
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8.25K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.8V
Data Converters	A/D 16x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	76-VFQFN Dual Rows, Exposed Pad
Supplier Device Package	76-DQFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f964-a-gmr

1.3. Serial Ports

The C8051F96x Family includes an SMBus/I^{2}C interface, a full-duplex UART with enhanced baud rate configuration, and two Enhanced SPI interfaces. Each of the serial buses is fully implemented in hardware and makes extensive use of the CIP-51's interrupts, thus requiring very little CPU intervention.

1.4. Programmable Counter Array

An on-chip Programmable Counter/Timer Array (PCA) is included in addition to the four 16-bit general purpose counter/timers. The PCA consists of a dedicated 16-bit counter/timer time base with six programmable capture/compare modules. The PCA clock is derived from one of six sources: the system clock divided by 12, the system clock divided by 4, Timer 0 overflows, an External Clock Input (ECI), the system clock, or the external oscillator clock source divided by 8.

Each capture/compare module can be configured to operate in a variety of modes: edge-triggered capture, software timer, high-speed output, pulse width modulator (8, 9, 10, 11, or 16-bit), or frequency output. Additionally, Capture/Compare Module 5 offers watchdog timer (WDT) capabilities. Following a system reset, Module 5 is configured and enabled in WDT mode. The PCA Capture/Compare Module I/O and External Clock Input may be routed to Port I/O via the Digital Crossbar.

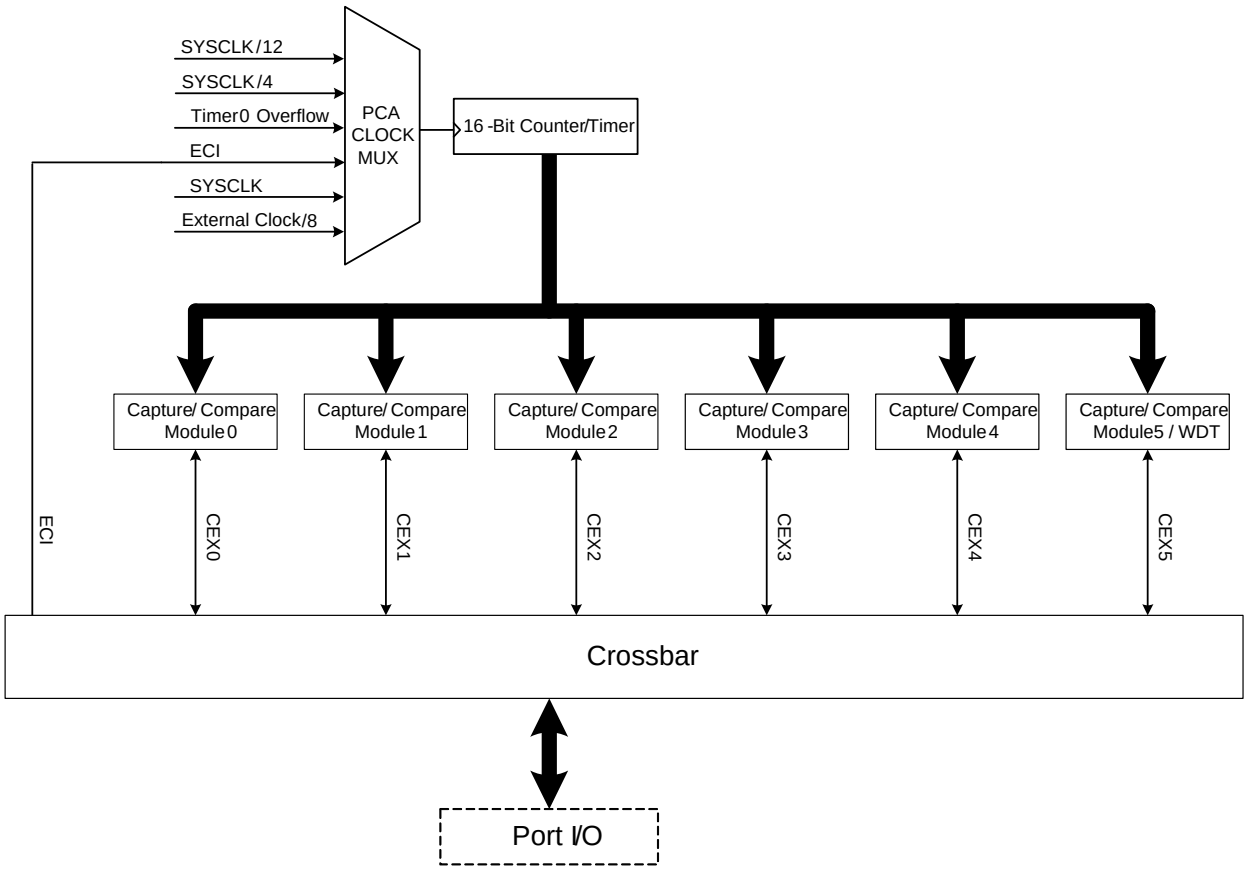

Figure 1.12. PCA Block Diagram

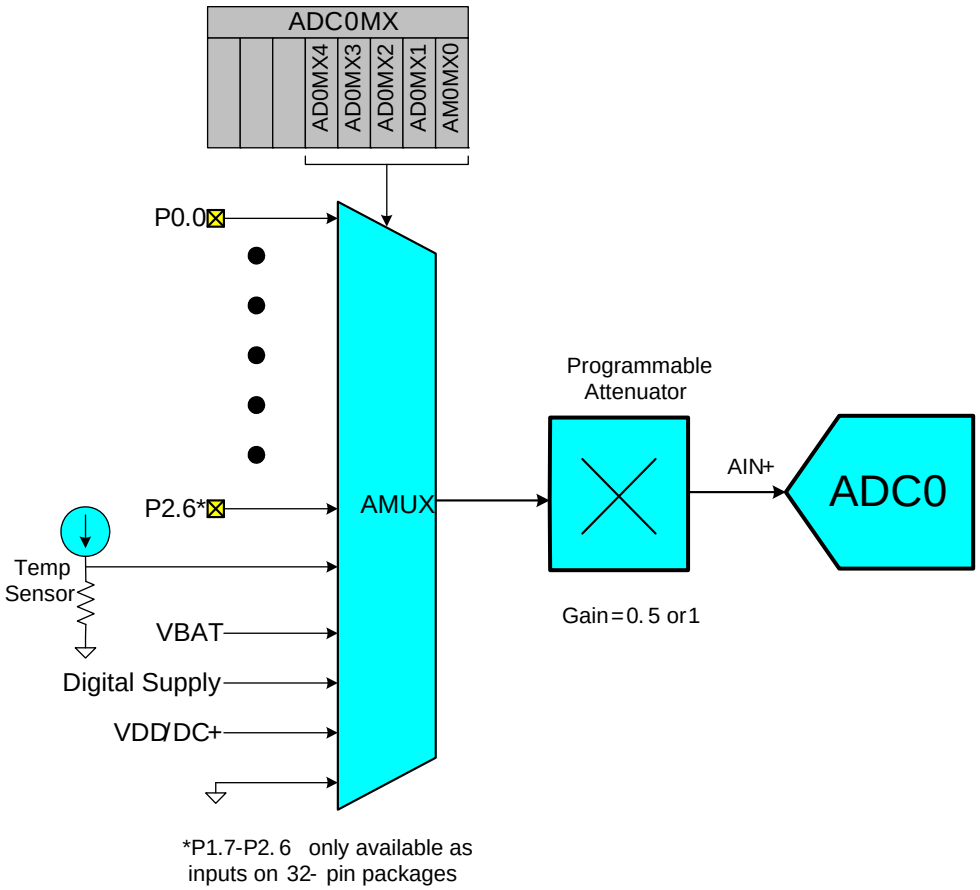

Figure 1.14. ADC0 Multiplexer Block Diagram

1.6. Programmable Current Reference (IREF0)

C8051F96x devices include an on-chip programmable current reference (source or sink) with two output current settings: low power mode and high current mode. The maximum current output in low power mode is 63 µA (1 µA steps) and the maximum current output in high current mode is 504 µA (8 µA steps).

1.7. Comparators

C8051F96x devices include two on-chip programmable voltage comparators: Comparator 0 (CPT0) which is shown in Figure 1.15; Comparator 1 (CPT1) which is shown in Figure 1.16. The two comparators operate identically but may differ in their ability to be used as reset or wake-up sources. See Section “22. Reset Sources” on page 278 and the Section “19. Power Management” on page 257 for details on reset sources and low power mode wake-up sources, respectively.

The Comparator offers programmable response time and hysteresis, an analog input multiplexer, and two outputs that are optionally available at the Port pins: a synchronous “latched” output (CP0, CP1), or an asynchronous “raw” output (CP0A, CP1A). The asynchronous CP0A signal is available even when the system clock is not active. This allows the Comparator to operate and generate an output when the device is in some low power modes.

The comparator inputs may be connected to Port I/O pins or to other internal signals. Port pins may also be used to directly sense capacitive touch switches. See Application Note AN338 for details on Capacitive Touch Switch sensing.

5. SAR ADC with 16-bit Auto-Averaging Accumulator and Autonomous Low Power Burst Mode

The ADC0 on C8051F96x devices is a 300 ksps, 10-bit or 75 ksps, 12-bit successive-approximation-register (SAR) ADC with integrated track-and-hold and programmable window detector. ADC0 also has an autonomous low power Burst Mode which can automatically enable ADC0, capture and accumulate samples, then place ADC0 in a low power shutdown mode without CPU intervention. It also has a 16-bit accumulator that can automatically oversample and average the ADC results. See Section 5.4 for more details on using the ADC in 12-bit mode.

The ADC is fully configurable under software control via Special Function Registers. The ADC0 operates in Single-ended mode and may be configured to measure various different signals using the analog multiplexer described in “5.7. ADC0 Analog Multiplexer” on page 95. The voltage reference for the ADC is selected as described in “5.9. Voltage and Ground Reference Options” on page 100.

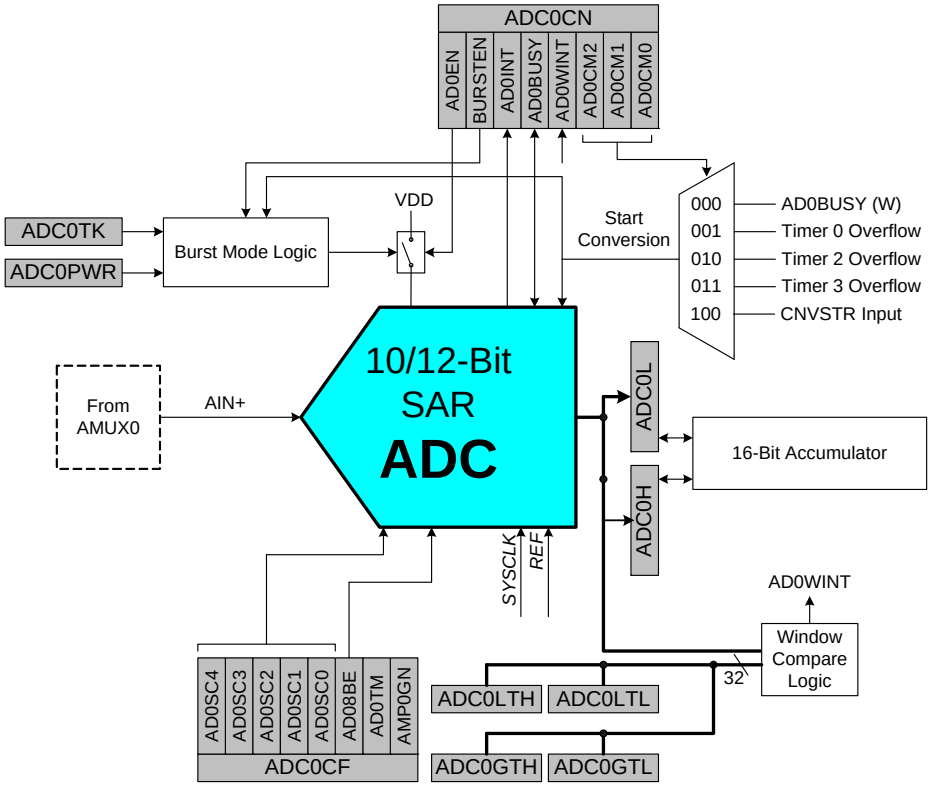

Figure 5.1. ADC0 Functional Block Diagram

5.1. Output Code Formatting

The registers ADC0H and ADC0L contain the high and low bytes of the output conversion code from the ADC at the completion of each conversion. Data can be right-justified or left-justified, depending on the setting of the AD0SJST[2:0]. When the repeat count is set to 1, conversion codes are represented as 10-bit unsigned integers. Inputs are measured from 0 to VREF x 1023/1024. Example codes are shown below for both right-justified and left-justified data. Unused bits in the ADC0H and ADC0L registers are set to 0.

SFR Definition 5.1. ADC0CN: ADC0 Control

Bit	7	6	5	4	3	2	1	0
Name	AD0EN	BURSTEN	AD0INT	AD0BUSY	AD0WINT	ADC0CM[2:0]		
Type	R/W	R/W	R/W	W	R/W	R/W		
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xE8; bit-addressable;

Bit	Name	Function
7	AD0EN	**ADC0 Enable.** 0: ADC0 Disabled (low-power shutdown). 1: ADC0 Enabled (active and ready for data conversions).
6	BURSTEN	**ADC0 Burst Mode Enable.** 0: ADC0 Burst Mode Disabled. 1: ADC0 Burst Mode Enabled.
5	AD0INT	**ADC0 Conversion Complete Interrupt Flag.** Set by hardware upon completion of a data conversion (BURSTEN=0), or a burst of conversions (BURSTEN=1). Can trigger an interrupt. Must be cleared by software.
4	AD0BUSY	**ADC0 Busy.** Writing 1 to this bit initiates an ADC conversion when ADC0CM[2:0] = 000.
3	AD0WINT	**ADC0 Window Compare Interrupt Flag.** Set by hardware when the contents of ADC0H:ADC0L fall within the window specified by ADC0GTH:ADC0GTL and ADC0LTH:ADC0LTL. Can trigger an interrupt. Must be cleared by software.
2:0	ADC0CM[2:0]	**ADC0 Start of Conversion Mode Select.** Specifies the ADC0 start of conversion source. 000: ADC0 conversion initiated on write of 1 to AD0BUSY. 001: ADC0 conversion initiated on overflow of Timer 0. 010: ADC0 conversion initiated on overflow of Timer 2. 011: ADC0 conversion initiated on overflow of Timer 3. 1xx: ADC0 conversion initiated on rising edge of CNVSTR.

SFR Definition 5.4. ADC0PWR: ADC0 Burst Mode Power-Up Time

Bit	7	6	5	4	3	2	1	0
Name	AD0LPM				AD0PWR[3:0]			
Type	R/W	R	R	R	R/W			
Reset	0	0	0	0	1	1	1	1

SFR Page = 0xF; SFR Address = 0xBA

Bit	Name	Function
7	AD0LPM	**ADC0 Low Power Mode Enable.** Enables Low Power Mode Operation. 0: Low Power Mode disabled. 1: Low Power Mode enabled.
6:4	Unused	Read = 0000b; Write = Don't Care.
3:0	AD0PWR[3:0]	**ADC0 Burst Mode Power-Up Time.** Sets the time delay required for ADC0 to power up from a low power state. For BURSTEN = 0: ADC0 power state controlled by AD0EN. For BURSTEN = 1 and AD0EN = 1: ADC0 remains enabled and does not enter a low power state after all conversions are complete. Conversions can begin immediately following the start-of-conversion signal. For BURSTEN = 1 and AD0EN = 0: ADC0 enters a low power state after all conversions are complete. Conversions can begin a programmed delay after the start-of-conversion signal. The ADC0 Burst Mode Power-Up time is programmed according to the following equation: $$AD0PWR = \frac{Tstartup}{400ns} - 1$$ or $$Tstartup = (AD0PWR + 1)400ns$$ **Note:** Setting AD0PWR to 0x04 provides a typical tracking time of 2 us for the first sample taken after the start of conversion.

SFR Definition 6.2. IREF0CF: Current Reference Configuration

Bit	7	6	5	4	3	2	1	0
Name	PWMEN					PWMSS[2:0]		
Type	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0xB9

Bit	Name	Function
7	PWMEN	**PWM Enhanced Mode Enable.** Enables the PWM Enhanced Mode. 0: PWM Enhanced Mode disabled. 1: PWM Enhanced Mode enabled.
6:3	Unused	Read = 0000b, Write = don't care.
2:0	PWMSS[2:0]	**PWM Source Select.** Selects the PCA channel to use for the fine-tuning control signal. 000: CEX0 selected as fine-tuning control signal. 001: CEX1 selected as fine-tuning control signal. 010: CEX2 selected as fine-tuning control signal. 011: CEX3 selected as fine-tuning control signal. 100: CEX4 selected as fine-tuning control signal. 101: CEX5 selected as fine tuning control signal. All Other Values: Reserved.

6.2. IREF0 Specifications

See Table 4.15 on page 73 for a detailed listing of IREF0 specifications.

14.1.3. Configuration sfrs

The AES Module has two configuration sfrs. The AES0BCFG sfr is used to configure the AES core. Bits 0 and 1 are used to select the Key size. The AES core supports 128-bit, 192-bit and 256-bit encryption. Bit 2 selects encrypt or decrypt. The AES enable bit (bit 3) is used to enable the AES module and start and new encryption operation. The AES DONE bit (bit 5) is the AES interrupt flag that signals a block of data has been completely encrypted or decrypted and is ready to be read from the AES0YOUT sfr. Note that the AES DONE interrupt is not normally used when the AES module is used with the DMA. Instead the DMA interrupt is used to signal that the encrypted or decrypted data has been transferred completely to memory. The DMA done interrupt is normally only used with direct sfr access.

The AES0DCFG sfr is used to select the data path for the AES module. Bits 0 through 2 are used to select the input and output multiplexer configuration. The AES data path should be configured prior to initiating a new encryption or decryption operation.

14.1.4. Input Multiplexer

The input multiplexer is used to select either the contents of the AES0BIN sfr or the contents of the AES0BIN sfr exclusive ORed with the contents of the AES0XIN sfr. The exclusive OR input data path provides support for CBC encryption.

14.1.5. Output Multiplexer

The output multiplexer selects the data source for the AES0YOUT sfr. The three possible sources are the AES Core data output, the AES Core Key output, and the AES core data output exclusive ORed with the AES0XIN sfr.

The AES core data output is used for simple encryption and decryption.

The exclusive OR output data path provides support for CBC mode decryption and CTR mode encryption/decryption. The AES0XIN is the source for both input and output exclusive OR data. When the AES0XIN is used with the input exclusive OR data path, the AEXIN data is written in sequence with the AES0BIN data. When used with the output XRO data path, the AES0XIN data is written after the encryption or decryption operation is complete.

The Key output is used to generate an inverse key. To generate a decryption key from an encryption key, the AES core should be configured for an encryption operation. To generate an encryption key from a decryption key, the AES core should be configured for a decryption operation.

14.1.6. Internal State Machine

The AES Module has an internal state machine that manages the data flow. The internal state machine accommodates the two different usage scenarios. When using the DMA, the internal state machine will send peripheral requests to the DMA requesting the DMA to transfer data from xram to the AES module input sfrs. Upon the completion of one block of data, the AES module will send peripheral requests requesting data to be transferred from the AES0YOUT sfr to xram. These peripheral requests are managed by the internal state machine.

When not using the DMA, data must be written and read in a specific order. The DMA state machine will advance with each byte written or read.

The internal state machine may be reset by clearing the enable bit in the AESBGFG sfr. Clearing the enable bit before encryption or decryption operation will ensure that the state machine starts at the proper starting state.

When encrypting or decrypting multiple blocks it is not necessary to disable the AES module between blocks, as long as the proper sequence of events is obeyed.

SFR Definition 15.1. ENC0CN: Encoder Decoder 0 Control

Bit	7	6	5	4	3	2	1	0
Name	READY	ERROR	ENC	DEC		DMA	ENDIAN	MODE
Type	R	R	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC5; SFR page = 0x2; Not bit-Addressable

Bit	Name	Function
7	READY	**Ready Flag.**
6	ERROR	**Error Flag.**
5	ENC	**Encode.** Setting this bit will initiate an Encode operation.
4	DEC	**Decode.** Setting this bit will initiate a Decode operation.
2	DMA	**DMA Mode Enable.** This bit should be set when using the encoder/decoder with the DMA.
1	ENDIAN	**Big-Endian DMA Mode Select.** This bit should be set when using the DMA with big-endian multiple byte DMA transfers. The DMA must also be configured for the same endian mode.
0	MODE	**Mode.** 0: Select Manchester encoding or decoding. 1:Select Three-out-of-Six encoding or decoding.

SFR Definition 19.7. PCON: Power Management Control Register

Bit	7	6	5	4	3	2	1	0
Name	GF[4:0]					PWRSEL	STOP	IDLE
Type	R/W					R/W	W	W
Reset	0	0	0	0	0	0	0	0

SFR Page = All Pages; SFR Address = 0x87

Bit	Name	Description	Write	Read
7:3	GF[5:0]	**General Purpose Flags**	Sets the logic value.	Returns the logic value.
2	PWRSEL	**Power Select**	0: VBAT is selected as the input to VREG0. 1: VDC is selected as the input to VREG0.	
1	STOP	**Stop Mode Select**	Writing 1 places the device in Stop Mode.	N/A
0	IDLE	**Idle Mode Select**	Writing 1 places the device in Idle Mode.	N/A

19.9. Power Management Specifications

See Table 4.7 on page 69 for detailed Power Management Specifications.

22.1. Power-On Reset

During power-up, the device is held in a reset state and the $\overline{\text{RST}}$ pin voltage tracks the supply voltage (through a weak pull-up) until the device is released from reset. After the supply settles above VPOR, a delay occurs before the device is released from reset; the delay decreases as the supply ramp time increases (ramp time is defined as how fast the supply ramps from 0 V to V_{POR}). Figure 22.2 plots the power-on and supply monitor reset timing. For valid ramp times (less than 3 ms), the power-on reset delay ($T_{PORDelay}$) is typically 7 ms (V_{DD} = 1.8 V) or 15 ms (V_{DD} = 3.6 V).

Note: The maximum supply ramp time is 3 ms; slower ramp times may cause the device to be released from reset before the supply reaches the V_{POR} level.

On exit from a power-on reset, the PORSF flag (RSTSRC.1) is set by hardware to logic 1. When PORSF is set, all of the other reset flags in the RSTSRC Register are indeterminate (PORSF is cleared by all other resets). Since all resets cause program execution to begin at the same location (0x0000), software can read the PORSF flag to determine if a power-up was the cause of reset. The contents of internal data memory should be assumed to be undefined after a power-on reset.

The POR supply monitor will continue to monitor the VBAT supply, even in Sleep Mode, to reset the system if the supply voltage drops below VPOR. It can be disabled to save power by writing 1 to the MONDIS (PMU0MD.5) bit. When the POR supply monitor is disabled, all reset sources will trigger a full POR and will re-enable the POR supply monitor.

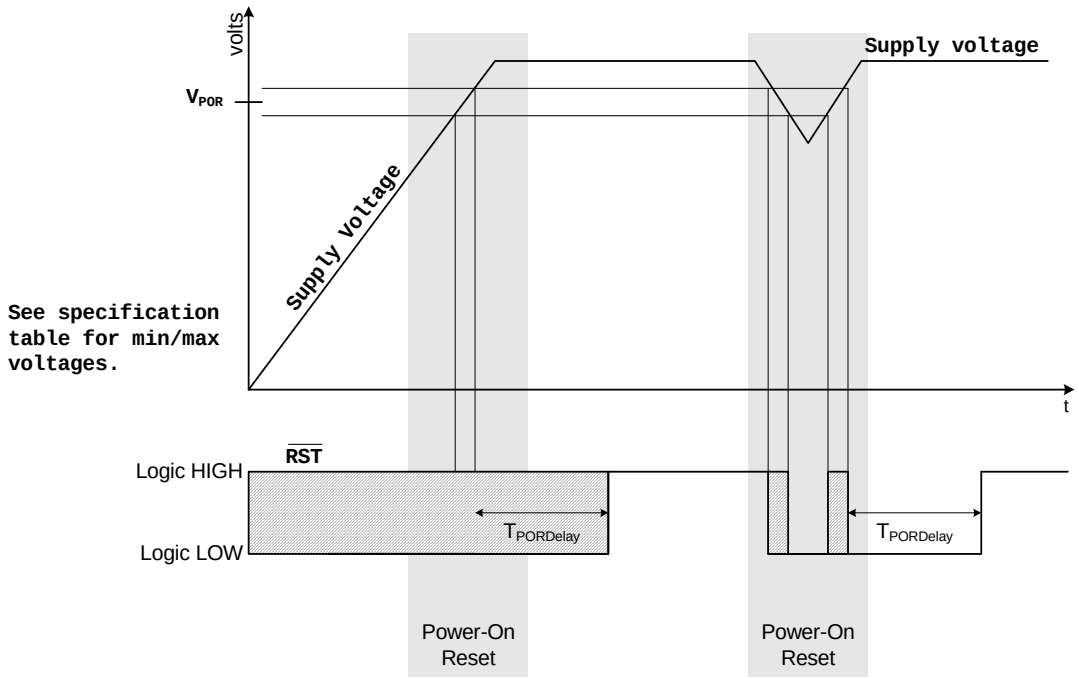

Figure 22.2. Power-On Reset Timing Diagram

Internal Register Definition 24.4. RTC0CN: SmaRTClock Control

Bit	7	6	5	4	3	2	1	0
Name	RTC0EN	MCLKEN	OSCFAIL	RTC0TR		HSMODE	RTC0SET	RTC0CAP
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	Varies	0	0	0	0	0

SmaRTClock Address = 0x04

Bit	Name	Function
7	RTC0EN	**SmaRTClock Enable.** Enables/disables the SmaRTClock oscillator and associated bias currents. 0: SmaRTClock oscillator disabled. 1: SmaRTClock oscillator enabled.
6	MCLKEN	**Missing SmaRTClock Detector Enable.** Enables/disables the missing SmaRTClock detector. 0: Missing SmaRTClock detector disabled. 1: Missing SmaRTClock detector enabled.
5	OSCFAIL	**SmaRTClock Oscillator Fail Event Flag.** Set by hardware when a missing SmaRTClock detector timeout occurs. Must be cleared by software. The value of this bit is not defined when the SmaRTClock oscillator is disabled.
4	RTC0TR	**SmaRTClock Timer Run Control.** Controls if the SmaRTClock timer is running or stopped (holds current value). 0: SmaRTClock timer is stopped. 1: SmaRTClock timer is running.
3	Reserved	Read = 0b; Must write 0b.
2	HSMODE	**High Speed Mode Enable.** Should be set to 1 if the system clock is faster than 4x the SmaRTClock frequency. 0: High Speed Mode is disabled. 1: High Speed Mode is enabled.
1	RTC0SET	**SmaRTClock Timer Set.** Writing 1 initiates a SmaRTClock timer set operation. This bit is cleared to 0 by hardware to indicate that the timer set operation is complete.
0	RTC0CAP	**SmaRTClock Timer Capture.** Writing 1 initiates a SmaRTClock timer capture operation. This bit is cleared to 0 by hardware to indicate that the timer capture operation is complete.

Internal Register Definition 24.10. ALARM1Bn: SmaRTClock Alarm 1 Match Value

Bit	7	6	5	4	3	2	1	0
Name	ALARM1[31:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SmaRTClock Address: ALARM1B0 = 0x0C; ALARM1B1 = 0x0D; ALARM1B2 = 0x0E; ALARM1B3 = 0x0F

Bit	Name	Function
7:0	ALARM1[31:0]	**SmaRTClock Alarm 1 Programmed Value.** These 4 registers (ALARM1B3–ALARM1B0) are used to set an alarm event for the SmaRTClock timer. The SmaRTClock alarm should be disabled (ALRM1EN=0) when updating these registers.
Note: The least significant bit of the alarm programmed value is iALARM1B0.0.		

Internal Register Definition 24.11. ALARM2Bn: SmaRTClock Alarm 2 Match Value

Bit	7	6	5	4	3	2	1	0
Name	ALARM2[31:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SmaRTClock Address: ALARM2B0 = 0x10; ALARM2B1 = 0x11; ALARM2B2 = 0x12; ALARM2B3 = 0x13

Bit	Name	Function
7:0	ALARM2[31:0]	**SmaRTClock Alarm 2 Programmed Value.** These 4 registers (ALARM2B3–ALARM2B0) are used to set an alarm event for the SmaRTClock timer. The SmaRTClock alarm should be disabled (ALRM2EN=0) when updating these registers.
Note: The least significant bit of the alarm programmed value is ALARM2B0.0.		

SFR Definition 26.3. LCD0CNTRST: LCD0 Contrast Adjustment

Bit	7	6	5	4	3	2	1	0
Name	Reserved	Reserved	Reserved	CNTRST				
Type	R/W	R/W	R/W	R/W				
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x2; SFR Address = 0x9C

Bit	Name	Function
7:5	Reserved	Read = 000. Write = Must write 000.
4:0	CNTRST	**Contrast Setpoint.** Determines the setpoint for the VLCD voltage necessary to achieve the desired contrast. 00000: 1.90 00001: 1.96 00010: 2.02 00011: 2.08 00100: 2.13 00101: 2.19 00110: 2.25 00111: 2.31 01000: 2.37 01001: 2.43 01010: 2.49 01011: 2.55 01100: 2.60 01101: 2.66 01110: 2.72 01111: 2.78 10000: 2.84 10001: 2.90 10010: 2.96 10011: 3.02 10100: 3.07 10101: 3.13 10110: 3.19 10111: 3.25 11000: 3.31 11001: 3.37 11010: 3.43 11011: 3.49 11100: 3.54 11101: 3.60 11110: 3.66 11111: 3.72

SFR Definition 27.34. P5DRV: Port5 Drive Strength

Bit	7	6	5	4	3	2	1	0
Name	P5DRV[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0xA3

Bit	Name	Function
7:0	P5DRV[7:0]	**Drive Strength Configuration Bits for P5.7–P5.0 (respectively).** Configures digital I/O Port cells to high or low output drive strength. 0: Corresponding P5.n Output has low output drive strength. 1: Corresponding P5.n Output has high output drive strength.

SFR Definition 27.35. P6: Port6

Bit	7	6	5	4	3	2	1	0
Name	P6[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Page = 0xF; SFR Address = 0xDB

Bit	Name	Description	Write	Read
7:0	P6[7:0]	**Port 6 Data.** Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P6.n Port pin is logic LOW. 1: P6.n Port pin is logic HIGH.

Table 28.1. SMBus Clock Source Selection

SMBCS1	SMBCS0	SMBus Clock Source
0	0	Timer 0 Overflow
0	1	Timer 1 Overflow
1	0	Timer 2 High Byte Overflow
1	1	Timer 2 Low Byte Overflow

The SMBCS1–0 bits select the SMBus clock source, which is used only when operating as a master or when the Free Timeout detection is enabled. When operating as a master, overflows from the selected source determine the absolute minimum SCL low and high times as defined in Equation 28.1. The selected clock source may be shared by other peripherals so long as the timer is left running at all times. For example, Timer 1 overflows may generate the SMBus and UART baud rates simultaneously. Timer configuration is covered in Section “32. Timers” on page 444.

$$T_{HighMin} = T_{LowMin} = \frac{1}{f_{ClockSourceOverflow}}$$

Equation 28.1. Minimum SCL High and Low Times

The selected clock source should be configured to establish the minimum SCL High and Low times as per Equation 28.1. When the interface is operating as a master (and SCL is not driven or extended by any other devices on the bus), the typical SMBus bit rate is approximated by Equation 28.1.

$$BitRate = \frac{f_{ClockSourceOverflow}}{3}$$

Equation 28.2. Typical SMBus Bit Rate

Figure 28.4 shows the typical SCL generation described by Equation 28.2. Notice that T_{HIGH} is typically twice as large as T_{LOW}. The actual SCL output may vary due to other devices on the bus (SCL may be extended low by slower slave devices, or driven low by contending master devices). The bit rate when operating as a master will never exceed the limits defined by Equation 28.2.

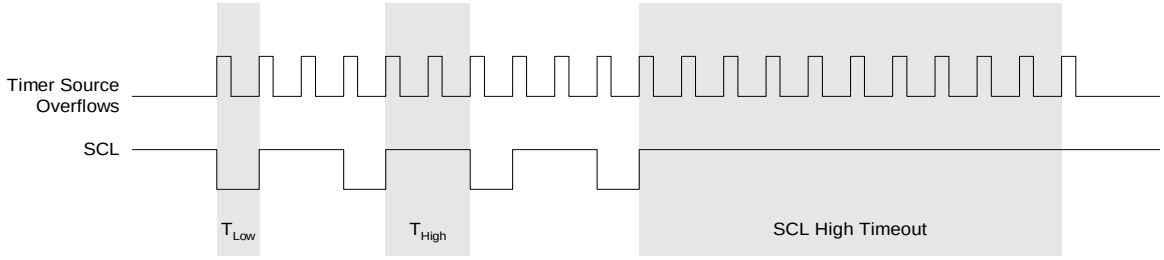

Figure 28.4. Typical SMBus SCL Generation

Setting the EXTHOLD bit extends the minimum setup and hold times for the SDA line. The minimum SDA setup time defines the absolute minimum time that SDA is stable before SCL transitions from low-to-high. The minimum SDA hold time defines the absolute minimum time that the current SDA value remains stable after SCL transitions from high-to-low. EXTHOLD should be set so that the minimum setup and hold times meet the SMBus Specification requirements of 250 ns and 300 ns, respectively. Table 28.2 shows the minimum setup and hold times for the two EXTHOLD settings. Setup and hold time extensions are typically necessary when SYSCLK is above 10 MHz.

Figure 28.5 shows a typical master write sequence. Two transmit data bytes are shown, though any number of bytes may be transmitted. All “data byte transferred” interrupts occur **after** the ACK cycle in this mode, regardless of whether hardware ACK generation is enabled.

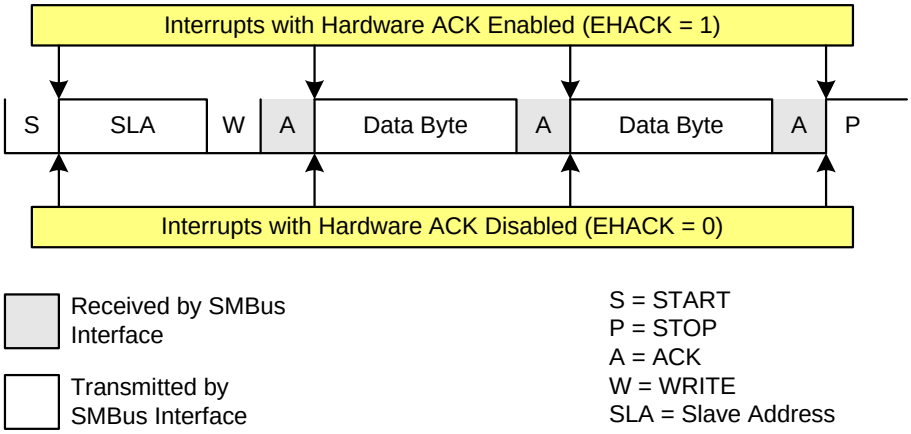

Figure 28.5. Typical Master Write Sequence

28.5.2. Read Sequence (Master)

During a read sequence, an SMBus master reads data from a slave device. The master in this transfer will be a transmitter during the address byte, and a receiver during all data bytes. The SMBus interface generates the START condition and transmits the first byte containing the address of the target slave and the data direction bit. In this case the data direction bit (R/W) will be logic 1 (READ). Serial data is then received from the slave on SDA while the SMBus outputs the serial clock. The slave transmits one or more bytes of serial data.

If hardware ACK generation is disabled, the ACKRQ is set to 1 and an interrupt is generated after each received byte. Software must write the ACK bit at that time to ACK or NACK the received byte.

With hardware ACK generation enabled, the SMBus hardware will automatically generate the ACK/NACK, and then post the interrupt. **It is important to note that the appropriate ACK or NACK value should be set up by the software prior to receiving the byte when hardware ACK generation is enabled.**

Writing a 1 to the ACK bit generates an ACK; writing a 0 generates a NACK. Software should write a 0 to the ACK bit for the last data transfer, to transmit a NACK. The interface exits Master Receiver Mode after the STO bit is set and a STOP is generated. The interface will switch to Master Transmitter Mode if SMB0DAT is written while an active Master Receiver. Figure 28.6 shows a typical master read sequence. Two received data bytes are shown, though any number of bytes may be received. The “data byte transferred” interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.

To initiate a Master mode Bidirectional data transfer:

1. Configure the SPI1 SFRs normally for Master mode.
   a. Enable Master mode by setting bit 6 in SPI1CFG.
   b. Configure the clock polarity CKPOL and clock phase CKPHA as desired in SPI1CFG.
   c. Configure SPI1CKR for the desired SPI clock rate.
   d. Configure the desired 4-wire master or 3-wire master mode in SPI1CN.
   e. Enable the SPI by setting bit 0 of SPI1CN.
2. Configure the first DMA channel for the XRAM-to-SPI1DATA transfer:
   a. Disable the first DMA channel by clearing the corresponding bit in DMA0EN.
   b. Select the first DMA channel by writing to DMA0SEL.
   c. Configure the selected DMA channel to use the XRAM-to-SPI1DAT peripheral request by writing 0x03 to DMA0NCF.
   d. Write 0 to DMA0NMD to disable wrapping.
   e. Write the address of the first byte of master output (MOSI) data to DMA0NBAH:L.
   f. Write the size of the SPI transfer in bytes to DMA0NSZH:L.
   g. Clear the address offset SFRs CMA0A0H:L.
3. Configure the second DMA channel for the SPI1DAT-to-XRAM transfer:
   a. Disable the second DMA channel by clearing the corresponding bit in DMA0EN.
   b. Select the second DMA channel by writing to DMA0SEL.
   c. Configure the selected DMA channel to use the SPI1DAT-to-XRAM peripheral request by writing 0x04 to DMA0NCF.
   d. Enable DMA interrupts for the second channel by setting bit 7 of DMA0NCF.
   e. Write 0 to DMA0NMD to disable wrapping.
   f. Write the address for the first byte of master input (MISO) data to DMA0NBAH:L.
   g. Write the size of the SPI transfer in bytes to DMA0NSZH:L.
   h. Clear the address offset SFRs CMA0A0H:L.
   i. Enable the interrupt on the second channel by setting the corresponding bit in DMA0INT.
   j. Enable DMA interrupts by setting bit 5 of EIE2.
4. Clear the interrupt bits in DMA0INT for both channels.
5. Enable both channels by setting the corresponding bits in the DMA0EN SFR to initiate the SPI transfer operation.
6. Wait on the DMA interrupt.
7. Clear the DMA enables in the DMA0EN SFR.
8. Clear the DMA interrupts in the DMA0INT SFR.

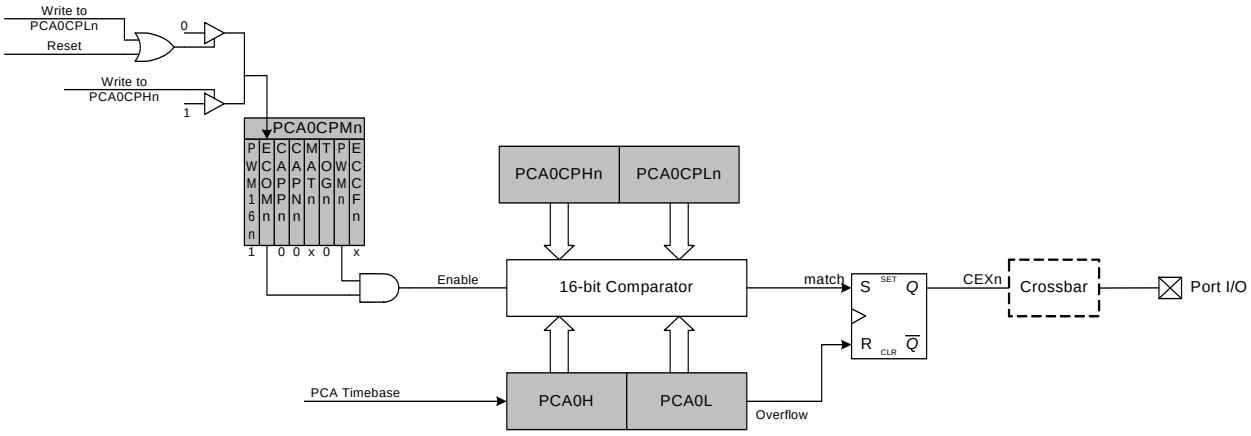

Figure 33.10. PCA 16-Bit PWM Mode

33.4. Watchdog Timer Mode

A programmable watchdog timer (WDT) function is available through the PCA Module 5. The WDT is used to generate a reset if the time between writes to the WDT update register (PCA0CPH2) exceed a specified limit. The WDT can be configured and enabled/disabled as needed by software.

With the WDTE bit set in the PCA0MD register, Module 5 operates as a watchdog timer (WDT). The Module 5 high byte is compared to the PCA counter high byte; the Module 5 low byte holds the offset to be used when WDT updates are performed. **The Watchdog Timer is enabled on reset. Writes to some PCA registers are restricted while the Watchdog Timer is enabled.** The WDT will generate a reset shortly after code begins execution. To avoid this reset, the WDT should be explicitly disabled (and optionally re-configured and re-enabled if it is used in the system).

33.4.1. Watchdog Timer Operation

While the WDT is enabled:

- PCA counter is forced on.
- Writes to PCA0L and PCA0H are not allowed.
- PCA clock source bits (CPS2–CPS0) are frozen.
- PCA Idle control bit (CIDL) is frozen.
- Module 5 is forced into software timer mode.
- Writes to the Module 5 mode register (PCA0CPM5) are disabled.

While the WDT is enabled, writes to the CR bit will not change the PCA counter state; the counter will run until the WDT is disabled. The PCA counter run control bit (CR) will read zero if the WDT is enabled but user software has not enabled the PCA counter. If a match occurs between PCA0CPH5 and PCA0H while the WDT is enabled, a reset will be generated. To prevent a WDT reset, the WDT may be updated with a write of any value to PCA0CPH5. Upon a PCA0CPH5 write, PCA0H plus the offset held in PCA0CPL5 is loaded into PCA0CPH5. (See Figure 33.11.)