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	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	EBI/EMI, SMBus (2-Wire/I²C), SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	40
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 20x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f344-gq

Table 1.1. Product Selection Guide

Ordering Part Number	MIPS (Peak)	Flash Memory (Bytes)	RAM	Calibrated Internal Oscillator	Low Frequency Oscillator	USB with 1k Endpoint RAM	Supply Voltage Regulator	SMBus/I2C	Enhanced SPI	UARTs	Timers (16-bit)	Programmable Counter Array	Digital Port I/Os	External Memory Interface (EMIF)	10-bit 200 ksps ADC	Temperature Sensor	Voltage Reference	Analog Comparators	Package
C8051F340-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F341-GQ	48	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F342-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F342-GM	48	64k	4352	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F343-GQ	48	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F343-GM	48	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F344-GQ	25	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F345-GQ	25	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	✓	✓	✓	2	TQFP48
C8051F346-GQ	25	64k	4352	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F346-GM	25	64k	4352	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F347-GQ	25	32k	2304	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F347-GM	25	32k	2304	✓	—	✓	✓	✓	✓	1	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F348-GQ	25	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	—	—	—	2	TQFP48
C8051F349-GQ	25	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	—	—	—	2	LQFP32
C8051F349-GM	25	32k	2304	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	—	—	—	2	QFN32
C8051F34A-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F34A-GM	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F34B-GQ	48	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	LQFP32
C8051F34B-GM	48	32k	2304	✓	✓	✓	✓	✓	✓	2	4	✓	25	—	✓	✓	✓	2	QFN32
C8051F34C-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	2	4	✓	40	✓	—	—	—	2	TQFP48
C8051F34D-GQ	48	64k	4352	✓	✓	✓	✓	✓	✓	1	4	✓	25	—	—	—	—	2	LQFP32

Table 4.1. Pin Definitions for the C8051F340/1/2/3/4/5/6/7/8/9/A/B/C/D (Continued)

Name	Pin Numbers		Type	Description
	48-pin	32-pin		
P1.0	46	26	D I/O or A In	Port 1.0. See **Section 15** for a complete description of Port 1.
P1.1	45	25	D I/O or A In	Port 1.1.
P1.2	44	24	D I/O or A In	Port 1.2.
P1.3	43	23	D I/O or A In	Port 1.3.
P1.4	42	22	D I/O or A In	Port 1.4.
P1.5	41	21	D I/O or A In	Port 1.5.
P1.6	40	20	D I/O or A In	Port 1.6.
P1.7	39	19	D I/O or A In	Port 1.7.
P2.0	38	18	D I/O or A In	Port 2.0. See **Section 15** for a complete description of Port 2.
P2.1	37	17	D I/O or A In	Port 2.1.
P2.2	36	16	D I/O or A In	Port 2.2.
P2.3	35	15	D I/O or A In	Port 2.3.
P2.4	34	14	D I/O or A In	Port 2.4.
P2.5	33	13	D I/O or A In	Port 2.5.
P2.6	32	12	D I/O or A In	Port 2.6.
P2.7	31	11	D I/O or A In	Port 2.7.
P3.0	30	—	D I/O or A In	Port 3.0. See **Section 15** for a complete description of Port 3.
P3.1	29	—	D I/O or A In	Port 3.1.
P3.2	28	—	D I/O or A In	Port 3.2.

SFR Definition 5.2. AMX0N: AMUX0 Negative Channel Select

R	R	R	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	AMX0N4	AMX0N3	AMX0N2	AMX0N1	AMX0N0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xBA

Bits7–5: UNUSED. Read = 000b; Write = don't care.
Bits4–0: AMX0N4–0: AMUX0 Negative Input Selection.
Note that when GND is selected as the Negative Input, ADC0 operates in Single-ended mode. For all other Negative Input selections, ADC0 operates in Differential mode.

AMX0N4-0	ADC0 Negative Input (32-pin Package)	ADC0 Negative Input (48-pin Package)
00000	P1.0	P2.0
00001	P1.1	P2.1
00010	P1.2	P2.2
00011	P1.3	P2.3
00100	P1.4	P2.5
00101	P1.5	P2.6
00110	P1.6	P3.0
00111	P1.7	P3.1
01000	P2.0	P3.4
01001	P2.1	P3.5
01010	P2.2	P3.7
01011	P2.3	P4.0
01100	P2.4	P4.3
01101	P2.5	P4.4
01110	P2.6	P4.5
01111	P2.7	P4.6
10000	P3.0	RESERVED
10001	P0.0	P0.3
10010	P0.1	P0.4
10011	P0.4	P1.1
10100	P0.5	P1.2
10101 - 11101	RESERVED	RESERVED
11110	VREF	VREF
11111	GND (Single-Ended Mode)	GND (Single-Ended Mode)

SFR Definition 5.6. ADC0CN: ADC0 Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM2	AD0CM1	AD0CM0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xE8

Bit7: AD0EN: ADC0 Enable Bit.
0: ADC0 Disabled. ADC0 is in low-power shutdown.
1: ADC0 Enabled. ADC0 is active and ready for data conversions.

Bit6: AD0TM: ADC0 Track Mode Bit.
0: Normal Track Mode: When ADC0 is enabled, tracking is continuous unless a conversion is in progress.
1: Low-power Track Mode: Tracking Defined by AD0CM2-0 bits (see below).

Bit5: AD0INT: ADC0 Conversion Complete Interrupt Flag.
0: ADC0 has not completed a data conversion since the last time AD0INT was cleared.
1: ADC0 has completed a data conversion.

Bit4: AD0BUSY: ADC0 Busy Bit.
Read:
0: ADC0 conversion is complete or a conversion is not currently in progress. AD0INT is set to logic 1 on the falling edge of AD0BUSY.
1: ADC0 conversion is in progress.
Write:
0: No Effect.
1: Initiates ADC0 Conversion if AD0CM2-0 = 000b

Bit3: AD0WINT: ADC0 Window Compare Interrupt Flag.
0: ADC0 Window Comparison Data match has not occurred since this flag was last cleared.
1: ADC0 Window Comparison Data match has occurred.

Bits2–0: AD0CM2–0: ADC0 Start of Conversion Mode Select.
When AD0TM = 0:
000: ADC0 conversion initiated on every 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 1.
100: ADC0 conversion initiated on rising edge of external CNVSTR.
101: ADC0 conversion initiated on overflow of Timer 3.
11x: Reserved.
When AD0TM = 1:
000: Tracking initiated on write of ‘1’ to AD0BUSY and lasts 3 SAR clocks, followed by conversion.
001: Tracking initiated on overflow of Timer 0 and lasts 3 SAR clocks, followed by conversion.
010: Tracking initiated on overflow of Timer 2 and lasts 3 SAR clocks, followed by conversion.
011: Tracking initiated on overflow of Timer 1 and lasts 3 SAR clocks, followed by conversion.
100: ADC0 tracks only when CNVSTR input is logic low; conversion starts on rising CNVSTR edge.
101: Tracking initiated on overflow of Timer 3 and lasts 3 SAR clocks, followed by conversion.
11x: Reserved.

9.1.2. MOVX Instruction and Program Memory

In the CIP-51, the MOVX instruction serves three purposes: accessing on-chip XRAM, accessing off-chip data XRAM (only on C8051F340/1/4/5/8 devices), and accessing on-chip program Flash memory. The Flash access feature provides a mechanism for user software to update program code and use the program memory space for non-volatile data storage (see **Section "12. Flash Memory" on page 107**). The External Memory Interface (only on C8051F340/1/4/5/8 devices) provides a fast access interface to off-chip data XRAM (or memory-mapped peripherals) via the MOVX instruction. Refer to Section "**13. External Data Memory Interface and On-Chip XRAM**" on page **114**. for details.

Table 9.1. CIP-51 Instruction Set Summary

Mnemonic	Description	Bytes	Clock Cycles
Arithmetic Operations			
ADD A, Rn	Add register to A	1	1
ADD A, direct	Add direct byte to A	2	2
ADD A, @Ri	Add indirect RAM to A	1	2
ADD A, #data	Add immediate to A	2	2
ADDC A, Rn	Add register to A with carry	1	1
ADDC A, direct	Add direct byte to A with carry	2	2
ADDC A, @Ri	Add indirect RAM to A with carry	1	2
ADDC A, #data	Add immediate to A with carry	2	2
SUBB A, Rn	Subtract register from A with borrow	1	1
SUBB A, direct	Subtract direct byte from A with borrow	2	2
SUBB A, @Ri	Subtract indirect RAM from A with borrow	1	2
SUBB A, #data	Subtract immediate from A with borrow	2	2
INC A	Increment A	1	1
INC Rn	Increment register	1	1
INC direct	Increment direct byte	2	2
INC @Ri	Increment indirect RAM	1	2
DEC A	Decrement A	1	1
DEC Rn	Decrement register	1	1
DEC direct	Decrement direct byte	2	2
DEC @Ri	Decrement indirect RAM	1	2
INC DPTR	Increment Data Pointer	1	1
MUL AB	Multiply A and B	1	4
DIV AB	Divide A by B	1	8
DA A	Decimal adjust A	1	1
Logical Operations			
ANL A, Rn	AND Register to A	1	1
ANL A, direct	AND direct byte to A	2	2
ANL A, @Ri	AND indirect RAM to A	1	2
ANL A, #data	AND immediate to A	2	2
ANL direct, A	AND A to direct byte	2	2
ANL direct, #data	AND immediate to direct byte	3	3
ORL A, Rn	OR Register to A	1	1
ORL A, direct	OR direct byte to A	2	2
ORL A, @Ri	OR indirect RAM to A	1	2

Table 9.1. CIP-51 Instruction Set Summary (Continued)

Mnemonic	Description	Bytes	Clock Cycles
Boolean Manipulation			
CLR C	Clear Carry	1	1
CLR bit	Clear direct bit	2	2
SETB C	Set Carry	1	1
SETB bit	Set direct bit	2	2
CPL C	Complement Carry	1	1
CPL bit	Complement direct bit	2	2
ANL C, bit	AND direct bit to Carry	2	2
ANL C, /bit	AND complement of direct bit to Carry	2	2
ORL C, bit	OR direct bit to carry	2	2
ORL C, /bit	OR complement of direct bit to Carry	2	2
MOV C, bit	Move direct bit to Carry	2	2
MOV bit, C	Move Carry to direct bit	2	2
JC rel	Jump if Carry is set	2	2/4
JNC rel	Jump if Carry is not set	2	2/4
JB bit, rel	Jump if direct bit is set	3	3/5
JNB bit, rel	Jump if direct bit is not set	3	3/5
JBC bit, rel	Jump if direct bit is set and clear bit	3	3/5
Program Branching			
ACALL addr11	Absolute subroutine call	2	4
LCALL addr16	Long subroutine call	3	5
RET	Return from subroutine	1	6
RETI	Return from interrupt	1	6
AJMP addr11	Absolute jump	2	4
LJMP addr16	Long jump	3	5
SJMP rel	Short jump (relative address)	2	4
JMP @A+DPTR	Jump indirect relative to DPTR	1	4
JZ rel	Jump if A equals zero	2	2/4
JNZ rel	Jump if A does not equal zero	2	2/4
CJNE A, direct, rel	Compare direct byte to A and jump if not equal	3	3/5
CJNE A, #data, rel	Compare immediate to A and jump if not equal	3	3/5
CJNE Rn, #data, rel	Compare immediate to Register and jump if not equal	3	3/5
CJNE @Ri, #data, rel	Compare immediate to indirect and jump if not equal	3	4/6
DJNZ Rn, rel	Decrement Register and jump if not zero	2	2/4
DJNZ direct, rel	Decrement direct byte and jump if not zero	3	3/5
NOP	No operation	1	1

SFR Definition 11.2. RSTSRC: Reset Source

R/W	R	R/W	R/W	R	R/W	R/W	R	Reset Value
USBRSF	FERROR	C0RSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF	Variable
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xEF

Bit7: USBRSF: USB Reset Flag
0: **Read:** Last reset was not a USB reset; **Write:** USB resets disabled.
1: **Read:** Last reset was a USB reset; **Write:** USB resets enabled.

Bit6: FERROR: Flash Error Indicator.
0: Source of last reset was not a Flash read/write/erase error.
1: Source of last reset was a Flash read/write/erase error.

Bit5: C0RSEF: Comparator0 Reset Enable and Flag.
0: **Read:** Source of last reset was not Comparator0; **Write:** Comparator0 is not a reset source.
1: **Read:** Source of last reset was Comparator0; **Write:** Comparator0 is a reset source (active-low).

Bit4: SWRSF: Software Reset Force and Flag.
0: **Read:** Source of last reset was not a write to the SWRSF bit; **Write:** No Effect.
1: **Read:** Source of last was a write to the SWRSF bit; **Write:** Forces a system reset.

Bit3: WDTRSF: Watchdog Timer Reset Flag.
0: Source of last reset was not a WDT timeout.
1: Source of last reset was a WDT timeout.

Bit2: MCDRSF: Missing Clock Detector Flag.
0: **Read:** Source of last reset was not a Missing Clock Detector timeout; **Write:** Missing Clock Detector disabled.
1: **Read:** Source of last reset was a Missing Clock Detector timeout; **Write:** Missing Clock Detector enabled; triggers a reset if a missing clock condition is detected.

Bit1: PORSF: Power-On / V_{DD} Monitor Reset Flag.
This bit is set anytime a power-on reset occurs. Writing this bit selects/deselects the V_{DD} monitor as a reset source. **Note: writing '1' to this bit before the V_{DD} monitor is enabled and stabilized can cause a system reset.** See register VDM0CN (SFR Definition 11.1).
0: **Read:** Last reset was not a power-on or V_{DD} monitor reset; **Write:** V_{DD} monitor is not a reset source.
1: **Read:** Last reset was a power-on or V_{DD} monitor reset; all other reset flags indeterminate; **Write:** V_{DD} monitor is a reset source.

Bit0: PINRSF: HW Pin Reset Flag.
0: Source of last reset was not $\overline{\text{RST}}$ pin.
1: Source of last reset was $\overline{\text{RST}}$ pin.

Note: For bits that act as both reset source enables (on a write) and reset indicator flags (on a read), read-modify-write instructions read and modify the source enable only. This applies to bits: USBRSF, C0RSEF, SWRSF, MCDRSF, PORSF.

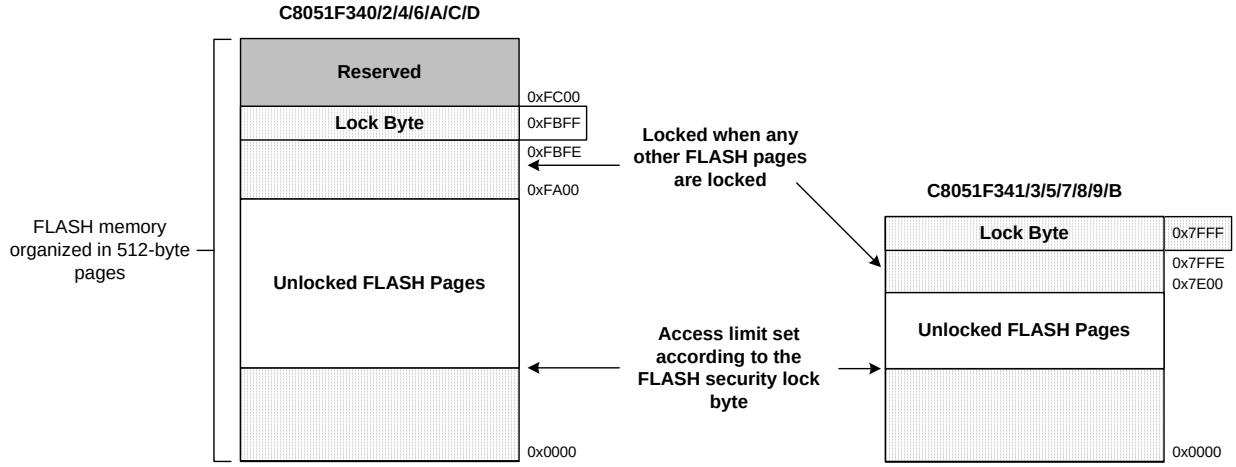

Figure 12.1. Flash Program Memory Map and Security Byte

Internal Oscillator		
Clock Signal	**Input Source Selection**	**Register Bit Settings**
USB Clock	External Oscillator / 4	USBCLK = 101b
External Oscillator	Crystal Oscillator Mode 24 MHz Crystal	XOSCMD = 110b XFCN = 111b

SFR Definition 14.6. CLKSEL: Clock Select

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	USBCLK			-	CLKSL			00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address
								0xA9

Bit 7: Unused. Read = 0b; Write = don't care.

Bits6–4: USBCLK2–0: USB Clock Select

These bits select the clock supplied to USB0. When operating USB0 in full-speed mode, the selected clock should be 48 MHz. When operating USB0 in low-speed mode, the selected clock should be 6 MHz.

USBCLK	Selected Clock
000	4x Clock Multiplier
001	Internal Oscillator / 2
010	External Oscillator
011	External Oscillator / 2
100	External Oscillator / 3
101	External Oscillator / 4
110	RESERVED
111	RESERVED

Bit3: Unused. Read = 0b; Write = don't care.

Bits2–0: CLKSL2–0: System Clock Select

These bits select the system clock source. When operating from a system clock of 25 MHz or less, the FLRT bit should be set to '0'. When operating with a system clock of greater than 25 MHz (up to 48 MHz), the FLRT bit (FLSCL.4) should be set to '1'. See **Section "10. Prefetch Engine" on page 99** for more details.

CLKSL	Selected Clock
000	Internal Oscillator (as determined by the IFCN bits in register OSCICN)
001	External Oscillator
010	4x Clock Multiplier / 2
011*	4x Clock Multiplier*
100	Low-Frequency Oscillator
101-111	RESERVED
***Note:** This option is only available on 48 MHz devices.	

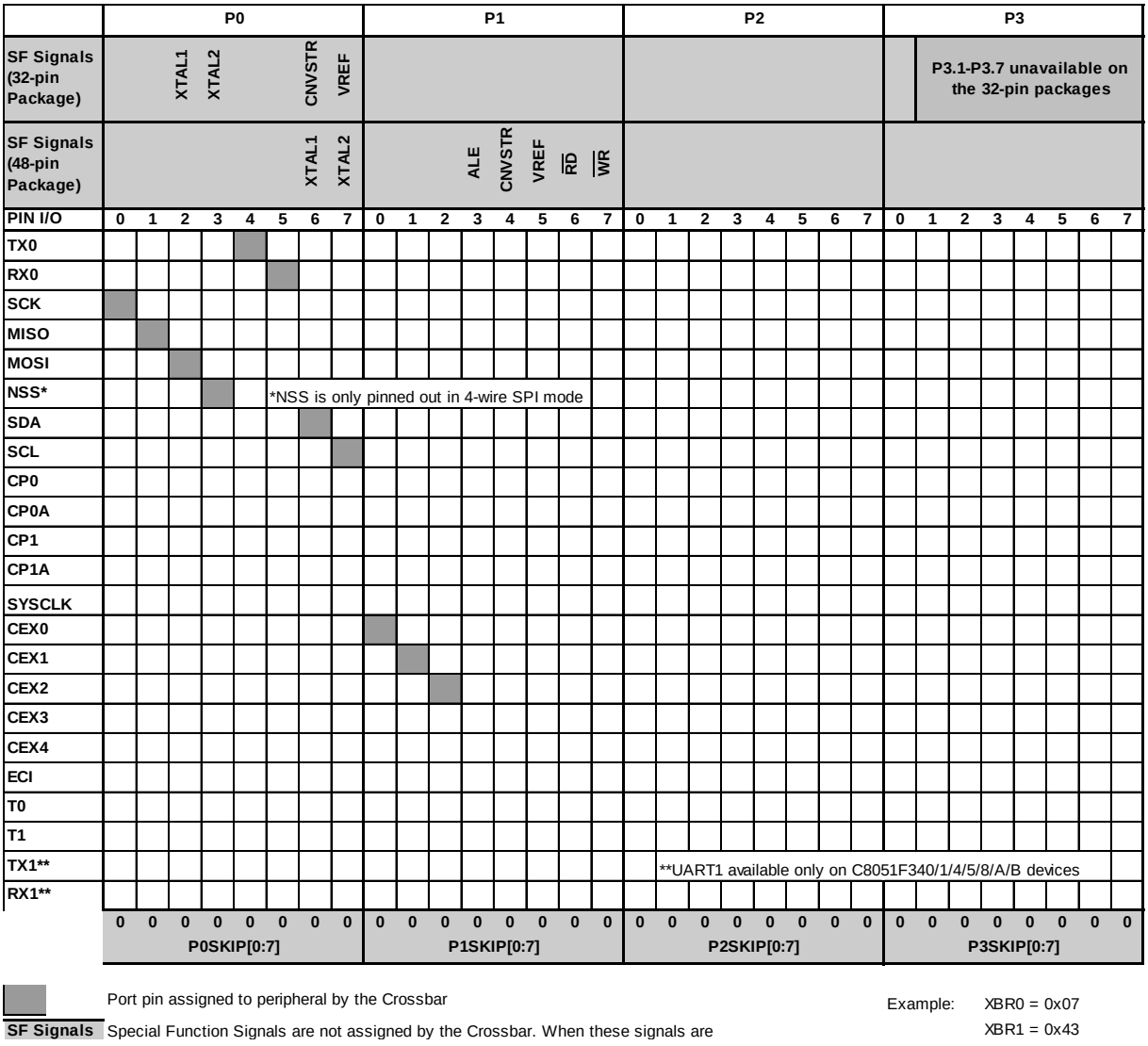

Figure 15.4. Crossbar Priority Decoder in Example Configuration (No Pins Skipped)

SFR Definition 15.2. XBR1: Port I/O Crossbar Register 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
WEAKPUD	XBARE	T1E	T0E	ECIE	PCA0ME			00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xE2

Bit7: WEAKPUD: Port I/O Weak Pull-up Disable.
0: Weak Pull-ups enabled (except for Ports whose I/O are configured as analog input or push-pull output).
1: Weak Pull-ups disabled.

Bit6: XBARE: Crossbar Enable.
0: Crossbar disabled; all Port drivers disabled.
1: Crossbar enabled.

Bit5: T1E: T1 Enable
0: T1 unavailable at Port pin.
1: T1 routed to Port pin.

Bit4: T0E: T0 Enable
0: T0 unavailable at Port pin.
1: T0 routed to Port pin.

Bit3: ECIE: PCA0 External Counter Input Enable
0: ECI unavailable at Port pin.
1: ECI routed to Port pin.

Bits2–0: PCA0ME: PCA Module I/O Enable Bits.
000: All PCA I/O unavailable at Port pins.
001: CEX0 routed to Port pin.
010: CEX0, CEX1 routed to Port pins.
011: CEX0, CEX1, CEX2 routed to Port pins.
100: CEX0, CEX1, CEX2, CEX3 routed to Port pins.
101: CEX0, CEX1, CEX2, CEX3, CEX4 routed to Port pins.
110: Reserved.
111: Reserved.

SFR Definition 15.3. XBR2: Port I/O Crossbar Register 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
							URT1E	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xE3

Bits7–1: RESERVED: Always write to 0000000b

Bit0: URT1E: UART1 I/O Output Enable (C8051F340/1/4/5/8/A/B Only)
0: UART1 I/O unavailable at Port pins.
1: UART1 TX1, RX1 routed to Port pins.

Table 16.2. USB0 Controller Registers

USB Register Name	USB Register Address	Description	Page Number
Interrupt Registers			
IN1INT	0x02	Endpoint0 and Endpoints1-3 IN Interrupt Flags	173
OUT1INT	0x04	Endpoints1-3 OUT Interrupt Flags	173
CMINT	0x06	Common USB Interrupt Flags	174
IN1IE	0x07	Endpoint0 and Endpoints1-3 IN Interrupt Enables	175
OUT1IE	0x09	Endpoints1-3 OUT Interrupt Enables	175
CMIE	0x0B	Common USB Interrupt Enables	176
Common Registers			
FADDR	0x00	Function Address	169
POWER	0x01	Power Management	171
FRAMEL	0x0C	Frame Number Low Byte	172
FRAMEH	0x0D	Frame Number High Byte	172
INDEX	0x0E	Endpoint Index Selection	165
CLKREC	0x0F	Clock Recovery Control	166
FIFOn	0x20–0x23	Endpoints0-3 FIFOs	168
Indexed Registers			
E0CSR	0x11	Endpoint0 Control / Status	179
EINCSRL		Endpoint IN Control / Status Low Byte	182
EINCSRH	0x12	Endpoint IN Control / Status High Byte	183
EOUTCSRL	0x14	Endpoint OUT Control / Status Low Byte	185
EOUTCSRH	0x15	Endpoint OUT Control / Status High Byte	186
E0CNT	0x16	Number of Received Bytes in Endpoint0 FIFO	180
EOUTCNTL		Endpoint OUT Packet Count Low Byte	186
EOUTCNTH	0x17	Endpoint OUT Packet Count High Byte	186

USB Register Definition 16.4. INDEX: USB0 Endpoint Index

R	R	R	R	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	EPSEL				00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	USB Address: 0x0E

Bits7–4: Unused. Read = 0000b; Write = don't care.
Bits3–0: EPSEL: Endpoint Select
These bits select which endpoint is targeted when indexed USB0 registers are accessed.

INDEX	Target Endpoint
0x0	0
0x1	1
0x2	2
0x3	3
0x4–0xF	Reserved

USB Register Definition 16.21. EOUTCSRL: USB0 OUT Endpoint Control Low Byte

W	R/W	R/W	R/W	R	R/W	R	R/W	Reset Value
CLRDT	STSTL	SDSTL	FLUSH	DATERR	OVRUN	FIFOFUL	OPRDY	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	USB Address: 0x14

Bit7: CLRDT: Clear Data Toggle
Write: Software should write ‘1’ to this bit to reset the OUT endpoint data toggle to ‘0’.
Read: This bit always reads ‘0’.

Bit6: STSTL: Sent Stall
Hardware sets this bit to ‘1’ when a STALL handshake signal is transmitted. This flag must be cleared by software.

Bit5: SDSTL: Send Stall
Software should write ‘1’ to this bit to generate a STALL handshake. Software should write ‘0’ to this bit to terminate the STALL signal. This bit has no effect in ISO mode.

Bit4: FLUSH: FIFO Flush
Writing a ‘1’ to this bit flushes the next packet to be read from the OUT endpoint FIFO. The FIFO pointer is reset and the OPRDY bit is cleared. If the FIFO contains multiple packets, software must write ‘1’ to FLUSH for each packet. Hardware resets the FLUSH bit to ‘0’ when the FIFO flush is complete.

Note: If data for the current packet has already been read from the FIFO, the FLUSH bit should not be used to flush the packet. Instead, the entire data packet should be read from the FIFO manually.

Bit3: DATERR: Data Error
In ISO mode, this bit is set by hardware if a received packet has a CRC or bit-stuffing error. It is cleared when software clears OPRDY. This bit is only valid in ISO mode.

Bit2: OVRUN: Data Overrun
This bit is set by hardware when an incoming data packet cannot be loaded into the OUT endpoint FIFO. This bit is only valid in ISO mode, and must be cleared by software.
0: No data overrun.
1: A data packet was lost because of a full FIFO since this flag was last cleared.

Bit1: FIFOFUL: OUT FIFO Full
This bit indicates the contents of the OUT FIFO. If double buffering is enabled for the endpoint (DBIEN = ‘1’), the FIFO is full when the FIFO contains two packets. If DBIEN = ‘0’, the FIFO is full when the FIFO contains one packet.
0: OUT endpoint FIFO is not full.
1: OUT endpoint FIFO is full.

Bit0: OPRDY: OUT Packet Ready
Hardware sets this bit to ‘1’ and generates an interrupt when a data packet is available. Software should clear this bit after each data packet is unloaded from the OUT endpoint FIFO.

The direction bit (R/W) occupies the least-significant bit position of the address byte. The direction bit is set to logic 1 to indicate a "READ" operation and cleared to logic 0 to indicate a "WRITE" operation.

All transactions are initiated by a master, with one or more addressed slave devices as the target. The master generates the START condition and then transmits the slave address and direction bit. If the transaction is a WRITE operation from the master to the slave, the master transmits the data a byte at a time waiting for an ACK from the slave at the end of each byte. For READ operations, the slave transmits the data waiting for an ACK from the master at the end of each byte. At the end of the data transfer, the master generates a STOP condition to terminate the transaction and free the bus. Figure 17.3 illustrates a typical SMBus transaction.

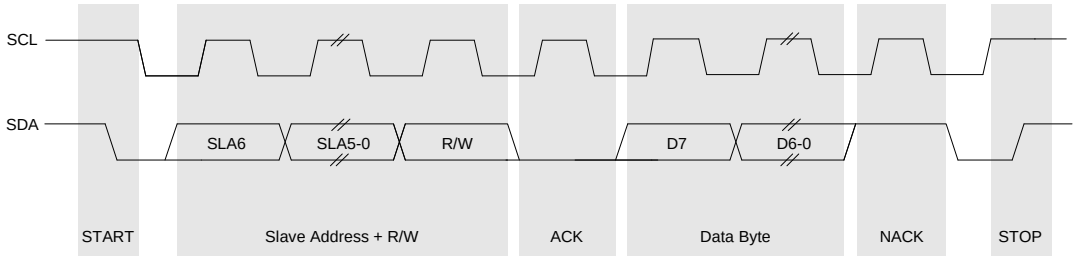

Figure 17.3. SMBus Transaction

17.3.1. Arbitration

A master may start a transfer only if the bus is free. The bus is free after a STOP condition or after the SCL and SDA lines remain high for a specified time (see **Section “17.3.4. SCL High (SMBus Free) Timeout” on page 191**). In the event that two or more devices attempt to begin a transfer at the same time, an arbitration scheme is employed to force one master to give up the bus. The master devices continue transmitting until one attempts a HIGH while the other transmits a LOW. Since the bus is open-drain, the bus will be pulled LOW. The master attempting the HIGH will detect a LOW SDA and lose the arbitration. The winning master continues its transmission without interruption; the losing master becomes a slave and receives the rest of the transfer if addressed. This arbitration scheme is non-destructive: one device always wins, and no data is lost.

20.3. SPI0 Slave Mode Operation

When SPI0 is enabled and not configured as a master, it will operate as a SPI slave. As a slave, bytes are shifted in through the MOSI pin and out through the MISO pin by a master device controlling the SCK signal. A bit counter in the SPI0 logic counts SCK edges. When 8 bits have been shifted through the shift register, the SPIF flag is set to logic 1, and the byte is copied into the receive buffer. Data is read from the receive buffer by reading SPI0DAT. A slave device cannot initiate transfers. Data to be transferred to the master device is pre-loaded into the shift register by writing to SPI0DAT. Writes to SPI0DAT are double-buffered, and are placed in the transmit buffer first. If the shift register is empty, the contents of the transmit buffer will immediately be transferred into the shift register. When the shift register already contains data, the SPI will load the shift register with the transmit buffer's contents after the last SCK edge of the next (or current) SPI transfer.

When configured as a slave, SPI0 can be configured for 4-wire or 3-wire operation. The default, 4-wire slave mode, is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. In 4-wire mode, the NSS signal is routed to a port pin and configured as a digital input. SPI0 is enabled when NSS is logic 0, and disabled when NSS is logic 1. The bit counter is reset on a falling edge of NSS. Note that the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 20.4 shows a connection diagram between two slave devices in 4-wire slave mode and a master device.

3-wire slave mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. NSS is not used in this mode, and is not mapped to an external port pin through the crossbar. Since there is no way of uniquely addressing the device in 3-wire slave mode, SPI0 must be the only slave device present on the bus. It is important to note that in 3-wire slave mode there is no external means of resetting the bit counter that determines when a full byte has been received. The bit counter can only be reset by disabling and re-enabling SPI0 with the SPIEN bit. Figure 20.3 shows a connection diagram between a slave device in 3-wire slave mode and a master device.

20.4. SPI0 Interrupt Sources

When SPI0 interrupts are enabled, the following four flags will generate an interrupt when they are set to logic 1:

Note that all of the following bits must be cleared by software.

1. The SPI Interrupt Flag, SPIF (SPI0CN.7) is set to logic 1 at the end of each byte transfer. This flag can occur in all SPI0 modes.
2. The Write Collision Flag, WCOL (SPI0CN.6) is set to logic 1 if a write to SPI0DAT is attempted when the transmit buffer has not been emptied to the SPI shift register. When this occurs, the write to SPI0DAT will be ignored, and the transmit buffer will not be written.This flag can occur in all SPI0 modes.
3. The Mode Fault Flag MODF (SPI0CN.5) is set to logic 1 when SPI0 is configured as a master, and for multi-master mode and the NSS pin is pulled low. When a Mode Fault occurs, the MSTEN and SPIEN bits in SPI0CN are set to logic 0 to disable SPI0 and allow another master device to access the bus.
4. The Receive Overrun Flag RXOVRN (SPI0CN.4) is set to logic 1 when configured as a slave, and a transfer is completed and the receive buffer still holds an unread byte from a previous transfer. The new byte is not transferred to the receive buffer, allowing the previously received data byte to be read. The data byte which caused the overrun is lost.

21.3. Timer 3

Timer 3 is a 16-bit timer formed by two 8-bit SFRs: TMR3L (low byte) and TMR3H (high byte). Timer 3 may operate in 16-bit auto-reload mode, (split) 8-bit auto-reload mode, USB Start-of-Frame (SOF) capture mode, or Low-Frequency Oscillator (LFO) Rising Edge capture mode. The Timer 3 operation mode is defined by the T3SPLIT (TMR3CN.3), T3CE (TMR3CN.4) bits, and T3CSS (TMR3CN.1) bits.

Timer 3 may be clocked by the system clock, the system clock divided by 12, or the external oscillator source divided by 8. The external clock mode is ideal for real-time clock (RTC) functionality, where the internal oscillator drives the system clock while Timer 3 (and/or the PCA) is clocked by an external precision oscillator. Note that the external oscillator source divided by 8 is synchronized with the system clock.

21.3.1. 16-bit Timer with Auto-Reload

When T3SPLIT (TMR3CN.3) is ‘0’ and T3CE = ‘0’, Timer 3 operates as a 16-bit timer with auto-reload. Timer 3 can be clocked by SYSCLK, SYSCLK divided by 12, or the external oscillator clock source divided by 8. As the 16-bit timer register increments and overflows from 0xFFFF to 0x0000, the 16-bit value in the Timer 3 reload registers (TMR3RLH and TM3RLL) is loaded into the Timer 3 register as shown in Figure 21.4, and the Timer 3 High Byte Overflow Flag (TMR3CN.7) is set. If Timer 3 interrupts are enabled, an interrupt will be generated on each Timer 3 overflow. Additionally, if Timer 3 interrupts are enabled and the TF3LEN bit is set (TMR3CN.5), an interrupt will be generated each time the lower 8 bits (TMR3L) overflow from 0xFF to 0x00.

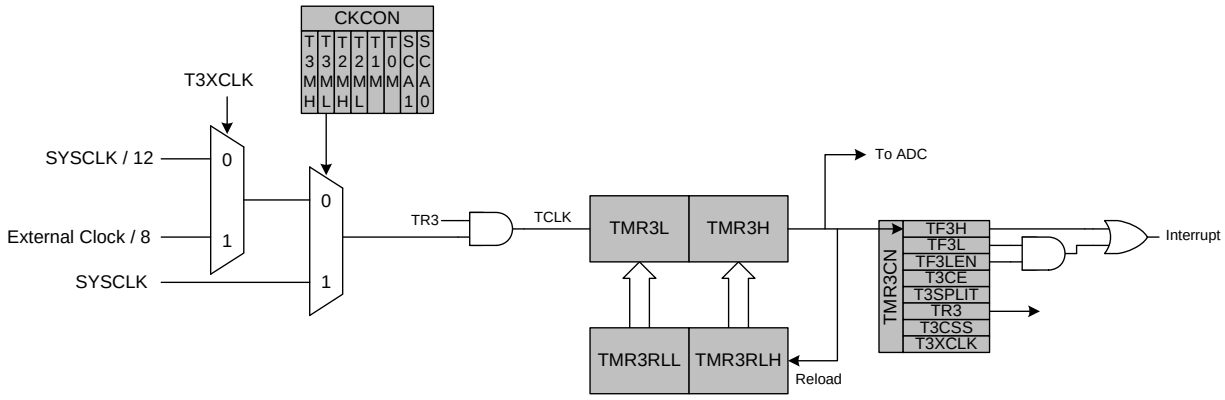

Figure 21.8. Timer 3 16-Bit Mode Block Diagram

22.2.1. Edge-triggered Capture Mode

In this mode, a valid transition on the CEXn pin causes the PCA to capture the value of the PCA counter/timer and load it into the corresponding module's 16-bit capture/compare register (PCA0CPLn and PCA0CPHn). The CAPPn and CAPNn bits in the PCA0CPMn register are used to select the type of transition that triggers the capture: low-to-high transition (positive edge), high-to-low transition (negative edge), or either transition (positive or negative edge). When a capture occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1 and an interrupt request is generated if CCF interrupts are enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. If both CAPPn and CAPNn bits are set to logic 1, then the state of the Port pin associated with CEXn can be read directly to determine whether a rising-edge or falling-edge caused the capture.

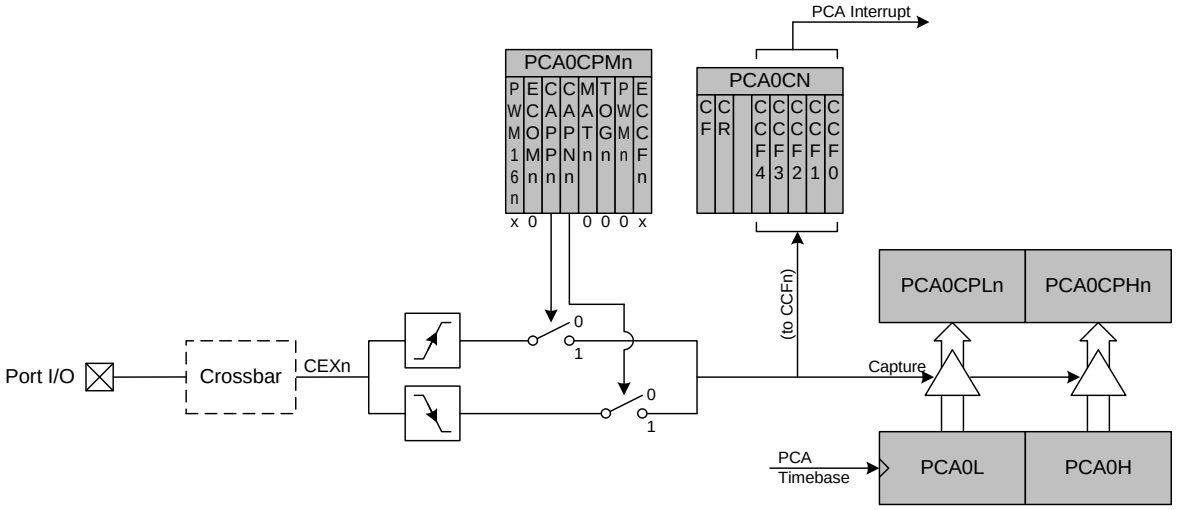

Figure 22.4. PCA Capture Mode Diagram

Note: The CEXn input signal must remain high or low for at least 2 system clock cycles to be recognized by the hardware.

SFR Definition 22.4. PCA0L: PCA Counter/Timer Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xF9

Bits 7–0: PCA0L: PCA Counter/Timer Low Byte.
The PCA0L register holds the low byte (LSB) of the 16-bit PCA Counter/Timer.

SFR Definition 22.5. PCA0H: PCA Counter/Timer High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xFA

Bits 7–0: PCA0H: PCA Counter/Timer High Byte.
The PCA0H register holds the high byte (MSB) of the 16-bit PCA Counter/Timer.

SFR Definition 22.6. PCA0CPLn: PCA Capture Module Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xFB, 0xE9, 0xEB, 0xED, 0xFD

PCA0CPLn Address: PCA0CPL0 = 0xFB (n = 0), PCA0CPL1 = 0xE9 (n = 1),
PCA0CPL2 = 0xEB (n = 2), PCA0CPL3 = 0xED (n = 3),
PCA0CPL4 = 0xFD (n = 4)

Bits7–0: PCA0CPLn: PCA Capture Module Low Byte.
The PCA0CPLn register holds the low byte (LSB) of the 16-bit capture module n.

NOTES: