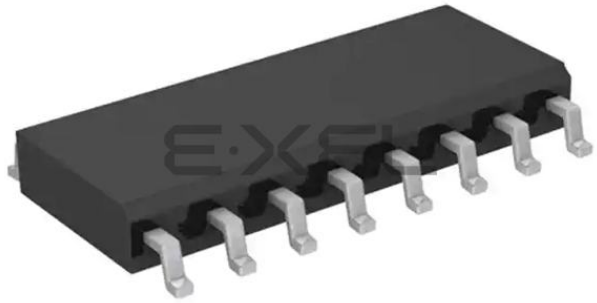

Welcome to E-XFL.COM

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	Active
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	13
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.2V ~ 3.6V
Data Converters	A/D 12x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SOIC (0.154", 3.90mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f862-c-gs

Table 1.11. Port I/O

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Output High Voltage (High Drive)	V_{OH}	I_{OH} = –3 mA	V_{DD} – 0.7	—	—	V
Output Low Voltage (High Drive)	V_{OL}	I_{OL} = 8.5 mA	—	—	0.6	V
Output High Voltage (Low Drive)	V_{OH}	I_{OH} = –1 mA	V_{DD} – 0.7	—	—	V
Output Low Voltage (Low Drive)	V_{OL}	I_{OL} = 1.4 mA	—	—	0.6	V
Input High Voltage	V_{IH}		V_{DD} – 0.6	—	—	V
Input Low Voltage	V_{IL}		—	—	0.6	V
Pin Capacitance	C_{IO}		—	7	—	pF
Weak Pull-Up Current (V_{IN} = 0 V)	I_{PU}	V_{DD} = 3.6	–30	–20	–10	µA
Input Leakage (Pullups off or Analog)	I_{LK}	GND ≤ V_{IN} ≤ V_{DD}	–1.1	—	1.1	µA
Input Leakage Current with V_{IN} above V_{DD}	I_{LK}	V_{DD} < V_{IN} < V_{DD}+2.0 V	0	5	150	µA

Table 3.2. Pin Definitions for C8051F850/1/2/3/4/5-GM and C8051F850/1/2/3/4/5-IM

Pin Name	Type	Pin Numbers	Crossbar Capability	Additional Digital Functions	Analog Functions
P0.0	Standard I/O	2	Yes	P0MAT.0 INT0.0 INT1.0	ADC0.0 CP0P.0 CP0N.0 VREF
P0.1	Standard I/O	1	Yes	P0MAT.1 INT0.1 INT1.1	ADC0.1 CP0P.1 CP0N.1 AGND
P0.2	Standard I/O	20	Yes	P0MAT.2 INT0.2 INT1.2	ADC0.2 CP0P.2 CP0N.2
P0.3	Standard I/O	19	Yes	P0MAT.3 EXTCLK INT0.3 INT1.3	ADC0.3 CP0P.3 CP0N.3
P0.4	Standard I/O	18	Yes	P0MAT.4 INT0.4 INT1.4	ADC0.4 CP0P.4 CP0N.4
P0.5	Standard I/O	17	Yes	P0MAT.5 INT0.5 INT1.5	ADC0.5 CP0P.5 CP0N.5
P0.6	Standard I/O	16	Yes	P0MAT.6 CNVSTR INT0.6 INT1.6	ADC0.6 CP0P.6 CP0N.6
P0.7	Standard I/O	15	Yes	P0MAT.7 INT0.7 INT1.7	ADC0.7 CP0P.7 CP0N.7

Table 3.3. Pin Definitions for C8051F860/1/2/3/4/5-GS and C8051F860/1/2/3/4/5-IS

Pin Name	Type	Pin Numbers	Crossbar Capability	Additional Digital Functions	Analog Functions
P1.1	Standard I/O	10	Yes	P1MAT.1	ADC0.9 CP1P.3 CP1N.3
P1.2	Standard I/O	9	Yes	P1MAT.2	ADC0.10 CP1P.4 CP1N.4
P1.3	Standard I/O	8	Yes	P1MAT.3	ADC0.11 CP1P.5 CP1N.5
P2.0 / C2D	Standard I/O / C2 Debug Data	7			

Table 12.1. Interrupt Summary

Interrupt Source	Interrupt Vector	Priority Order	Pending Flags	Bit addressable?	Cleared by HW?	Enable Flag
Reset	0x0000	Top	None	N/A	N/A	Always Enabled
External Interrupt 0 ($\overline{INT0}$)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)
External Interrupt 1 ($\overline{INT1}$)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)
UART0	0x0023	4	RI (SCON0.0) TI (SCON0.1)	Y	N	ES0 (IE.4)
Timer 2 Overflow	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	Y	N	ET2 (IE.5)
SPI0	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN (SPI0CN.4)	Y	N	ESPI0 (IE.6)
SMB0	0x003B	7	SI (SMB0CN.0)	Y	N	ESMB0 (EIE1.0)
Port Match	0x0043	8	None	N/A	N/A	EMAT (EIE1.1)
ADC0 Window Compare	0x004B	9	ADWINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.2)
ADC0 Conversion Complete	0x0053	10	ADINT (ADC0CN.5)	Y	N	EADC0 (EIE1.3)
Programmable Counter Array	0x005B	11	CF (PCA0CN.7) CCFn (PCA0CN.n) COVF (PCA0PWM.6)	Y	N	EPCA0 (EIE1.4)
Comparator0	0x0063	12	CPFIF (CPT0CN.4) CPRIF (CPT0CN.5)	N	N	ECP0 (EIE1.5)
Comparator1	0x006B	13	CPFIF (CPT1CN.4) CPRIF (CPT1CN.5)	N	N	ECP1 (EIE1.6)
Timer 3 Overflow	0x0073	14	TF3H (TMR3CN.7) TF3L (TMR3CN.6)	N	N	ET3 (EIE1.7)

Register 14.2. ADC0CN1: ADC0 Control 1

Bit	7	6	5	4	3	2	1	0
Name	Reserved							ADCMBE
Type	R							RW
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xB2								

Table 14.5. ADC0CN1 Register Bit Descriptions

Bit	Name	Function
7:1	Reserved	Must write reset value.
0	ADCMBE	**Common Mode Buffer Enable.** 0: Disable the common mode buffer. This setting should be used only if the tracking time of the signal is greater than 1.5 us. 1: Enable the common mode buffer. This setting should be used in most cases, and will give the best dynamic ADC performance. The common mode buffer must be enabled if signal tracking time is less than or equal to 1.5 us.

Register 14.9. ADC0GTH: ADC0 Greater-Than High Byte

Bit	7	6	5	4	3	2	1	0
Name	ADC0GTH							
Type	RW							
Reset	1	1	1	1	1	1	1	1
SFR Address: 0xC4								

Table 14.12. ADC0GTH Register Bit Descriptions

Bit	Name	Function
7:0	ADC0GTH	**Greater-Than High Byte.** Most Significant Byte of the 16-bit Greater-Than window compare register.

17. Comparators (CMP0 and CMP1)

C8051F85x/86x devices include two on-chip programmable voltage comparators, CMP0 and CMP1. The two comparators are functionally identical, but have different connectivity within the device. A functional block diagram is shown in Figure 17.1.

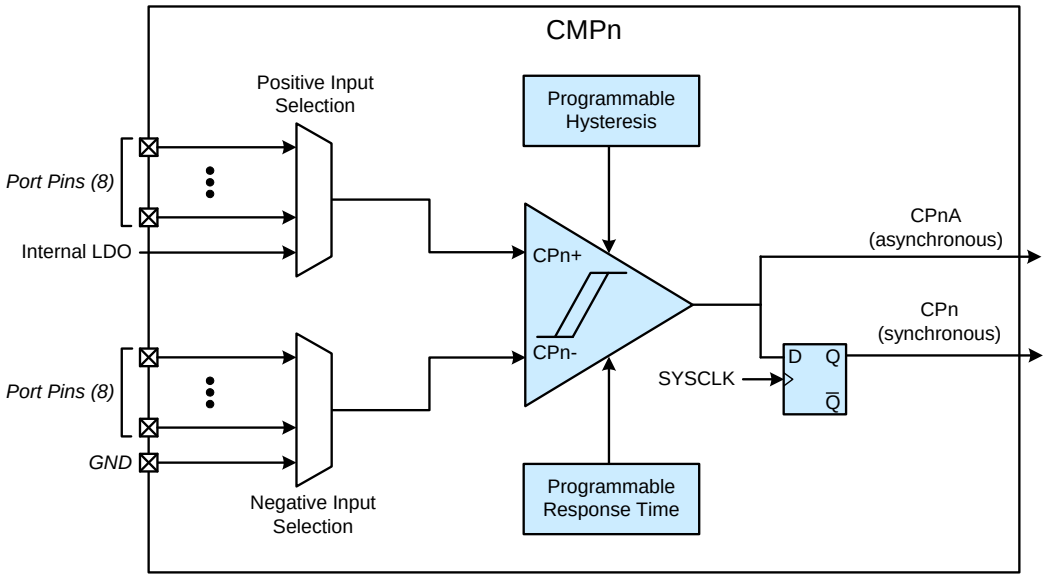

Figure 17.1. Comparator Functional Block Diagram

17.1. System Connectivity

Comparator inputs are routed to port I/O pins or internal signals using the comparator mux registers. The comparator's synchronous and asynchronous outputs can optionally be routed to port I/O pins through the port I/O crossbar. The output of either comparator may also be configured to generate a system interrupt. CMP0 may also be used as a reset source, or as a trigger to kill a PCA output channel.

The CMP0 inputs are selected in the CPT0MX register, while CPT1MX selects the CMP1 inputs. The CMXP field selects the comparator's positive input (CPnP.x); the CMXN field selects the comparator's negative input (CPnN.x). Table 17.1 through Table 17.4 detail the comparator input multiplexer options on the C8051F85x/86x family. See the port I/O crossbar sections for details on configuring comparator outputs via the digital crossbar. Comparator inputs can be externally driven from –0.25 V to (V_{DD}) + 0.25 V without damage or upset.

Important Note About Comparator Inputs: The port pins selected as comparator inputs should be configured as analog inputs in their associated port configuration register, and configured to be skipped by the crossbar.

Register 17.4. CPT1CN: Comparator 1 Control

Bit	7	6	5	4	3	2	1	0
Name	CPEN	CPOUT	CPRIF	CPFIF	CPHYP		CPHYN	
Type	RW	R	RW	RW	RW		RW	
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xBF								

Table 17.8. CPT1CN Register Bit Descriptions

Bit	Name	Function
7	CPEN	**Comparator 1 Enable Bit.** 0: Comparator Disabled. 1: Comparator Enabled.
6	CPOUT	**Comparator 1 Output State Flag.** 0: Voltage on CP1P < CP1N. 1: Voltage on CP1P > CP1N.
5	CPRIF	**Comparator 1 Rising-Edge Flag. Must be cleared by software.** 0: No Comparator Rising Edge has occurred since this flag was last cleared. 1: Comparator Rising Edge has occurred.
4	CPFIF	**Comparator 1 Falling-Edge Flag. Must be cleared by software.** 0: No Comparator Falling Edge has occurred since this flag was last cleared. 1: Comparator Falling Edge has occurred.
3:2	CPHYP	**Comparator 1 Positive Hysteresis Control Bits.** 00: Positive Hysteresis Disabled. 01: Positive Hysteresis = 5 mV. 10: Positive Hysteresis = 10 mV. 11: Positive Hysteresis = 20 mV.
1:0	CPHYN	**Comparator 1 Negative Hysteresis Control Bits.** 00: Negative Hysteresis Disabled. 01: Negative Hysteresis = 5 mV. 10: Negative Hysteresis = 10 mV. 11: Negative Hysteresis = 20 mV.

18.6. CRC Control Registers

Register 18.1. CRC0CN: CRC0 Control

Bit	7	6	5	4	3	2	1	0
Name	Reserved				CRCINIT	CRCVAL	Reserved	CRCPNT
Type	R				RW	RW	R	RW
Reset	0	0	0	1	0	0	0	0
SFR Address: 0xCE								

Table 18.2. CRC0CN Register Bit Descriptions

Bit	Name	Function
7:4	Reserved	Must write reset value.
3	CRCINIT	**CRC Result Initialization Bit.** Writing a 1 to this bit initializes the entire CRC result based on CRCVAL.
2	CRCVAL	**CRC Set Value Initialization Bit.** This bit selects the set value of the CRC result. 0: CRC result is set to 0x0000 on write of 1 to CRCINIT. 1: CRC result is set to 0xFFFF on write of 1 to CRCINIT.
1	Reserved	Must write reset value.
0	CRCPNT	**CRC Result Pointer.** Specifies the byte of the CRC result to be read/written on the next access to CRC0DAT. This bit will automatically toggle upon each read or write. 0: CRC0DAT accesses bits 7-0 of the 16-bit CRC result. 1: CRC0DAT accesses bits 15-8 of the 16-bit CRC result.
Note: Upon initiation of an automatic CRC calculation, the three cycles following a write to CRC0CN that initiate a CRC operation must only contain instructions which execute in the same number of cycles as the number of bytes in the instruction. An example of such an instruction is a 3-byte MOV that targets the CRC0FLIP register. When programming in C, the dummy value written to CRC0FLIP should be a non-zero value to prevent the compiler from generating a 2-byte MOV instruction.		

20.4.2. Center Aligned PWM

When configured for center-aligned mode, a module will generate an edge transition at two points for every $2^{(N+1)}$ PCA clock cycles, where N is the selected PWM resolution in bits. In center-aligned mode, these two edges are referred to as the “up” and “down” edges. The polarity at the output pin is selectable, and can be inverted by setting the appropriate channel bit to ‘1’ in the PCA0POL register.

The generated waveforms are centered about the points where the lower N bits of the PCA0 counter are zero. The $(N+1)^{th}$ bit in the PCA0 counter acts as a selection between up and down edges. In 16-bit mode, a special 17th bit is implemented internally for this purpose. At the center point, the (non-inverted) channel output will be low when the $(N+1)^{th}$ bit is ‘0’ and high when the $(N+1)^{th}$ bit is ‘1’, except for cases of 0% and 100% duty cycle. Prior to inversion, an up edge sets the channel to logic high, and a down edge clears the channel to logic low.

Down edges occur when the $(N+1)^{th}$ bit in the PCA0 counter is one, and a logical inversion of the value in the module’s PCA0CPn register matches the main PCA0 counter register for the lowest N bits. For example, with 10-bit PWM, the down edge will occur when the one’s complement of bits 9-0 of the PCA0CPn register match bits 9-0 of the PCA0 counter, and bit 10 of the PCA0 counter is ‘1’.

Up edges occur when the $(N+1)^{th}$ bit in the PCA0 counter is zero, and the lowest N bits of the module’s PCA0CPn register match the value of (PCA0 - 1). For example, with 10-bit PWM, the up edge will occur when bits 9-0 of the PCA0CPn register are one less than bits 9-0 of the PCA0 counter, and bit 10 of the PCA0 counter is ‘0’.

An example of the PWM timing in center-aligned mode for two channels is shown in Figure 20.7. In this example, the CEX0POL and CEX1POL bits are cleared to 0.

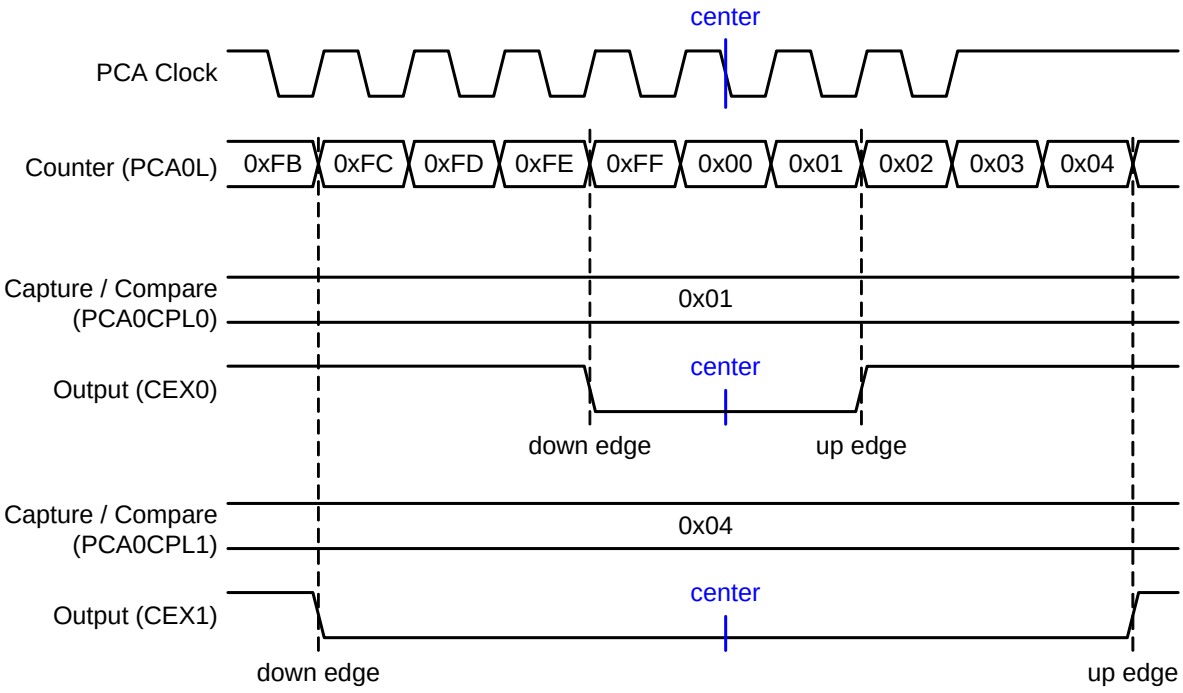

Figure 20.7. Center-Aligned PWM Timing

Register 21.6. P0MAT: Port 0 Match

Bit	7	6	5	4	3	2	1	0
Name	P0MAT							
Type	RW							
Reset	1	1	1	1	1	1	1	1
SFR Address: 0xFD								

Table 21.9. P0MAT Register Bit Descriptions

Bit	Name	Function
7:0	P0MAT	**Port 0 Match Value.** Match comparison value used on P0 pins for bits in P0MASK which are set to 1. 0: P0.x pin logic value is compared with logic LOW. 1: P0.x pin logic value is compared with logic HIGH.

23.1. Signal Descriptions

The four signals used by SPI0 (MOSI, MISO, SCK, NSS) are described below.

23.1.1. Master Out, Slave In (MOSI)

The master-out, slave-in (MOSI) signal is an output from a master device and an input to slave devices. It is used to serially transfer data from the master to the slave. This signal is an output when SPI0 is operating as a master and an input when SPI0 is operating as a slave. Data is transferred most-significant bit first. When configured as a master, MOSI is driven by the MSB of the shift register in both 3- and 4-wire mode.

23.1.2. Master In, Slave Out (MISO)

The master-in, slave-out (MISO) signal is an output from a slave device and an input to the master device. It is used to serially transfer data from the slave to the master. This signal is an input when SPI0 is operating as a master and an output when SPI0 is operating as a slave. Data is transferred most-significant bit first. The MISO pin is placed in a high-impedance state when the SPI module is disabled and when the SPI operates in 4-wire mode as a slave that is not selected. When acting as a slave in 3-wire mode, MISO is always driven by the MSB of the shift register.

23.1.3. Serial Clock (SCK)

The serial clock (SCK) signal is an output from the master device and an input to slave devices. It is used to synchronize the transfer of data between the master and slave on the MOSI and MISO lines. SPI0 generates this signal when operating as a master. The SCK signal is ignored by a SPI slave when the slave is not selected (NSS = 1) in 4-wire slave mode.

23.1.4. Slave Select (NSS)

The function of the slave-select (NSS) signal is dependent on the setting of the NSSMD1 and NSSMD0 bits in the SPI0CN register. There are three possible modes that can be selected with these bits:

1. NSSMD[1:0] = 00: 3-Wire Master or 3-Wire Slave Mode: SPI0 operates in 3-wire mode, and NSS is disabled. When operating as a slave device, SPI0 is always selected in 3-wire mode. Since no select signal is present, SPI0 must be the only slave on the bus in 3-wire mode. This is intended for point-to-point communication between a master and one slave.
2. NSSMD[1:0] = 01: 4-Wire Slave or Multi-Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an input. When operating as a slave, NSS selects the SPI0 device. When operating as a master, a 1-to-0 transition of the NSS signal disables the master function of SPI0 so that multiple master devices can be used on the same SPI bus.
3. NSSMD[1:0] = 1x: 4-Wire Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an output. The setting of NSSMD0 determines what logic level the NSS pin will output. This configuration should only be used when operating SPI0 as a master device.

See Figure 23.2, Figure 23.3, and Figure 23.4 for typical connection diagrams of the various operational modes. **Note that the setting of NSSMD bits affects the pinout of the device.** When in 3-wire master or 3-wire slave mode, the NSS pin will not be mapped by the crossbar. In all other modes, the NSS signal will be mapped to a pin on the device.

Register 23.4. SPI0DAT: SPI0 Data

Bit	7	6	5	4	3	2	1	0
Name	SPI0DAT							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xA3								

Table 23.5. SPI0DAT Register Bit Descriptions

Bit	Name	Function
7:0	SPI0DAT	**SPI0 Transmit and Receive Data.** The SPI0DAT register is used to transmit and receive SPI0 data. Writing data to SPI0-DAT places the data into the transmit buffer and initiates a transfer when in master mode. A read of SPI0DAT returns the contents of the receive buffer.

minimum setup and hold times for the two EXTHOLD settings. Setup and hold time extensions are typically necessary for SMBus compliance when SYSCLK is above 10 MHz.

Table 24.2. Minimum SDA Setup and Hold Times

EXTHOLD	Minimum SDA Setup Time	Minimum SDA Hold Time
0	T_{low} – 4 system clocks or 1 system clock + s/w delay*	3 system clocks
1	11 system clocks	12 system clocks

Note: Setup Time for ACK bit transmissions and the MSB of all data transfers. When using software acknowledgment, the s/w delay occurs between the time SMB0DAT or ACK is written and when SI0 is cleared. Note that if SI is cleared in the same write that defines the outgoing ACK value, s/w delay is zero.

With the SMBTOE bit set, Timer 3 should be configured to overflow after 25 ms in order to detect SCL low timeouts (see Section “24.3.4. SCL Low Timeout” on page 235). The SMBus interface will force the associated timer to reload while SCL is high, and allow the timer to count when SCL is low. The timer interrupt service routine should be used to reset SMBus communication by disabling and re-enabling the SMBus.

SMBus Free Timeout detection can be enabled by setting the SMBFTE bit. When this bit is set, the bus will be considered free if SDA and SCL remain high for more than 10 SMBus clock source periods (see Figure 24.4).

24.4.2. SMBus Pin Swap

The SMBus peripheral is assigned to pins using the priority crossbar decoder. By default, the SMBus signals are assigned to port pins starting with SDA on the lower-numbered pin, and SCL on the next available pin. The SWAP bit in the SMBTC register can be set to 1 to reverse the order in which the SMBus signals are assigned.

24.4.3. SMBus Timing Control

The SDD field in the SMBTC register is used to restrict the detection of a START condition under certain circumstances. In some systems where there is significant mismatch between the impedance or the capacitance on the SDA and SCL lines, it may be possible for SCL to fall after SDA during an address or data transfer. Such an event can cause a false START detection on the bus. These kind of events are not expected in a standard SMBus or I2C-compliant system. **In most systems this parameter should not be adjusted, and it is recommended that it be left at its default value.**

By default, if the SCL falling edge is detected after the falling edge of SDA (i.e. one SYSCLK cycle or more), the device will detect this as a START condition. The SDD field is used to increase the amount of hold time that is required between SDA and SCL falling before a START is recognized. An additional 2, 4, or 8 SYSCLKs can be added to prevent false START detection in systems where the bus conditions warrant this.

24.4.4. SMB0CN Control Register

SMB0CN is used to control the interface and to provide status information. The higher four bits of SMB0CN (MASTER, TXMODE, STA, and STO) form a status vector that can be used to jump to service routines. MASTER indicates whether a device is the master or slave during the current transfer. TXMODE indicates whether the device is transmitting or receiving data for the current byte.

Register 24.3. SMB0CN: SMBus0 Control

Bit	7	6	5	4	3	2	1	0
Name	MASTER	TXMODE	STA	STO	ACKRQ	ARBLOST	ACK	SI
Type	R	R	RW	RW	R	R	RW	RW
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xC0 (bit-addressable)								

Table 24.9. SMB0CN Register Bit Descriptions

Bit	Name	Function
7	MASTER	**SMBus0 Master/Slave Indicator.** This read-only bit indicates when the SMBus0 is operating as a master. 0: SMBus0 operating in slave mode. 1: SMBus0 operating in master mode.
6	TXMODE	**SMBus0 Transmit Mode Indicator.** This read-only bit indicates when the SMBus0 is operating as a transmitter. 0: SMBus0 in Receiver Mode. 1: SMBus0 in Transmitter Mode.
5	STA	**SMBus0 Start Flag.** When reading STA, a 1 indicates that a start or repeated start condition was detected on the bus. Writing a 1 to the STA bit initiates a start or repeated start on the bus.
4	STO	**SMBus0 Stop Flag.** When reading STO, a 1 indicates that a stop condition was detected on the bus (in slave mode) or is pending (in master mode). When acting as a master, writing a 1 to the STO bit initiates a stop condition on the bus. This bit is cleared by hardware.
3	ACKRQ	**SMBus0 Acknowledge Request.** 0: No ACK requested. 1: ACK requested.
2	ARBLOST	**SMBus0 Arbitration Lost Indicator.** 0: No arbitration error. 1: Arbitration error occurred.
1	ACK	**SMBus0 Acknowledge.** When read as a master, the ACK bit indicates whether an ACK (1) or NACK (0) is received during the most recent byte transfer. As a slave, this bit should be written to send an ACK (1) or NACK (0) to a master request. Note that the logic level of the ACK bit on the SMBus interface is inverted from the logic of the register ACK bit.

Register 24.5. SMB0ADM: SMBus0 Slave Address Mask

Bit	7	6	5	4	3	2	1	0
Name	SLVM							EHACK
Type	RW							RW
Reset	1	1	1	1	1	1	1	0
SFR Address: 0xD6								

Table 24.11. SMB0ADM Register Bit Descriptions

Bit	Name	Function
7:1	SLVM	**SMBus0 Slave Address Mask.** Defines which bits of register SMB0ADR are compared with an incoming address byte, and which bits are ignored. Any bit set to 1 in SLVM enables comparisons with the corresponding bit in SLV. Bits set to 0 are ignored (can be either 0 or 1 in the incoming address).
0	EHACK	**Hardware Acknowledge Enable.** Enables hardware acknowledgement of slave address and received data bytes. 0: Firmware must manually acknowledge all incoming address and data bytes. 1: Automatic slave address recognition and hardware acknowledge is enabled.

Register 25.9. TMR2RLL: Timer 2 Reload Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TMR2RLL							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xCA								

Table 25.11. TMR2RLL Register Bit Descriptions

Bit	Name	Function
7:0	TMR2RLL	**Timer 2 Reload Low Byte.** When operating in one of the auto-reload modes, TMR2RLL holds the reload value for the low byte of Timer 2 (TMR2L). When operating in capture mode, TMR2RLL is the captured value of TMR2L.

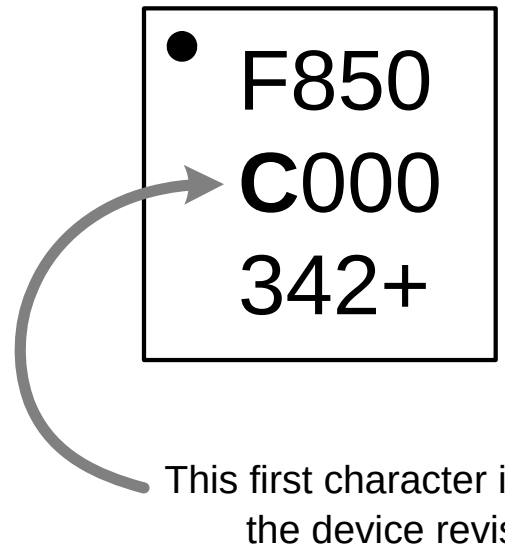

Figure 28.2. QFN-20 Package Revision Marking

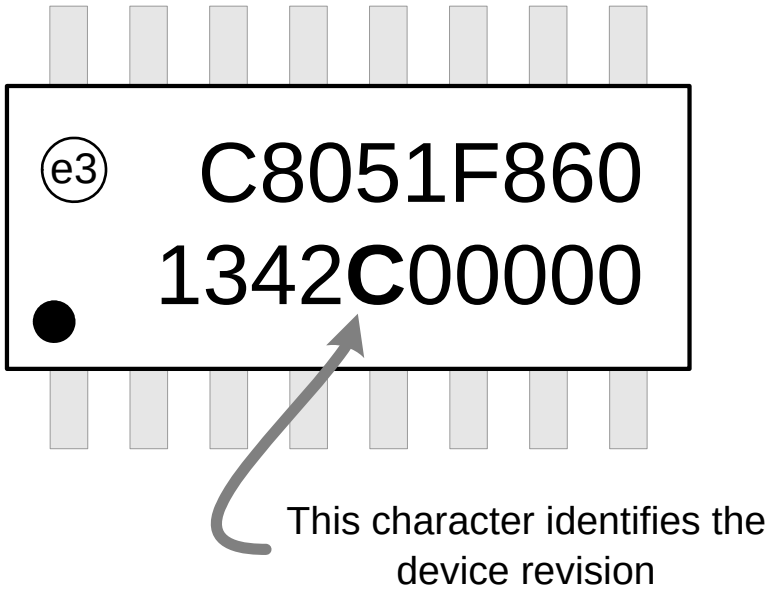

Figure 28.3. SOIC-16 Package Revision Marking

29.2. C2 Interface Registers

The following describes the C2 registers necessary to perform flash programming through the C2 interface. All C2 registers are accessed through the C2 interface, and are not available in the SFR map for firmware access.

Register 29.1. C2ADD: C2 Address

Bit	7	6	5	4	3	2	1	0
Name	C2ADD							
Type	RW							
Reset	0	0	0	0	0	0	0	0
This register is part of the C2 protocol.								

Table 29.1. C2ADD Register Bit Descriptions

Bit	Name	Function
7:0	C2ADD	**C2 Address.** The C2ADD register is accessed via the C2 interface. The value written to C2ADD selects the target data register for C2 Data Read and Data Write commands. 0x00: C2DEVID 0x01: C2REVID 0x02: C2FPCTL 0xB4: C2FPDAT