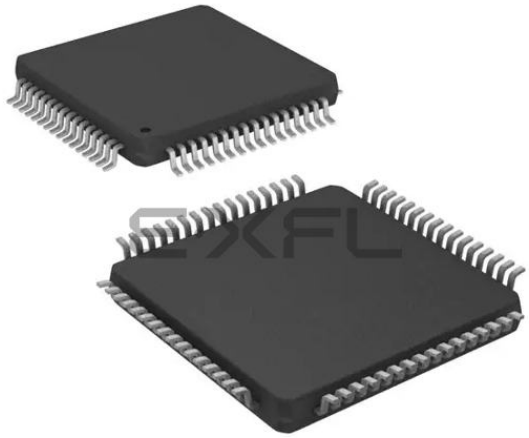


Welcome to E-XFL.COM

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	CANbus, SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	24
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 2x16b, 8x10b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f061

Figure 13.17. ACC: Accumulator 150
Figure 13.18. B: B Register 150
Figure 13.19. IE: Interrupt Enable 154
Figure 13.20. IP: Interrupt Priority 155
Figure 13.21. EIE1: Extended Interrupt Enable 1 156
Figure 13.22. EIE2: Extended Interrupt Enable 2 157
Figure 13.23. EIP1: Extended Interrupt Priority 1 158
Figure 13.24. EIP2: Extended Interrupt Priority 2 159
Figure 13.25. PCON: Power Control 161
14. Reset Sources 163
Figure 14.1. Reset Sources 163
Figure 14.2. Reset Timing 164
Figure 14.3. WDTCN: Watchdog Timer Control Register 167
Figure 14.4. RSTSRC: Reset Source Register 168
15. Oscillators 171
Figure 15.1. Oscillator Diagram 171
Figure 15.2. OSCICL: Internal Oscillator Calibration Register 172
Figure 15.3. OSCICN: Internal Oscillator Control Register 172
Figure 15.4. CLKSEL: Oscillator Clock Selection Register 173
Figure 15.5. OSCXCN: External Oscillator Control Register 174
16. Flash Memory 177
Figure 16.1. C8051F060/1/2/3/4/5 Flash Program Memory Map and Security Bytes .. 180
Figure 16.2. C8051F066/7 Flash Program Memory Map and Security Bytes 181
Figure 16.3. FLACL: Flash Access Limit 182
Figure 16.4. FLSCL: Flash Memory Control 184
Figure 16.5. PSCTL: Program Store Read/Write Control 185
17. External Data Memory Interface and On-Chip XRAM 187
Figure 17.1. EMI0CN: External Memory Interface Control 189
Figure 17.2. EMI0CF: External Memory Configuration 189
Figure 17.3. Multiplexed Configuration Example 190
Figure 17.4. Non-multiplexed Configuration Example 191
Figure 17.5. EMIF Operating Modes 192
Figure 17.6. EMI0TC: External Memory Timing Control 194
Figure 17.7. Non-multiplexed 16-bit MOVX Timing 196
Figure 17.8. Non-multiplexed 8-bit MOVX without Bank Select Timing 197
Figure 17.9. Non-multiplexed 8-bit MOVX with Bank Select Timing 198
Figure 17.10. Multiplexed 16-bit MOVX Timing 199
Figure 17.11. Multiplexed 8-bit MOVX without Bank Select Timing 200
Figure 17.12. Multiplexed 8-bit MOVX with Bank Select Timing 201
18. Port Input/Output 203
Figure 18.1. Port I/O Cell Block Diagram 203
Figure 18.2. Port I/O Functional Block Diagram 204
Figure 18.3. Priority Crossbar Decode Table 205
Figure 18.4. Crossbar Example: 209

Figure 5.8. ADC1CF: ADC1 Configuration Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD1SC3	AD1SC2	AD1SC1	AD1SC0	AD1SCAL	AD1GCAL	AD1LCAL	AD1OCAL	11110000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xBC
SFR Page: 1

Bits 7-4: AD1SC3-0: ADC1 SAR Conversion Clock Period Bits.
SAR Conversion clock is divided down from the system clock according to the AD1SC bits (AD1SC3-0). The number of system clocks used for each SAR conversion clock is equal to AD1SC + 1. (Note: the ADC1 SAR Conversion Clock should be less than or equal to 25 MHz). See Table 5.1 for conversion timing details.

Bit 3: AD1SCAL: System Calibration Enable.
0: Internal ground and reference voltage are used for offset and gain calibration.
1: External voltages can be used for offset and gain calibration.

Bit 2: AD1GCAL: Gain Calibration.
Read:
0: Gain Calibration is completed or not yet started.
1: Gain Calibration is in progress.
Write:
0: No Effect.
1: Initiates a gain calibration if ADC1 is idle.

Bit 1: AD1LCAL: Linearity Calibration
Read
0: Linearity Calibration is completed or not yet started
1: Linearity Calibration is in progress
Write
0: No Effect
1: Initiates a linearity calibration if ADC1 is idle

Bit 0: AD1OCAL: Offset Calibration.
Read:
0: Offset Calibration is completed or not yet started.
1: Offset Calibration is in progress.
Write:
0: No Effect.
1: Initiates an offset calibration if ADC1 is idle.

Figure 5.20. Offset and Gain Register Mapping

The offset register value affects the offset at the analog input as follows:

Offset Register (14 Bits)	Approximate Offset Change (V)
0x3FFF	-3.125% * VREF
0x2000	0
0x0000	+3.125% * VREF

$$\text{Offset Change} \cong \frac{\text{0x2000} - \text{Offset Register}}{8192} \times 3.125\% \times VREF$$

The gain register value affects the slope of the ADC transfer function as follows:

Gain Register (13 Bits)	Approximate Slope Change
0x1FFF	+3.125%
0x1000	0
0x0000	-3.125%

$$\text{Slope Change} \cong \frac{\text{Gain Register} - \text{0x1000}}{4096} \times 3.125\%$$

Figure 5.21. Offset and Gain Calibration Block Diagram

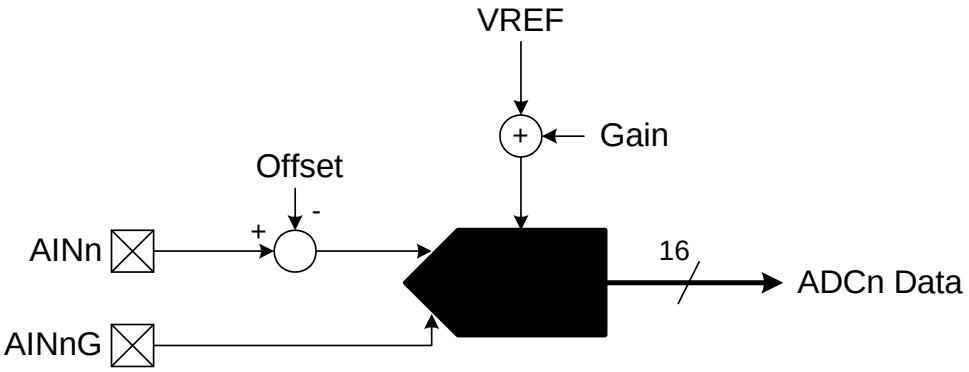


Figure 5.26. ADC0LTH: ADC0 Less-Than Data High Byte Register

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: 0xC7
SFR Page: 0

Bits 7-0: High byte of ADC0 Less-Than Data Word.

Figure 5.27. ADC0LTL: ADC0 Less-Than Data Low Byte Register

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: 0xC6
SFR Page: 0

Bits 7-0: Low byte of ADC0 Less-Than Data Word.

6.6. Interrupt Sources

The DMA contains multiple interrupt sources. Some of these can be individually enabled to generate interrupts as necessary. The DMA Control Register (DMA0CN, Figure 6.4) and DMA Configuration Register (DMA0CF, Figure 6.5) contain the enable bits and flags for the DMA interrupt sources. When an interrupt is enabled and the interrupt condition occurs, a DMA interrupt will be generated (EIE2.7 is set to '1').

The DMA flags that can generate a DMA0 interrupt are:

1. DMA Operations Complete (DMA0CN.6, DMA0INT) occurs when all DMA operations have been completed, and the DMA interface is idle.
2. ADC1 Data Overflow Error (DMA0CN.4, DMA0DE1) occurs when the DMA interface cannot access XRAM for two conversion cycles of ADC1. This flag indicates that at least one conversion result from ADC1 has been discarded.
3. ADC0 Data Overflow Error (DMA0CN.3, DMA0DE0) occurs when the DMA interface cannot access XRAM for two conversion cycles of ADC0. This flag indicates that at least one conversion result from ADC0 has been discarded.
4. ADC1 Data Overflow Warning (DMA0CN.1, DMA0DO1) occurs when data from ADC0 becomes available and the DMA has not yet written the previous results to XRAM. This interrupt source can be enabled and disabled with the Data Overflow Warning Enable bit (DMA0CN.2, DMA0DOE).
5. ADC0 Data Overflow Warning (DMA0CN.0, DMA0DO0) occurs when data from ADC1 becomes available and the DMA has not yet written the previous results to XRAM. This interrupt source can be enabled and disabled with the Data Overflow Warning Enable bit (DMA0CN.2, DMA0DOE).
6. Repeat Counter Overflow (DMA0CF.2, DMA0CI) occurs when the Repeat Counter reaches the Repeat Counter Limit. This interrupt source can be enabled and disabled with the Repeat Counter Overflow Interrupt Enable bit (DMA0CF.3, DMA0CIE).
7. End Of Operation (DMA0CF.0, DMA0EO) occurs when an End Of Operation instruction is reached in the Instruction Buffer. This interrupt source can be enabled and disabled with the End Of Operation Interrupt Enable bit (DMA0CF.1, DMA0EOE).

6.7. Data Buffer Overflow Warnings and Errors

The data paths from the ADCs to XRAM are double-buffered when using the DMA interface. When a conversion is completed by the ADC, it first enters the ADCs data register. If the DMA's data buffer is empty, the conversion results will immediately be written into the DMA's internal data buffer for that ADC. Data in the DMA's internal data buffer is written to XRAM at the first available opportunity (see Section "6.3. XRAM Addressing and Setup" on page 76). Conversion results from the ADC's data registers are not copied into the DMA's data buffer until data in the buffer has been written to XRAM. When a conversion is completed and the DMA's data buffer is not empty, an overflow warning flag is generated. If a second conversion data word becomes available before the DMA's data buffer is written to XRAM, the data in the ADC's data registers is over-written with the new data word, and a data overflow error flag is generated.

7.3. Programmable Window Detector

The ADC Programmable Window Detector continuously compares the ADC2 output registers to user-programmed limits, and notifies the system when a desired condition is detected. This is especially effective in an interrupt-driven system, saving code space and CPU bandwidth while delivering faster system response times. The window detector interrupt flag (AD2WINT in register ADC2CN) can also be used in polled mode. The ADC2 Greater-Than (ADC2GTH, ADC2GTL) and Less-Than (ADC2LTH, ADC2LTL) registers hold the comparison values. The window detector flag can be programmed to indicate when measured data is inside or outside of the user-programmed limits, depending on the contents of the ADC2 Less-Than and ADC2 Greater-Than registers.

Figure 7.11. ADC2GTH: ADC2 Greater-Than Data High Byte Register

SFR Page: 2
SFR Address: 0xC5

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bits7-0: High byte of ADC2 Greater-Than Data Word.

Figure 7.12. ADC2GTL: ADC2 Greater-Than Data Low Byte Register

SFR Page: 2
SFR Address: 0xC4

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bits7-0: Low byte of ADC2 Greater-Than Data Word.

Figure 7.13. ADC2LTH: ADC2 Less-Than Data High Byte Register

SFR Page: 2
SFR Address: 0xC7

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	

Bits7-0: High byte of ADC2 Less-Than Data Word.

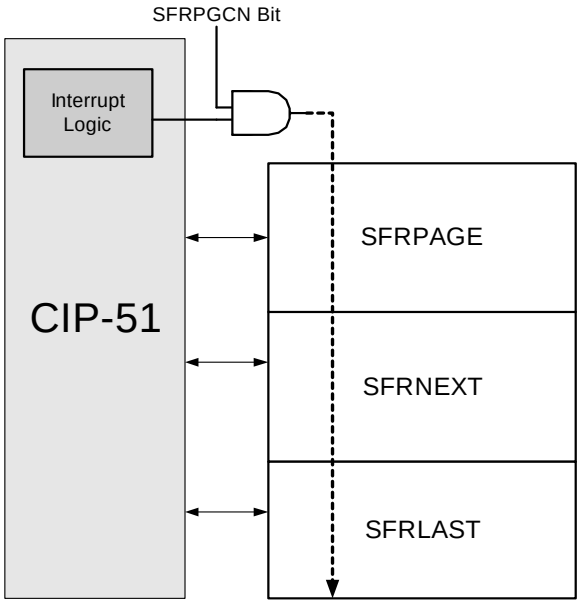
Figure 7.14. ADC2LTL: ADC2 Less-Than Data Low Byte Register

SFR Page: 2
SFR Address: 0xC6

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	

Bits7-0: Low byte of ADC2 Less-Than Data Word.

Figure 13.3. SFR Page Stack



Automatic hardware switching of the SFR Page on interrupts may be enabled or disabled as desired using the SFR Automatic Page Control Enable Bit located in the SFR Page Control Register (SFRPGCN). This function defaults to ‘enabled’ upon reset. In this way, the autoswitching function will be enabled unless disabled in software.

A summary of the SFR locations (address and SFR page) is provided in Table 13.2. in the form of an SFR memory map. Each memory location in the map has an SFR page row, denoting the page in which that SFR resides. Note that certain SFRs are accessible from ALL SFR pages, and are denoted by the “**(ALL PAGES)**” designation. For example, the Port I/O registers P0, P1, P2, and P3 all have the “**(ALL PAGES)**” designation, indicating these SFRs are accessible from all SFR pages regardless of the SFRPAGE register value.

Table 13.2. Special Function Register (SFR) Memory Map

B0	0 1 2 3 F	P3 **(ALL PAGES)**							FLSCL FLACL
A8	0 1 2 3 F	IE **(ALL PAGES)**	SADDR0				P1MDIN	P2MDIN	
A0	0 1 2 3 F	P2 **(ALL PAGES)**	EMI0TC	EMI0CN	EMI0CF	P0MDOUT	P1MDOUT	P2MDOUT	P3MDOUT
98	0 1 2 3 F	SCON0 SCON1	SBUF0 SBUF1	SPI0CFG	SPI0DAT	P4MDOUT	SPI0CKR P5MDOUT	P6MDOUT	P7MDOUT
90	0 1 2 3 F	P1 **(ALL PAGES)**	SSTA0					SFRPGCN	CLKSEL
88	0 1 2 3 F	TCON CPT0CN CPT1CN CPT2CN	TMOD CPT0MD CPT1MD CPT2MD	TL0 OSCICN	TL1 OSCICL	TH0 OSCXCN	TH1	CKCON	PSCTL
80	0 1 2 3 F	P0 **(ALL PAGES)**	SP **(ALL PAGES)**	DPL **(ALL PAGES)**	DPH **(ALL PAGES)**	SFRPAGE **(ALL PAGES)**	SFRNEXT **(ALL PAGES)**	SFRLAST **(ALL PAGES)**	PCON **(ALL PAGES)**
		0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

Table 13.3. Special Function Registers (Continued)

SFRs are listed in alphabetical order. All undefined SFR locations are reserved.

Register	Address	SFR Page	Description	Page No.
P2MDIN	0xAE	F	Port 2 Input Mode	page 217
P2MDOUT	0xA6	F	Port 2 Output Mode Configuration	page 217
P3	0xB0	All Pages	Port 3 Latch	page 218*1
P3MDOUT	0xA7	F	Port 3 Output Mode Configuration	page 218*1
P4	0xC8	F	Port 4 Latch	page 221*1
P4MDOUT	0x9C	F	Port 4 Output Mode Configuration	page 221*1
P5	0xD8	F	Port 5 Latch	page 222*1
P5MDOUT	0x9D	F	Port 5 Output Mode Configuration	page 222*1
P6	0xE8	F	Port 6 Latch	page 223*1
P6MDOUT	0x9E	F	Port 6 Output Mode Configuration	page 223*1
P7	0xF8	F	Port 7 Latch	page 224*1
P7MDOUT	0x9F	F	Port 7 Output Mode Configuration	page 224*1
PCA0CN	0xD8	0	PCA Control	page 312
PCA0CPH0	0xFC	0	PCA Capture 0 High	page 316
PCA0CPH1	0xFE	0	PCA Capture 1 High	page 316
PCA0CPH2	0xEA	0	PCA Capture 2 High	page 316
PCA0CPH3	0xEC	0	PCA Capture 3 High	page 316
PCA0CPH4	0xEE	0	PCA Capture 4 High	page 316
PCA0CPH5	0xE2	0	PCA Capture 5 High	page 316
PCA0CPL0	0xFB	0	PCA Capture 0 Low	page 316
PCA0CPL1	0xFD	0	PCA Capture 1 Low	page 316
PCA0CPL2	0xE9	0	PCA Capture 2 Low	page 316
PCA0CPL3	0xEB	0	PCA Capture 3 Low	page 316
PCA0CPL4	0xED	0	PCA Capture 4 Low	page 316
PCA0CPL5	0xE1	0	PCA Capture 5 Low	page 316
PCA0CPM0	0xDA	0	PCA Module 0 Mode Register	page 314
PCA0CPM1	0xDB	0	PCA Module 1 Mode Register	page 314
PCA0CPM2	0xDC	0	PCA Module 2 Mode Register	page 314
PCA0CPM3	0xDD	0	PCA Module 3 Mode Register	page 314
PCA0CPM4	0xDE	0	PCA Module 4 Mode Register	page 314
PCA0CPM5	0xDF	0	PCA Module 5 Mode Register	page 314
PCA0H	0xFA	0	PCA Counter High	page 315
PCA0L	0xF9	0	PCA Counter Low	page 315
PCA0MD	0xD9	0	PCA Mode	page 313
PCON	0x87	All Pages	Power Control	page 161
PSCTL	0x8F	0	Program Store R/W Control	page 185
PSW	0xD0	All Pages	Program Status Word	page 149
RCAP2H	0xCB	0	Timer/Counter 2 Capture/Reload High	page 301
RCAP2L	0xCA	0	Timer/Counter 2 Capture/Reload Low	page 301
RCAP3H	0xCB	1	Timer/Counter 3 Capture/Reload High	page 301
RCAP3L	0xCA	1	Timer/Counter 3 Capture/Reload Low	page 301
RCAP4H	0xCB	2	Timer/Counter 4 Capture/Reload High	page 301
RCAP4L	0xCA	2	Timer/Counter 4 Capture/Reload Low	page 301

Figure 13.21. EIE1: Extended Interrupt Enable 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EADC0	CP2IE	CP1IE	CP0IE	EPCA0	EWADC0	ESMB0	ESPI0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xE6
SFR Page: All Pages

Bit7: EADC0: Enable ADC0 End of Conversion Interrupt.
This bit sets the masking of the ADC0 End of Conversion Interrupt.
0: Disable ADC0 Conversion Interrupt.
1: Enable interrupt requests generated by the ADC1 Conversion Interrupt.

Bit6: CP2IE: Enable Comparator (CP2) Interrupt.
This bit sets the masking of the CP2 interrupt.
0: Disable CP2 interrupts.
1: Enable interrupt requests generated by the CP2IF flag.

Bit6: CP1IE: Enable Comparator (CP1) Interrupt.
This bit sets the masking of the CP1 interrupt.
0: Disable CP1 interrupts.
1: Enable interrupt requests generated by the CP1IF flag.

Bit6: CP0IE: Enable Comparator (CP0) Interrupt.
This bit sets the masking of the CP0 interrupt.
0: Disable CP0 interrupts.
1: Enable interrupt requests generated by the CP0IF flag.

Bit3: EPCA0: Enable Programmable Counter Array (PCA0) Interrupt.
This bit sets the masking of the PCA0 interrupts.
0: Disable all PCA0 interrupts.
1: Enable interrupt requests generated by PCA0.

Bit2: EWADC0: Enable Window Comparison ADC0 Interrupt.
This bit sets the masking of ADC0 Window Comparison interrupt.
0: Disable ADC0 Window Comparison Interrupt.
1: Enable Interrupt requests generated by ADC0 Window Comparisons.

Bit1: ESMB0: Enable System Management Bus (SMBus0) Interrupt.
This bit sets the masking of the SMBus interrupt.
0: Disable all SMBus interrupts.
1: Enable interrupt requests generated by the SI flag.

Bit0: ESPI0: Enable Serial Peripheral Interface (SPI0) Interrupt.
This bit sets the masking of SPI0 interrupt.
0: Disable all SPI0 interrupts.
1: Enable Interrupt requests generated by the SPI0 flag.

Figure 13.23. EIP1: Extended Interrupt Priority 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
PADC0	PCP2	PCP1	PCP0	PPCA0	PWADC0	PSMB0	PSPI0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xF6
SFR Page: All Pages

Bit7: PADC0: ADC End of Conversion Interrupt Priority Control.
This bit sets the priority of the ADC0 End of Conversion Interrupt.
0: ADC0 End of Conversion interrupt set to low priority level.
1: ADC0 End of Conversion interrupt set to high priority level.

Bit6: PCP2: Comparator2 (CP2) Interrupt Priority Control.
This bit sets the priority of the CP2 interrupt.
0: CP2 interrupt set to low priority level.
1: CP2 interrupt set to high priority level.

Bit5: PCP1: Comparator1 (CP1) Interrupt Priority Control.
This bit sets the priority of the CP1 interrupt.
0: CP1 interrupt set to low priority level.
1: CP1 interrupt set to high priority level.

Bit4: PCP0: Comparator0 (CP0) Interrupt Priority Control.
This bit sets the priority of the CP0 interrupt.
0: CP0 interrupt set to low priority level.
1: CP0 interrupt set to high priority level.

Bit3: PPCA0: Programmable Counter Array (PCA0) Interrupt Priority Control.
This bit sets the priority of the PCA0 interrupt.
0: PCA0 interrupt set to low priority level.
1: PCA0 interrupt set to high priority level.

Bit2: PWADC0: ADC0 Window Comparator Interrupt Priority Control.
This bit sets the priority of the ADC0 Window interrupt.
0: ADC0 Window interrupt set to low priority level.
1: ADC0 Window interrupt set to high priority level.

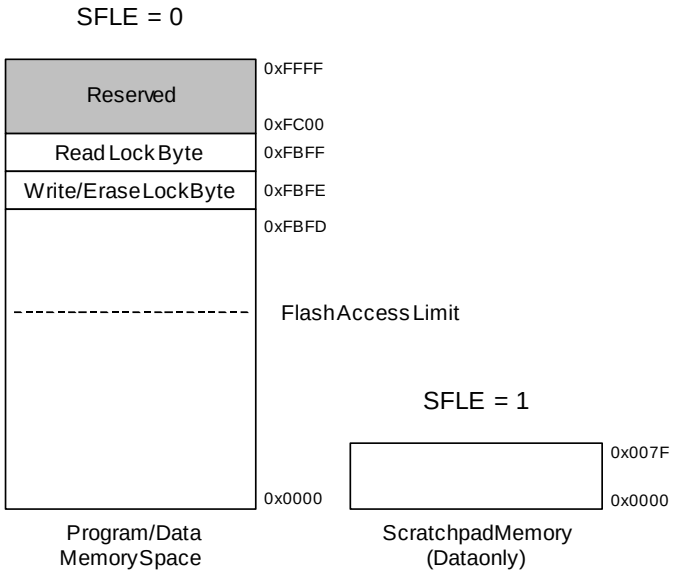
Bit1: PSMB0: System Management Bus (SMBus0) Interrupt Priority Control.
This bit sets the priority of the SMBus0 interrupt.
0: SMBus interrupt set to low priority level.
1: SMBus interrupt set to high priority level.

Bit0: PSPI0: Serial Peripheral Interface (SPI0) Interrupt Priority Control.
This bit sets the priority of the SPI0 interrupt.
0: SPI0 interrupt set to low priority level.
1: SPI0 interrupt set to high priority level.

Figure 16.1. C8051F060/1/2/3/4/5 Flash Program Memory Map and Security Bytes

Read and Write/Erase Security Bits
(Bit 7 is MSB)

Bit	Memory Block
7	0xE000-0xFBFD
6	0xC000-0xDFFF
5	0xA000-0xBFFF
4	0x8000-0x9FFF
3	0x6000-0x7FFF
2	0x4000-0x5FFF
1	0x2000-0x3FFF
0	0x0000-0x1FFF



Flash Read Lock Byte

Bits7-0: Each bit locks a corresponding block of memory. (Bit7 is MSB).

0: Read operations are locked (disabled) for corresponding block across the JTAG interface.

1: Read operations are unlocked (enabled) for corresponding block across the JTAG interface.

Flash Write/Erase Lock Byte

Bits7-0: Each bit locks a corresponding block of memory.

0: Write/Erase operations are locked (disabled) for corresponding block across the JTAG interface.

1: Write/Erase operations are unlocked (enabled) for corresponding block across the JTAG interface.

NOTE: When the block containing the security bytes is locked, the security bytes may be written but not erased.

Flash Access Limit

The Flash Access Limit is defined by the setting of the FLACL register, as described in Figure 16.3. Firmware running at or above this address is prohibited from using the MOVX and MOVC instructions to read, write, or erase Flash locations below this address.

eral's enable bits are not set to a logic 1, then its ports are not accessible at the Port pins of the device. Also note that the Crossbar assigns pins to all associated functions when the SMBus, UART0 or UART1 are selected (i.e. SMBus, SPI, UART). It would be impossible, for example, to assign TX0 to a Port pin without assigning RX0 as well. The SPI can operate in 3 or 4-wire mode (with or without NSS). Each combination of enabled peripherals results in a unique device pinout.

All Port pins on Ports 0 through 3 that are not allocated by the Crossbar can be accessed as General-Purpose I/O (GPIO) pins by reading and writing the associated Port Data registers (See Figure 18.9, Figure 18.11, Figure 18.14, and Figure 18.17), a set of SFRs which are both byte- and bit-addressable. The output states of Port pins that are allocated by the Crossbar are controlled by the digital peripheral that is mapped to those pins. Writes to the Port Data registers (or associated Port bits) will have no effect on the states of these pins.

A Read of a Port Data register (or Port bit) will always return the logic state present at the pin itself, regardless of whether the Crossbar has allocated the pin for peripheral use or not. An exception to this occurs during the execution of a *read-modify-write* instruction (ANL, ORL, XRL, CPL, INC, DEC, DJNZ, JBC, CLR, SETB, and the bitwise MOV write operation). During the *read* cycle of the *read-modify-write* instruction, it is the contents of the Port Data register, not the state of the Port pins themselves, which is read.

Because the Crossbar registers affect the pinout of the peripherals of the device, they are typically configured in the initialization code of the system before the peripherals themselves are configured. Once configured, the Crossbar registers are typically left alone.

Once the Crossbar registers have been properly configured, the Crossbar is enabled by setting XBARE (XBR2.4) to a logic 1. **Until XBARE is set to a logic 1, the output drivers on Ports 0 through 3 are explicitly disabled in order to prevent possible contention on the Port pins while the Crossbar registers and other registers which can affect the device pinout are being written**.

The output drivers on Crossbar-assigned input signals (like RX0, for example) are explicitly disabled; thus the values of the Port Data registers and the PnMDOUT registers have no effect on the states of these pins.

18.1.2. Configuring the Output Modes of the Port Pins

The output drivers on Ports 0 through 3 remain disabled until the Crossbar is enabled by setting XBARE (XBR2.4) to a logic 1.

The output mode of each port pin can be configured to be either Open-Drain or Push-Pull. In the Push-Pull configuration, writing a logic 0 to the associated bit in the Port Data register will cause the Port pin to be driven to GND, and writing a logic 1 will cause the Port pin to be driven to VDD. In the Open-Drain configuration, writing a logic 0 to the associated bit in the Port Data register will cause the Port pin to be driven to GND, and a logic 1 will cause the Port pin to assume a high-impedance state. The Open-Drain configuration is useful to prevent contention between devices in systems where the Port pin participates in a shared interconnection in which multiple outputs are connected to the same physical wire (like the SDA signal on an SMBus connection).

The output modes of the Port pins on Ports 0 through 3 are determined by the bits in the associated PnMDOUT registers (See Figure 18.10, Figure 18.13, Figure 18.16, and Figure 18.18). For example, a logic 1 in P3MDOUT.7 will configure the output mode of P3.7 to Push-Pull; a logic 0 in P3MDOUT.7 will configure the output mode of P3.7 to Open-Drain. All Port pins default to Open-Drain output.

19. Controller Area Network (CAN0, C8051F060/1/2/3)

IMPORTANT DOCUMENTATION NOTE: *The Bosch CAN Controller is integrated in the C8051F060/1/2/3 devices. This section of the data sheet gives a description of the CAN controller as an overview and offers a description of how the Silicon Labs CIP-51 MCU interfaces with the on-chip Bosch CAN controller. In order to use the CAN controller, please refer to Bosch's C_CAN User's Manual (revision 1.2) as an accompanying manual to Silicon Labs' C8051F060/1/2/3/4/5/6/7 Data sheet.*

The C8051F060/1/2/3 family of devices feature a Control Area Network (CAN) controller that enables serial communication using the CAN protocol. Silicon Labs CAN controller facilitates communication on a CAN network in accordance with the Bosch specification 2.0A (basic CAN) and 2.0B (full CAN). The CAN controller consists of a CAN Core, Message RAM (separate from the CIP-51 RAM), a message handler state machine, and control registers. Silicon Labs CAN is a protocol controller and does not provide physical layer drivers (i.e., transceivers). Figure 19.2 shows an example typical configuration on a CAN bus.

Silicon Labs CAN operates at bit rates of up to 1 Mbit/second, though this can be limited by the physical layer chosen to transmit data on the CAN bus. The CAN processor has 32 Message Objects that can be configured to transmit or receive data. Incoming data, message objects and their identifier masks are stored in the CAN message RAM. All protocol functions for transmission of data and acceptance filtering is performed by the CAN controller and not by the CIP-51 MCU. In this way, minimal CPU bandwidth is needed to use CAN communication. The CIP-51 configures the CAN controller, accesses received data, and passes data for transmission via Special Function Registers (SFR) in the CIP-51. The CAN controller's clock (f_{sys}, or CAN_CLK in the C_CAN User's Guide) is equal to the CIP-51 MCU's clock (SYSCLK).

24.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only)

In Mode 3, Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. The counter/timer in TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0 and TF0. TL0 can use either the system clock or an external input signal as its timebase. The TH0 register is restricted to a timer function sourced by the system clock or prescaled clock. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3. When Timer 0 is operating in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but cannot be clocked by external signals nor set the TF1 flag and generate an interrupt. However, the Timer 1 overflow can be used to generate baud rates for the SMBus and/or UART, and/or initiate ADC conversions. While Timer 0 is operating in Mode 3, Timer 1 run control is handled through its mode settings. To run Timer 1 while Timer 0 is in Mode 3, set the Timer 1 Mode as 0, 1, or 2. To disable Timer 1, configure it for Mode 3.

Figure 24.3. T0 Mode 3 Block Diagram

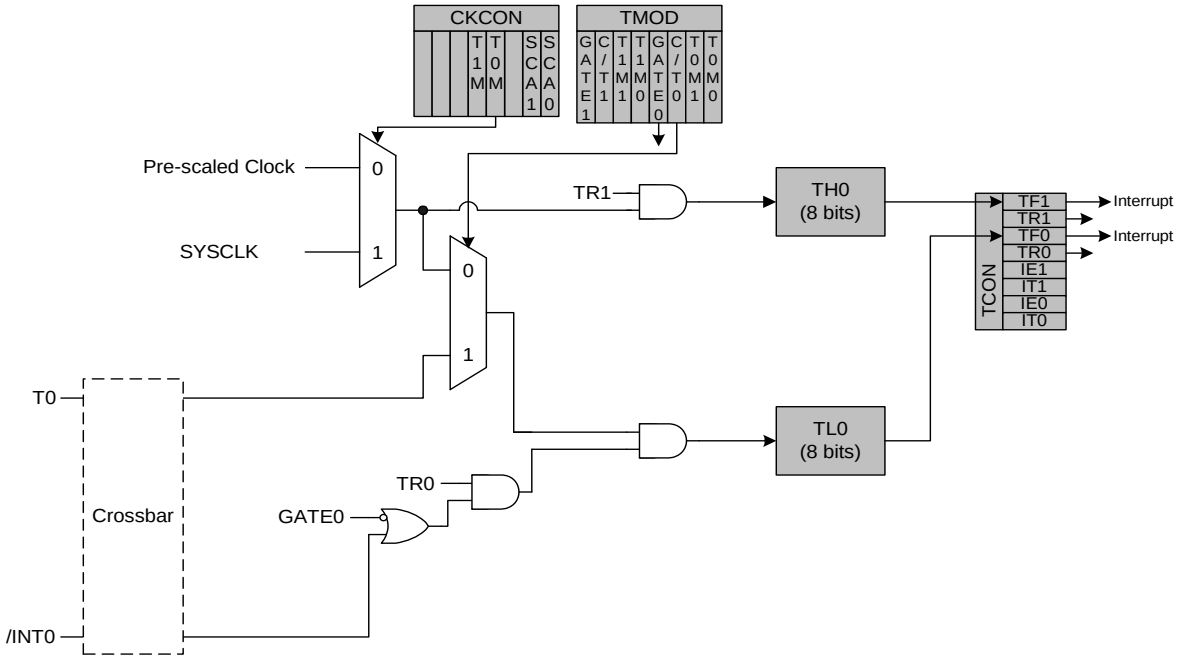


Figure 24.18. TMRnH: Timer 2, 3, and 4 High Byte

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: TMR2H: 0xCD; TMR3H: 0xCD; TMR4H: 0xCD

SFR Page: TMR2H: page 0; TMR3H: page 1; TMR4H: page 2

Bits 7-0: TH2, 3, and 4: Timer 2, 3, and 4 High Byte.
The TH2, 3, and 4 register contains the high byte of the 16-bit Timer 2, 3, and 4

25.3. Register Descriptions for PCA0

Following are detailed descriptions of the special function registers related to the operation of PCA0.

Figure 25.10. PCA0CN: PCA Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CF	CR	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xD8
SFR Page: 0

Bit7: CF: PCA Counter/Timer Overflow Flag.
Set by hardware when the PCA0 Counter/Timer overflows from 0xFFFF to 0x0000. When the Counter/Timer Overflow (CF) interrupt is enabled, setting this bit causes the CPU to vector to the CF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit6: CR: PCA0 Counter/Timer Run Control.
This bit enables/disables the PCA0 Counter/Timer.
0: PCA0 Counter/Timer disabled.
1: PCA0 Counter/Timer enabled.

Bit5: CCF5: PCA0 Module 5 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit4: CCF4: PCA0 Module 4 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit3: CCF3: PCA0 Module 3 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit2: CCF2: PCA0 Module 2 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit1: CCF1: PCA0 Module 1 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit0: CCF0: PCA0 Module 0 Capture/Compare Flag.
This bit is set by hardware when a match or capture occurs. When the CCF interrupt is enabled, setting this bit causes the CPU to vector to the CCF interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.