



Welcome to [E-XFL.COM](https://www.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	50MHz
Connectivity	SMBus (2-Wire/I ² C), SPI, UART/USART
Peripherals	POR, PWM, Temp Sensor, WDT
Number of I/O	25
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8.25K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.25V
Data Converters	A/D 25x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-QFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f583-im

C8051F58x/F59x

12.2.1.2. Bit Addressable Locations	105
12.2.1.3. Stack	105
13. Special Function Registers.....	106
13.1. SFR Paging	106
13.2. Interrupts and SFR Paging	106
13.3. SFR Page Stack Example	107
14. Interrupts	126
14.1. MCU Interrupt Sources and Vectors.....	126
14.1.1. Interrupt Priorities.....	127
14.1.2. Interrupt Latency	127
14.2. Interrupt Register Descriptions	129
14.3. External Interrupts INT0 and INT1.....	136
15. Flash Memory	138
15.1. Programming The Flash Memory	138
15.1.1. Flash Lock and Key Functions	138
15.1.2. Flash Erase Procedure	138
15.1.3. Flash Write Procedure	139
15.1.4. Flash Write Optimization	139
15.2. Non-volatile Data Storage	140
15.3. Security Options	140
15.4. Flash Write and Erase Guidelines.....	142
15.4.1. V _{DD} Maintenance and the V _{DD} monitor	142
15.4.2. PSWE Maintenance	142
15.4.3. System Clock	143
16. Power Management Modes.....	147
16.1. Idle Mode.....	147
16.2. Stop Mode	148
16.3. Suspend Mode	148
17. Reset Sources	150
17.1. Power-On Reset.....	151
17.2. Power-Fail Reset/VDD Monitor	152
17.3. External Reset.....	153
17.4. Missing Clock Detector Reset	153
17.5. Comparator0 Reset	154
17.6. PCA Watchdog Timer Reset	154
17.7. Flash Error Reset	154
17.8. Software Reset	154
18. External Data Memory Interface and On-Chip XRAM	156
18.1. Accessing XRAM.....	156
18.1.1. 16-Bit MOVX Example	156
18.1.2. 8-Bit MOVX Example	156
18.2. Configuring the External Memory Interface	157
18.3. Port Configuration.....	157
18.4. Multiplexed and Non-multiplexed Selection.....	162
18.4.1. Multiplexed Configuration.....	162

C8051F58x/F59x

Figure 13.4. SFR Page Stack Upon PCA Interrupt Occurring During a CAN0 ISR	110
Figure 13.5. SFR Page Stack Upon Return From PCA Interrupt	111
Figure 13.6. SFR Page Stack Upon Return From CAN0 Interrupt	112
Figure 15.1. Flash Program Memory Map	141
Figure 17.1. Reset Sources	152
Figure 17.2. Power-On and VDD Monitor Reset Timing	153
Figure 19.1. Oscillator Options	176
Figure 19.2. Example Clock Multiplier Output	181
Figure 19.3. External 32.768 kHz Quartz Crystal Oscillator Connection Diagram	186
Figure 20.1. Port I/O Functional Block Diagram	189
Figure 20.2. Port I/O Cell Block Diagram	190
Figure 20.3. Peripheral Availability on Port I/O Pins	193
Figure 20.4. Crossbar Priority Decoder in Example Configuration	194
Figure 21.1. LIN Block Diagram	214
Figure 22.1. Typical CAN Bus Configuration	231
Figure 22.2. CAN Controller Diagram	232
Figure 22.3. Four segments of a CAN Bit	234
Figure 23.1. SMBus Block Diagram	239
Figure 23.2. Typical SMBus Configuration	240
Figure 23.3. SMBus Transaction	241
Figure 23.4. Typical SMBus SCL Generation	243
Figure 23.5. Typical Master Write Sequence	250
Figure 23.6. Typical Master Read Sequence	251
Figure 23.7. Typical Slave Write Sequence	252
Figure 23.8. Typical Slave Read Sequence	253
Figure 24.1. UART0 Block Diagram	256
Figure 24.2. UART0 Timing Without Parity or Extra Bit	258
Figure 24.3. UART0 Timing With Parity	258
Figure 24.4. UART0 Timing With Extra Bit	258
Figure 24.5. Typical UART Interconnect Diagram	259
Figure 24.6. UART Multi-Processor Mode Interconnect Diagram	260
Figure 26.1. SPI Block Diagram	272
Figure 26.2. Multiple-Master Mode Connection Diagram	275
Figure 26.3. 3-Wire Single Master and 3-Wire Single Slave Mode Connection Diagram	275
Figure 26.4. 4-Wire Single Master Mode and 4-Wire Slave Mode Connection Diagram	275
Figure 26.5. Master Mode Data/Clock Timing	277
Figure 26.6. Slave Mode Data/Clock Timing (CKPHA = 0)	278
Figure 26.7. Slave Mode Data/Clock Timing (CKPHA = 1)	278
Figure 26.8. SPI Master Timing (CKPHA = 0)	282
Figure 26.9. SPI Master Timing (CKPHA = 1)	282
Figure 26.10. SPI Slave Timing (CKPHA = 0)	283
Figure 26.11. SPI Slave Timing (CKPHA = 1)	283
Figure 27.1. T0 Mode 0 Block Diagram	288

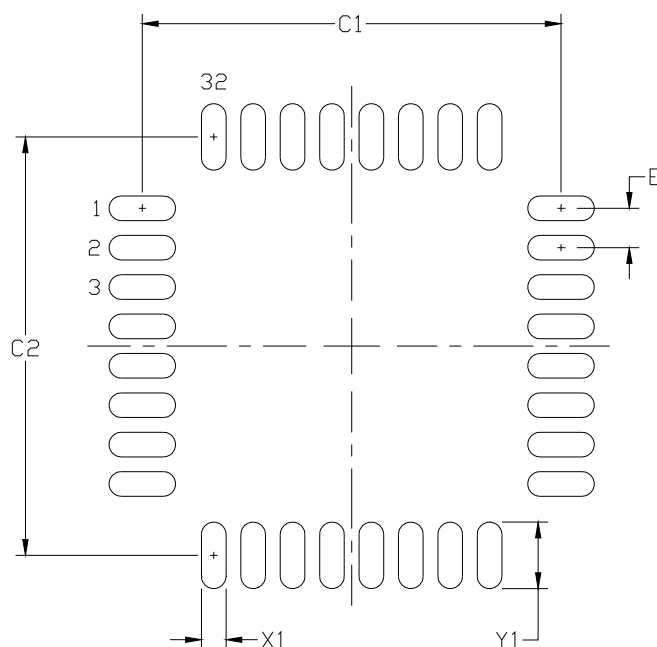


Figure 4.8. QFP-32 Package Drawing

Table 4.8. QFP-32 Landing Diagram Dimensions

Dimension	Min	Max	Dimension	Min	Max
C1	8.40	8.50	X1	0.40	0.50
C2	8.40	8.50	Y1	1.25	1.35
E	0.80 BSC				

Notes:

General

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. This Land Pattern Design is based on the IPC-7351 guidelines.

Solder Mask Design

3. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60µm minimum, all the way around the pad.

Stencil Design

4. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
5. The stencil thickness should be 0.125mm (5 mils).
6. The ratio of stencil aperture to land pad size should be 1:1 for all perimeter pads.

Card Assembly

7. A No-Clean, Type-3 solder paste is recommended.
8. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

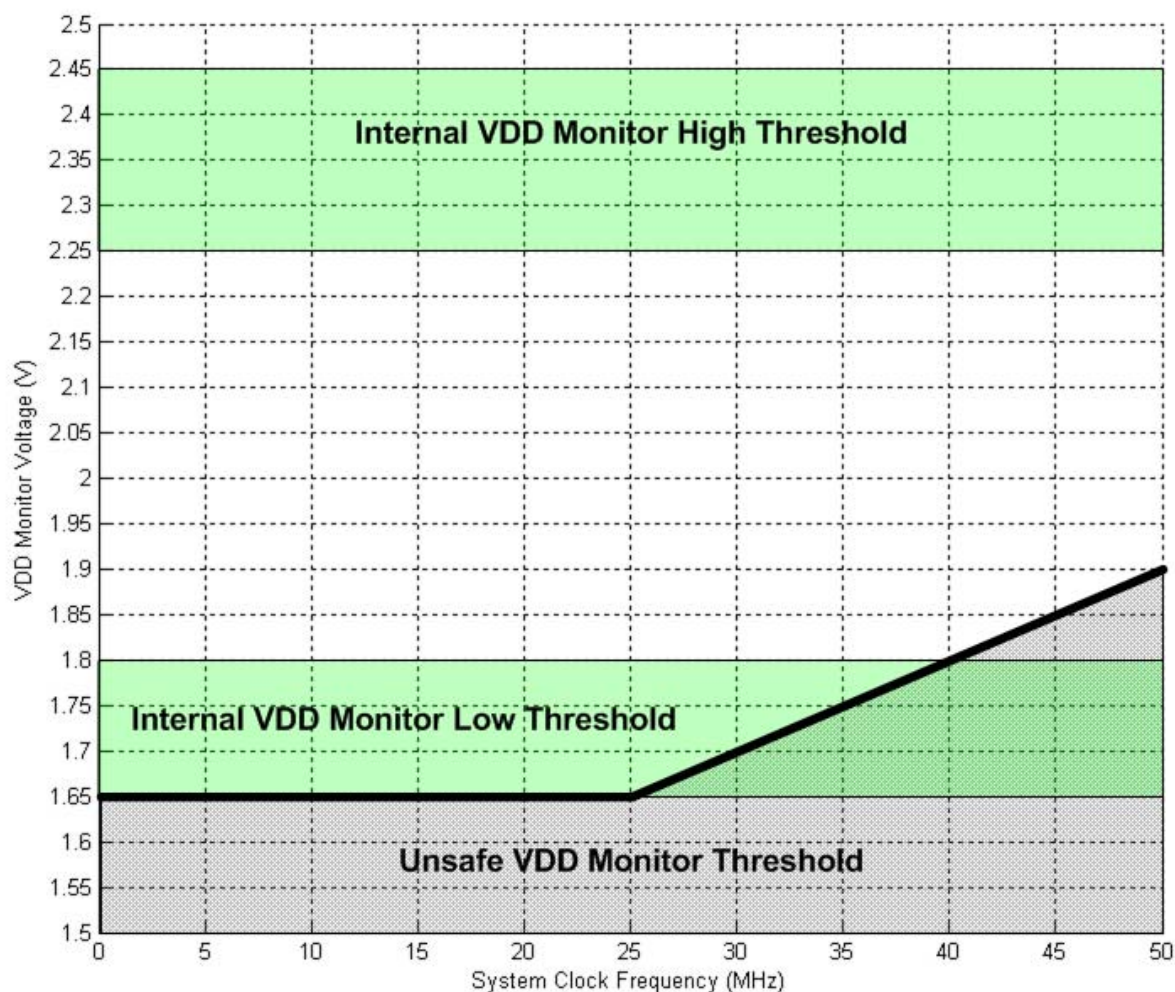


Figure 5.1. Maximum System Clock Frequency vs. VDD Voltage

Note: With system clock frequencies greater than 25 MHz, the V_{DD} monitor level should be set to the high threshold ($V_{DMLVL} = 1b$ in SFR $VDM0CN$) to prevent undefined CPU operation. The high threshold should only be used with an external regulator powering V_{DD} directly. See Figure 10.2 on page 90 for the recommended power supply connections.

6.2. Output Code Formatting

The registers ADC0H and ADC0L contain the high and low bytes of the output conversion code. When the repeat count is set to 1, conversion codes are represented in 12-bit unsigned integer format and the output conversion code is updated after each conversion. Inputs are measured from 0 to $V_{REF} \times 4095/4096$. Data can be right-justified or left-justified, depending on the setting of the AD0LJST bit (ADC0CN.2). Unused bits in the ADC0H and ADC0L registers are set to 0. Example codes are shown below for both right-justified and left-justified data.

Input Voltage	Right-Justified ADC0H:ADC0L (AD0LJST = 0)	Left-Justified ADC0H:ADC0L (AD0LJST = 1)
$V_{REF} \times 4095/4096$	0x0FFF	0xFFFF0
$V_{REF} \times 2048/4096$	0x0800	0x8000
$V_{REF} \times 2047/4096$	0x07FF	0x7FF0
0	0x0000	0x0000

When the ADC0 Repeat Count is greater than 1, the output conversion code represents the accumulated result of the conversions performed and is updated after the last conversion in the series is finished. Sets of 4, 8, or 16 consecutive samples can be accumulated and represented in unsigned integer format. The repeat count can be selected using the AD0RPT bits in the ADC0CF register. The value must be right-justified (AD0LJST = 0), and unused bits in the ADC0H and ADC0L registers are set to 0. The following example shows right-justified codes for repeat counts greater than 1. Notice that accumulating 2^n samples is equivalent to left-shifting by n bit positions when all samples returned from the ADC have the same value.

Input Voltage	Repeat Count = 4	Repeat Count = 8	Repeat Count = 16
$V_{REF} \times 4095/4096$	0x3FFC	0x7FF8	0xFFFF0
$V_{REF} \times 2048/4096$	0x2000	0x4000	0x8000
$V_{REF} \times 2047/4096$	0x1FFC	0x3FF8	0x7FF0
0	0x0000	0x0000	0x0000

6.2.1. Settling Time Requirements

A minimum tracking time is required before an accurate conversion is performed. This tracking time is determined by any series impedance, including the AMUX0 resistance, the ADC0 sampling capacitance, and the accuracy required for the conversion.

Figure 6.5 shows the equivalent ADC0 input circuit. The required ADC0 settling time for a given settling accuracy (SA) may be approximated by Equation 6.1. When measuring the Temperature Sensor output, use the tracking time specified in Table 5.11 on page 52. When measuring V_{DD} with respect to GND, R_{TOTAL} reduces to R_{MUX} . See Table 5.10 for ADC0 minimum settling time requirements as well as the mux impedance and sampling capacitor values.

$$t = \ln\left(\frac{2^n}{SA}\right) \times R_{TOTAL} C_{SAMPLE}$$

Equation 6.1. ADC0 Settling Time Requirements

Where:

SA is the settling accuracy, given as a fraction of an LSB (for example, 0.25 to settle within 1/4 LSB)

t is the required settling time in seconds

R_{TOTAL} is the sum of the AMUX0 resistance and any external source resistance.

n is the ADC resolution in bits (10).

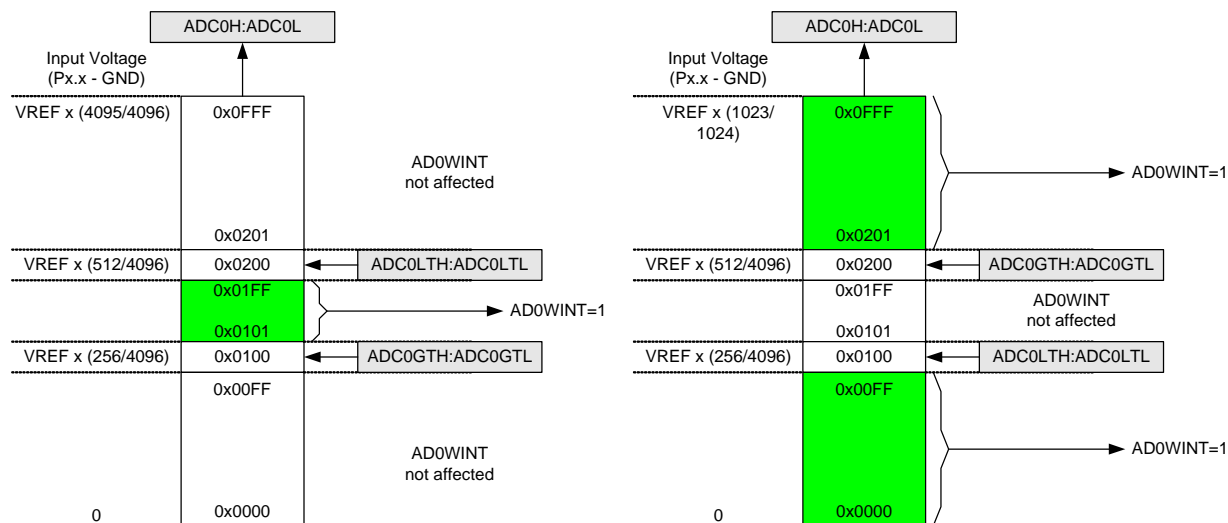


Figure 6.6. ADC Window Compare Example: Right-Justified Data

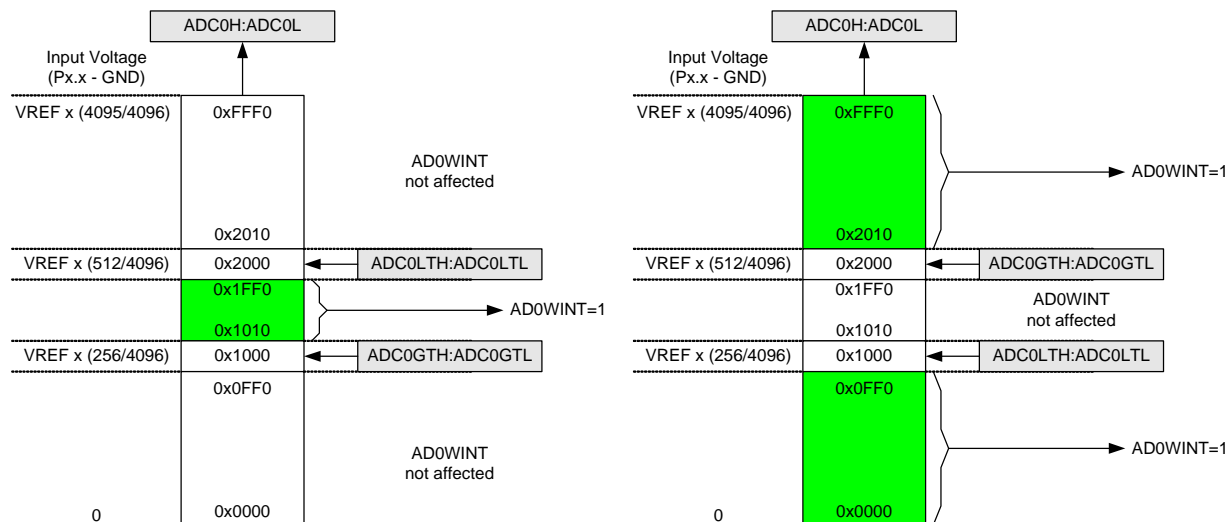


Figure 6.7. ADC Window Compare Example: Left-Justified Data

C8051F58x/F59x

SFR Definition 9.2. CPT0MD: Comparator0 Mode Selection

Bit	7	6	5	4	3	2	1	0
Name			CP0RIE	CP0FIE			CP0MD[1:0]	
Type	R	R	R/W	R/W	R	R	R/W	
Reset	0	0	0	0	0	0	1	0

SFR Address = 0x9B; SFR Page = 0x00

Bit	Name	Function
7:6	Unused	Read = 00b, Write = Don't Care.
5	CP0RIE	Comparator0 Rising-Edge Interrupt Enable. 0: Comparator0 Rising-edge interrupt disabled. 1: Comparator0 Rising-edge interrupt enabled.
4	CP0FIE	Comparator0 Falling-Edge Interrupt Enable. 0: Comparator0 Falling-edge interrupt disabled. 1: Comparator0 Falling-edge interrupt enabled.
3:2	Unused	Read = 00b, Write = don't care.
1:0	CP0MD[1:0]	Comparator0 Mode Select. These bits affect the response time and power consumption for Comparator0. 00: Mode 0 (Fastest Response Time, Highest Power Consumption) 01: Mode 1 10: Mode 2 11: Mode 3 (Slowest Response Time, Lowest Power Consumption)

SFR Definition 9.8. CPT1MX: Comparator1 MUX Selection

Bit	7	6	5	4	3	2	1	0
Name	CMX1N[3:0]				CMX1P[3:0]			
Type	R/W				R/W			
Reset	0	1	1	1	0	1	1	1

SFR Address = 0x9F; SFR Page = 0x00

Bit	Name	Function
7:4	CMX1N[3:0]	Comparator1 Negative Input MUX Selection. 0000: P0.1 0001: P0.3 0010: P0.5 0011: P0.7 0100: P1.1 0101: P1.3 0110: P1.5 0111: P1.7 1000: P2.1 1001: P2.3 1010: P2.5 1011: P2.7 1100–1111: None
3:0	CMX1P[3:0]	Comparator1 Positive Input MUX Selection. 0000: P0.0 0001: P0.2 0010: P0.4 0011: P0.6 0100: P1.0 0101: P1.2 0110: P1.4 0111: P1.6 1000: P2.0 1001: P2.2 1010: P2.4 1011: P2.6 1100–1111: None

SFR Definition 11.3. SP: Stack Pointer

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

SFR Address = 0x81; SFR Page = All Pages

Bit	Name	Function
7:0	SP[7:0]	Stack Pointer. The Stack Pointer holds the location of the top of the stack. The stack pointer is incremented before every PUSH operation. The SP register defaults to 0x07 after reset.

SFR Definition 11.4. ACC: Accumulator

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

SFR Address = 0xE0; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7:0	ACC[7:0]	Accumulator. This register is the accumulator for arithmetic operations.

SFR Definition 11.5. B: B Register

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

SFR Address = 0xF0; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7:0	B[7:0]	B Register. This register serves as a second accumulator for certain arithmetic operations.

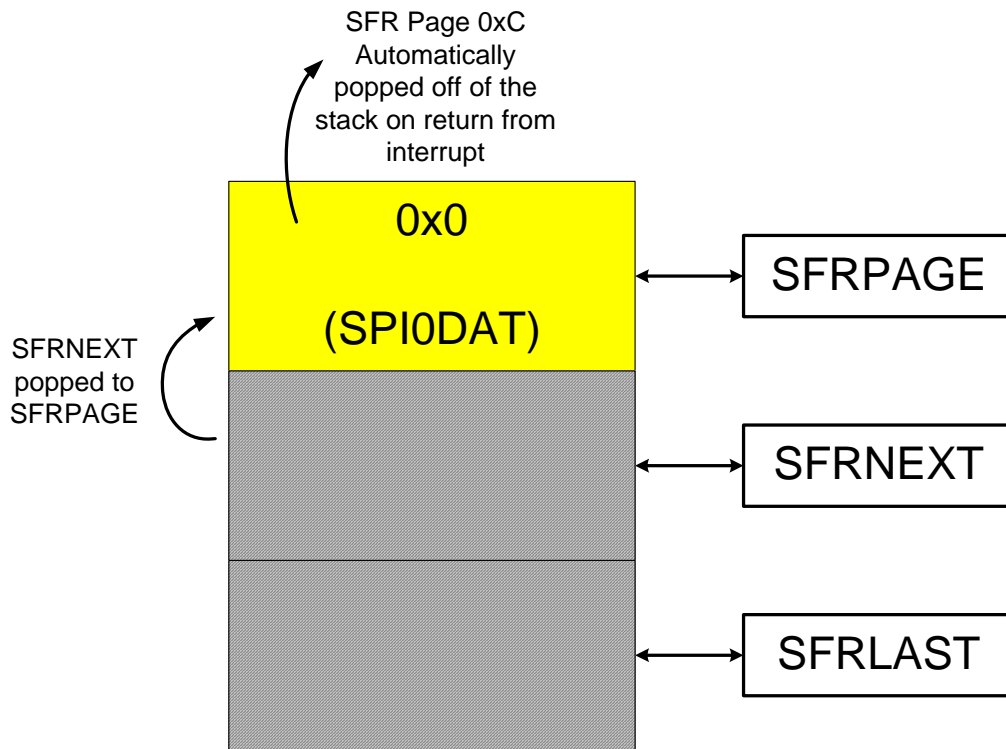


Figure 13.6. SFR Page Stack Upon Return From CAN0 Interrupt

In the example above, all three bytes in the SFR Page Stack are accessible via the SFRPAGE, SFRNEXT, and SFRLAST special function registers. If the stack is altered while servicing an interrupt, it is possible to return to a different SFR Page upon interrupt exit than selected prior to the interrupt call. Direct access to the SFR Page stack can be useful to enable real-time operating systems to control and manage context switching between multiple tasks.

Push operations on the SFR Page Stack only occur on interrupt service, and pop operations only occur on interrupt exit (execution on the RETI instruction). The automatic switching of the SFRPAGE and operation of the SFR Page Stack as described above can be disabled in software by clearing the SFR Automatic Page Enable Bit (SFRPGEN) in the SFR Page Control Register (SFR0CN). See SFR Definition 13.1.

14. Interrupts

The C8051F58x/F59x devices include an extended interrupt system supporting a total of 23 interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external inputs pins varies according to the specific version of the device. Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt-pending flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a pre-determined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. (The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.)

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE, EIE1, or EIE2). However, interrupts must first be globally enabled by setting the EA bit (IE.7) to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings.

Note: Any instruction that clears a bit to disable an interrupt should be immediately followed by an instruction that has two or more opcode bytes. Using EA (global interrupt enable) as an example:

```
// in 'C':
EA = 0; // clear EA bit.
EA = 0; // this is a dummy instruction with two-byte opcode.

; in assembly:
CLR EA ; clear EA bit.
CLR EA ; this is a dummy instruction with two-byte opcode.
```

For example, if an interrupt is posted during the execution phase of a "CLR EA" opcode (or any instruction which clears a bit to disable an interrupt source), and the instruction is followed by a single-cycle instruction, the interrupt may be taken. However, a read of the enable bit will return a 0 inside the interrupt service routine. When the bit-clearing opcode is followed by a multi-cycle instruction, the interrupt will not be taken.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

14.1. MCU Interrupt Sources and Vectors

The C8051F58x/F59x MCUs support 23 interrupt sources. Software can simulate an interrupt by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 14.1. Refer to the datasheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

Table 18.2. EMIF Pinout (C8051F588/9-F590/1)

Multiplexed Mode	
Signal Name	Port Pin
$\overline{\text{RD}}$	P1.6
$\overline{\text{WR}}$	P1.7
ALE	P1.5
D0/A0	P3.0
D1/A1	P3.1
D2/A2	P3.2
D3/A3	P3.3
D4/A4	P3.4
D5/A5	P3.5
D6/A6	P3.6
D7/A7	P3.7
A8	P2.0
A9	P2.1
A10	P2.2
A11	P2.3
A12	P2.4
A13	P2.5
A14	P2.6
A15	P2.7

18.4. Multiplexed and Non-multiplexed Selection

The External Memory Interface is capable of acting in a Multiplexed mode or a Non-multiplexed mode, depending on the state of the EMD2 (EMI0CF.4) bit.

18.4.1. Multiplexed Configuration

In Multiplexed mode, the Data Bus and the lower 8-bits of the Address Bus share the same Port pins: AD[7:0]. In this mode, an external latch (74HC373 or equivalent logic gate) is used to hold the lower 8-bits of the RAM address. The external latch is controlled by the ALE (Address Latch Enable) signal, which is driven by the External Memory Interface logic. An example of a Multiplexed Configuration is shown in Figure 18.1.

In Multiplexed mode, the external MOVX operation can be broken into two phases delineated by the state of the ALE signal. During the first phase, ALE is high and the lower 8-bits of the Address Bus are presented to AD[7:0]. During this phase, the address latch is configured such that the Q outputs reflect the states of the 'D' inputs. When ALE falls, signaling the beginning of the second phase, the address latch outputs remain fixed and are no longer dependent on the latch inputs. Later in the second phase, the Data Bus controls the state of the AD[7:0] port at the time \overline{RD} or \overline{WE} is asserted.

See Section "18.6.2. Multiplexed Mode" on page 172 for more information.

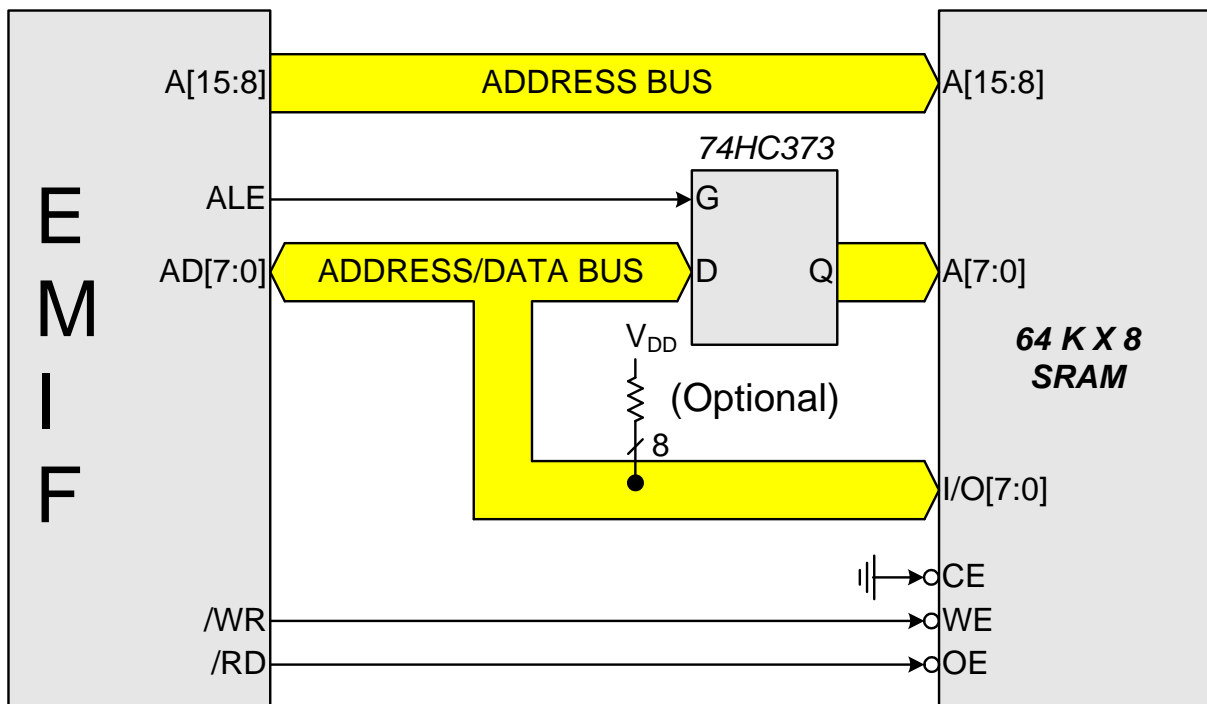


Figure 18.1. Multiplexed Configuration Example

18.6.2.3. 8-bit MOVX with Bank Select: EMI0CF[4:2] = 010

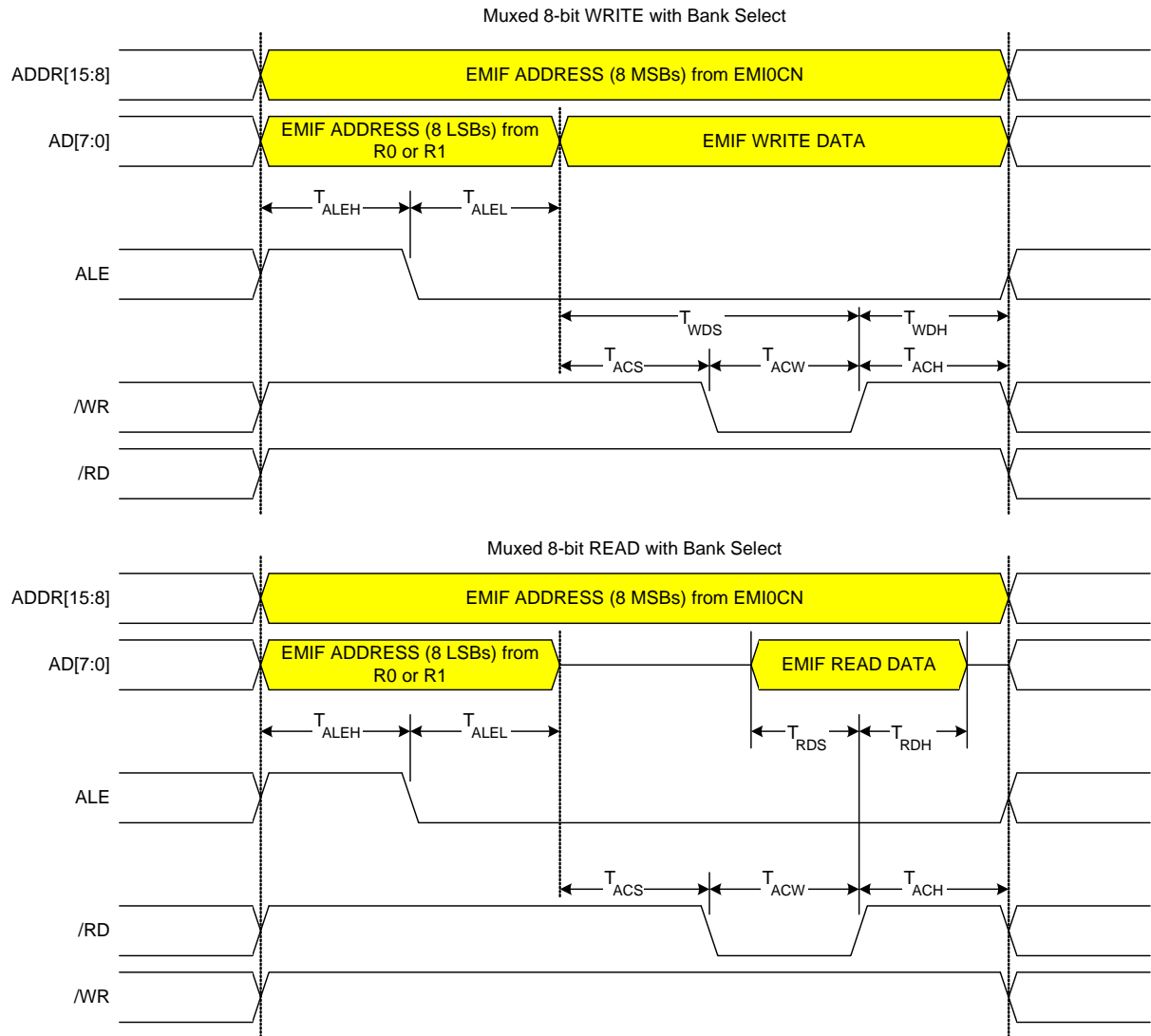


Figure 18.9. Multiplexed 8-bit MOVX with Bank Select Timing

20.1. Port I/O Modes of Operation

Port pins P0.0–P3.7 use the Port I/O cell shown in Figure 20.2. Each of these Port I/O cells can be configured by software for analog I/O or digital I/O using the PnMDIN registers. P4.0–P4.7 use a similar cell, except that they can only be configured as digital I/O pins and do not have a corresponding PnMDIN or PnSKIP register. On reset, all Port I/O cells default to a high impedance state with weak pull-ups enabled until the Crossbar is enabled (XBARE = 1).

20.1.1. Port Pins Configured for Analog I/O

Any pins to be used as Comparator or ADC inputs, external oscillator inputs, or VREF should be configured for analog I/O (PnMDIN.n = 0). When a pin is configured for analog I/O, its weak pullup, digital driver, and digital receiver are disabled. Port pins configured for analog I/O will always read back a value of 0.

Configuring pins as analog I/O saves power and isolates the Port pin from digital interference. Port pins configured as digital inputs may still be used by analog peripherals; however, this practice is not recommended and may result in measurement errors.

20.1.2. Port Pins Configured For Digital I/O

Any pins to be used by digital peripherals (UART, SPI, SMBus, etc.), external digital event capture functions, or as GPIO should be configured as digital I/O (PnMDIN.n = 1). For digital I/O pins, one of two output modes (push-pull or open-drain) must be selected using the PnMDOUT registers.

Push-pull outputs (PnMDOUT.n = 1) drive the Port pad to the VIO or GND supply rails based on the output logic value of the Port pin. Open-drain outputs have the high side driver disabled; therefore, they only drive the Port pad to GND when the output logic value is 0 and become high impedance inputs (both high low drivers turned off) when the output logic value is 1.

When a digital I/O cell is placed in the high impedance state, a weak pull-up transistor pulls the Port pad to the VIO supply voltage to ensure the digital input is at a defined logic state. Weak pull-ups are disabled when the I/O cell is driven to GND to minimize power consumption and may be globally disabled by setting WEAKPUD to 1. The user should ensure that digital I/O are always internally or externally pulled or driven to a valid logic state to minimize power consumption. Port pins configured for digital I/O always read back the logic state of the Port pad, regardless of the output logic value of the Port pin.

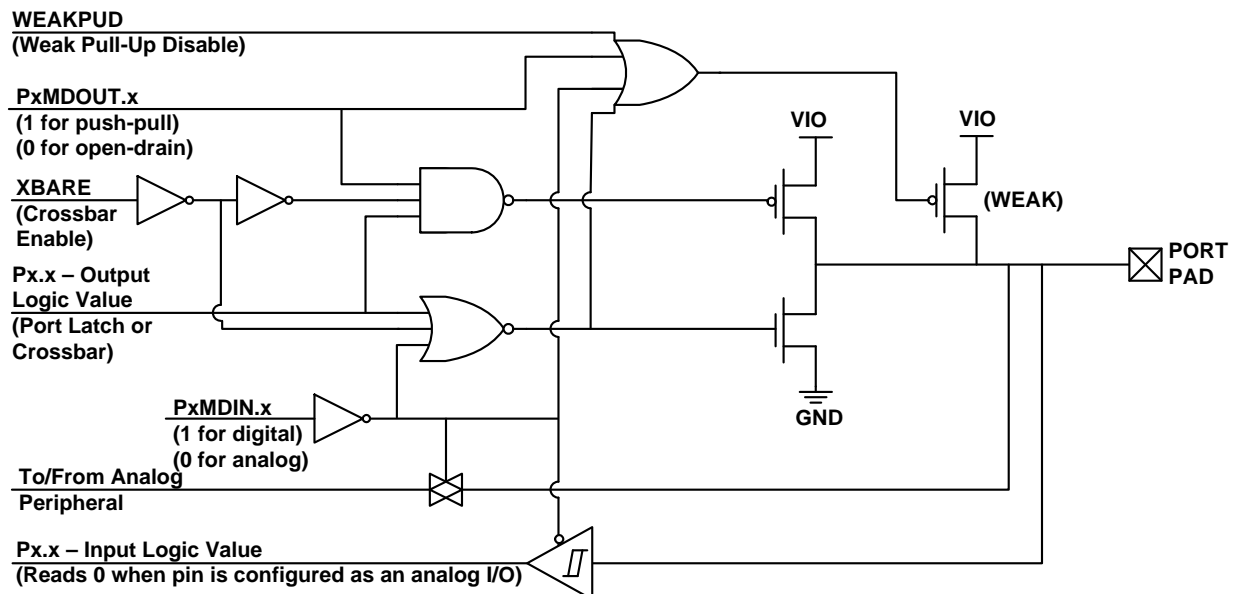


Figure 20.2. Port I/O Cell Block Diagram

SFR Definition 20.28. P3SKIP: Port 3Skip

Bit	7	6	5	4	3	2	1	0
Name	P3SKIP[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD7; SFR Page = 0x0F

Bit	Name	Function
7:0	P3SKIP[7:0]	Port 3 Crossbar Skip Enable Bits. These bits select Port 3 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P3.n pin is not skipped by the Crossbar. 1: Corresponding P3.n pin is skipped by the Crossbar.
Note: Port P3.1–P3.7 are only available on the 48-pin and 40-pin packages.		

SFR Definition 20.29. P4: Port 4

Bit	7	6	5	4	3	2	1	0
Name	P4[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xB5; SFR Page = All Pages

Bit	Name	Description	Write	Read
7:0	P4[7:0]	Port 4 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: P4.n Port pin is logic LOW. 1: P4.n Port pin is logic HIGH.
Note: Port 4.0 is only available on the 48-pin and 40-pin packages.; P4.1-P4.7 is only available on the 48-pin packages.				

22.2.4. CAN Register Assignment

The standard Bosch CAN registers are mapped to SFR space as shown below and their full definitions are available in the CAN User's Guide. The name shown in the Name column matches what is provided in the CAN User's Guide. One additional SFR which is not a standard Bosch CAN register, CAN0CFG, is provided to configure the CAN clock. All CAN registers are located on SFR Page 0x0C.

Table 22.2. Standard CAN Registers and Reset Values

CAN Addr.	Name	SFR Name (High)	SFR Addr.	SFR Name (Low)	SFR Addr.	16-bit SFR	Reset Value
0x00	CAN Control Register	—	—	CAN0CN	0xC0	—	0x01
0x02	Status Register	—	—	CAN0STAT	0x94	—	0x00
0x04	Error Counter ¹	CAN0ERRH	0x97	CAN0ERRL	0x96	CAN0ERR	0x0000
0x06	Bit Timing Register ²	CAN0BTH	0x9B	CAN0BTL	0x9A	CAN0BT	0x2301
0x08	Interrupt Register ¹	CAN0IIDH	0x9D	CAN0IIDL	0x9C	CAN0IID	0x0000
0x0A	Test Register	—	—	CAN0TST	0x9E	—	0x00 ^{3,4}
0x0C	BRP Extension Register ²	—	—	CAN0BRPE	0xA1	—	0x00
0x10	IF1 Command Request	CAN0IF1CRH	0xBF	CAN0IF1CRL	0xBE	CAN0IF1CR	0x0001
0x12	IF1 Command Mask	CAN0IF1CMH	0xC3	CAN0IF1CML	0xC2	CAN0IF1CM	0x0000
0x14	IF1 Mask 1	CAN0IF1M1H	0xC5	CAN0IF1M1L	0xC4	CAN0IF1M1	0xFFFF
0x16	IF1 Mask 2	CAN0IF1M2H	0xC7	CAN0IF1M2L	0xC6	CAN0IF1M2	0xFFFF
0x18	IF1 Arbitration 1	CAN0IF1A1H	0xCB	CAN0IF1A1L	0xCA	CAN0IF1A1	0x0000
0x1A	IF1 Arbitration 2	CAN0IF1A2H	0xCD	CAN0IF1A2L	0xCC	CAN0IF1A2	0x0000
0x1C	IF1 Message Control	CAN0IF1MCH	0xD3	CAN0IF1MCL	0xD2	CAN0IF1MC	0x0000
0x1E	IF1 Data A 1	CAN0IF1DA1H	0xD5	CAN0IF1DA1L	0xD4	CAN0IF1DA1	0x0000
0x20	IF1 Data A 2	CAN0IF1DA2H	0xD7	CAN0IF1DA2L	0xD6	CAN0IF1DA2	0x0000
0x22	IF1 Data B 1	CAN0IF1DB1H	0xDB	CAN0IF1DB1L	0xDA	CAN0IF1DB1	0x0000
0x24	IF1 Data B 2	CAN0IF1DB2H	0xDD	CAN0IF1DB2L	0xDC	CAN0IF1DB2	0x0000
0x40	IF2 Command Request	CAN0IF2CRH	0xDF	CAN0IF2CRL	0xDE	CAN0IF2CR	0x0001
0x42	IF2 Command Mask	CAN0IF2CMH	0xE3	CAN0IF2CML	0xE2	CAN0IF2CM	0x0000
0x44	IF2 Mask 1	CAN0IF2M1H	0xEB	CAN0IF2M1L	0xEA	CAN0IF2M1	0xFFFF
0x46	IF2 Mask 2	CAN0IF2M2H	0xED	CAN0IF2M2L	0xEC	CAN0IF2M2	0xFFFF
0x48	IF2 Arbitration 1	CAN0IF2A1H	0xEF	CAN0IF2A1L	0xEE	CAN0IF2A1	0x0000
0x4A	IF2 Arbitration 2	CAN0IF2A2H	0xF3	CAN0IF2A2L	0xF2	CAN0IF2A2	0x0000
0x4C	IF2 Message Control	CAN0IF2MCH	0xCF	CAN0IF2MCL	0xCE	CAN0IF2MC	0x0000
0x4E	IF2 Data A 1	CAN0IF2DA1H	0xF7	CAN0IF2DA1L	0xF6	CAN0IF2DA1	0x0000
Notes: <ol style="list-style-type: none"> 1. Read-only register. 2. Write-enabled by CCE. 3. The reset value of CAN0TST could also be r0000000b, where r signifies the value of the CAN RX pin. 4. Write-enabled by Test. 							

C8051F58x/F59x

SFR Definition 26.1. SPI0CFG: SPI0 Configuration

Bit	7	6	5	4	3	2	1	0
Name	SPIBSY	MSTEN	CKPHA	CKPOL	SLVSEL	NSSIN	SRMT	RXBMT
Type	R	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	0	1	1	1

SFR Address = 0xA1; SFR Page = 0x00

Bit	Name	Function
7	SPIBSY	SPI Busy. This bit is set to logic 1 when a SPI transfer is in progress (master or slave mode).
6	MSTEN	Master Mode Enable. 0: Disable master mode. Operate in slave mode. 1: Enable master mode. Operate as a master.
5	CKPHA	SPI0 Clock Phase. 0: Data centered on first edge of SCK period.* 1: Data centered on second edge of SCK period.*
4	CKPOL	SPI0 Clock Polarity. 0: SCK line low in idle state. 1: SCK line high in idle state.
3	SLVSEL	Slave Selected Flag. This bit is set to logic 1 whenever the NSS pin is low indicating SPI0 is the selected slave. It is cleared to logic 0 when NSS is high (slave not selected). This bit does not indicate the instantaneous value at the NSS pin, but rather a de-glitched version of the pin input.
2	NSSIN	NSS Instantaneous Pin Input. This bit mimics the instantaneous value that is present on the NSS port pin at the time that the register is read. This input is not de-glitched.
1	SRMT	Shift Register Empty (valid in slave mode only). This bit will be set to logic 1 when all data has been transferred in/out of the shift register, and there is no new information available to read from the transmit buffer or write to the receive buffer. It returns to logic 0 when a data byte is transferred to the shift register from the transmit buffer or by a transition on SCK. SRMT = 1 when in Master Mode.
0	RXBMT	Receive Buffer Empty (valid in slave mode only). This bit will be set to logic 1 when the receive buffer has been read and contains no new information. If there is new information available in the receive buffer that has not been read, this bit will return to logic 0. RXBMT = 1 when in Master Mode.
Note: In slave mode, data on MOSI is sampled in the center of each data bit. In master mode, data on MISO is sampled one SYSCLK before the end of each data bit, to provide maximum settling time for the slave device. See Table 26.1 for timing parameters.		

29.3.5.2. 9/10/11-bit Pulse Width Modulator Mode

The duty cycle of the PWM output signal in 9/10/11-bit PWM mode should be varied by writing to an “Auto-Reload” Register, which is dual-mapped into the PCA1CPHn and PCA1CPLn register locations. The data written to define the duty cycle should be right-justified in the registers. The auto-reload registers are accessed (read or written) when the bit ARSEL1 in PCA1PWM is set to 1. The capture/compare registers are accessed when ARSEL1 is set to 0.

When the least-significant N bits of the PCA1 counter match the value in the associated module’s capture/compare register (PCA1CPn), the output on CEXn is asserted high. When the counter overflows from the Nth bit, CEXn is asserted low (see Figure 29.9). Upon an overflow from the Nth bit, the COVF1 flag is set, and the value stored in the module’s auto-reload register is loaded into the capture/compare register. The value of N is determined by the CLSEL1 bits in register PCA1PWM.

The 9, 10 or 11-bit PWM mode is selected by setting the ECOM1n and PWM1n bits in the PCA1CPMn register, and setting the CLSEL1 bits in register PCA1PWM to the desired cycle length (other than 8-bits). If the MAT1n bit is set to 1, the CCFn flag for the module will be set each time a comparator match (rising edge) occurs. The COVF1 flag in PCA1PWM can be used to detect the overflow (falling edge), which will occur every 512 (9-bit), 1024 (10-bit) or 2048 (11-bit) PCA1 clock cycles. The duty cycle for 9/10/11-Bit PWM Mode is given in Equation 29.2, where N is the number of bits in the PWM cycle.

Important Note About PCA1CPHn and PCA1CPLn Registers: When writing a 16-bit value to the PCA1CPn registers, the low byte should always be written first. Writing to PCA1CPLn clears the ECOM1n bit to 0; writing to PCA1CPHn sets ECOM1n to 1.

$$\text{Duty Cycle} = \frac{(2^N - \text{PCA1CPn})}{2^N}$$

Equation 29.3. 9, 10, and 11-Bit PWM Duty Cycle

A 0% duty cycle may be generated by clearing the ECOM1n bit to 0.

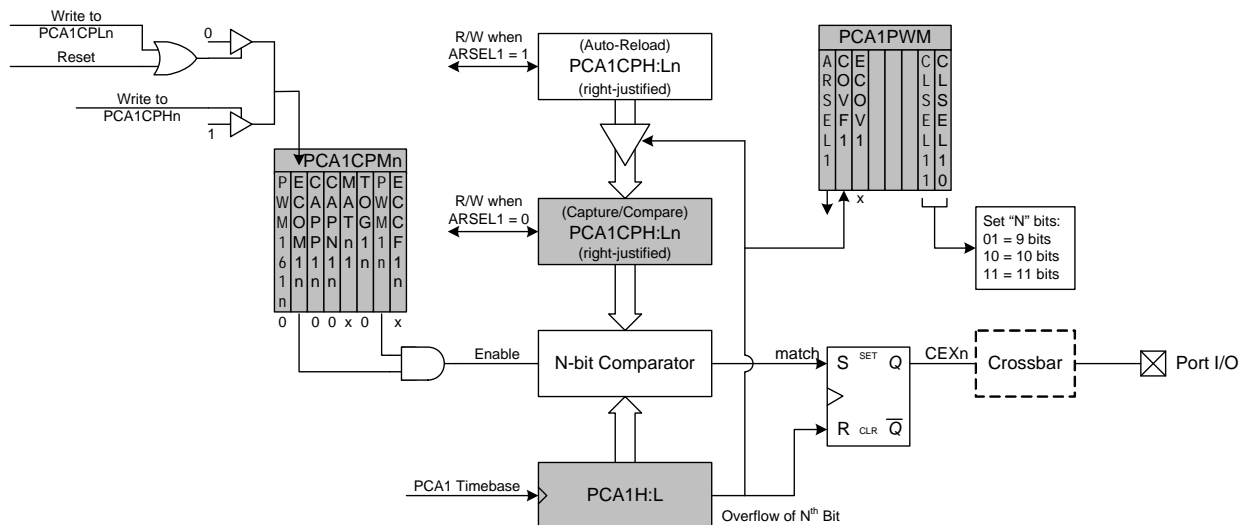


Figure 29.9. PCA1 9, 10 and 11-Bit PWM Mode Diagram

C8051F58x/F59x

C2 Register Definition 30.2. DEVICEID: C2 Device ID

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

C2 Address = 0xFD; SFR Address = 0xFD; SFR Page = 0x0F

Bit	Name	Function
7:0	DEVICEID[7:0]	Device ID. This read-only register returns the 8-bit device ID: 0x20 (C8051F58x/F59x).

C2 Register Definition 30.3. REVID: C2 Revision ID

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

C2 Address = 0xFE; SFR Address = 0xFE; SFR Page = 0x0F

Bit	Name	Function
7:0	REVID[7:0]	Revision ID. This read-only register returns the 8-bit revision ID. For example: 0x00 = Revision A.