

Welcome to [E-XFL.COM](http://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](#)"

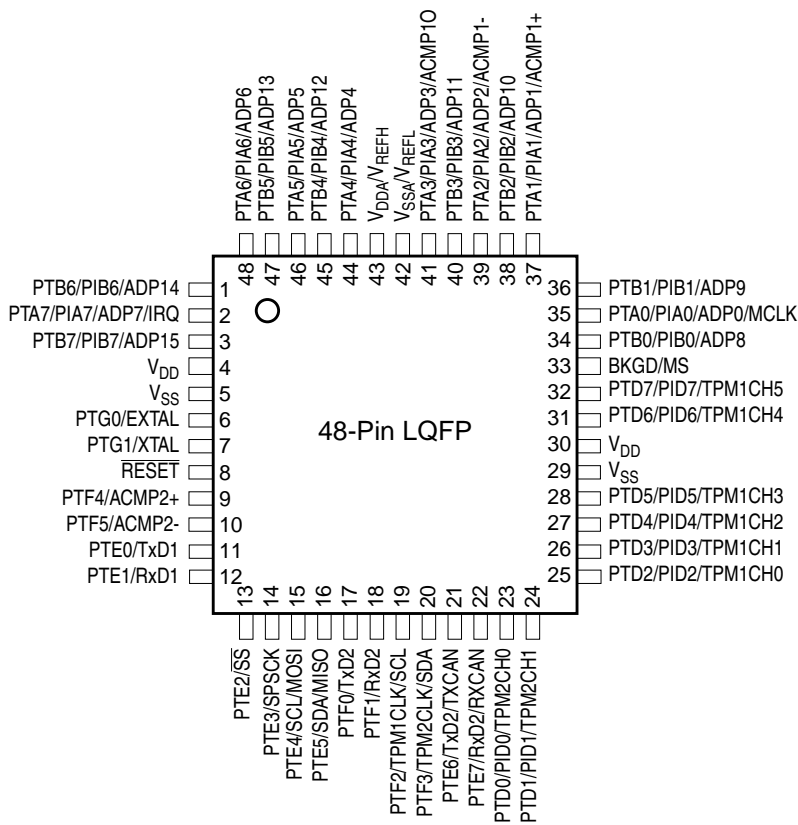
#### Details

Product Status	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	CANbus, I <sup>2</sup> C, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	39
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	<a href="https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s08dv32clf">https://www.e-xfl.com/pro/item?MUrl=&amp;PartUrl=mc9s08dv32clf</a>

Section Number	Title	Page
10.2	External Signal Description .....	176
10.2.1	Analog Power ( $V_{DDAD}$ ) .....	177
10.2.2	Analog Ground ( $V_{SSAD}$ ) .....	177
10.2.3	Voltage Reference High ( $V_{REFH}$ ) .....	177
10.2.4	Voltage Reference Low ( $V_{REFL}$ ) .....	177
10.2.5	Analog Channel Inputs ( $ADx$ ) .....	177
10.3	Register Definition .....	177
10.3.1	Status and Control Register 1 (ADCSC1) .....	177
10.3.2	Status and Control Register 2 (ADCSC2) .....	179
10.3.3	Data Result High Register (ADCRH) .....	179
10.3.4	Data Result Low Register (ADCRL) .....	180
10.3.5	Compare Value High Register (ADCCVH) .....	180
10.3.6	Compare Value Low Register (ADCCVL) .....	181
10.3.7	Configuration Register (ADCCFG) .....	181
10.3.8	Pin Control 1 Register (APCTL1) .....	182
10.3.9	Pin Control 2 Register (APCTL2) .....	183
10.3.10	Pin Control 3 Register (APCTL3) .....	184
10.4	Functional Description .....	185
10.4.1	Clock Select and Divide Control .....	186
10.4.2	Input Select and Pin Control .....	186
10.4.3	Hardware Trigger .....	186
10.4.4	Conversion Control .....	186
10.4.5	Automatic Compare Function .....	189
10.4.6	MCU Wait Mode Operation .....	189
10.4.7	MCU Stop3 Mode Operation .....	190
10.4.8	MCU Stop2 Mode Operation .....	190
10.5	Initialization Information .....	191
10.5.1	ADC Module Initialization Example .....	191
10.6	Application Information .....	193
10.6.1	External Pins and Routing .....	193
10.6.2	Sources of Error .....	194

## Chapter 11 Inter-Integrated Circuit (S08IICV2)

11.1	Introduction .....	197
11.1.1	Features .....	199
11.1.2	Modes of Operation .....	199
11.1.3	Block Diagram .....	200
11.2	External Signal Description .....	200
11.2.1	SCL — Serial Clock Line .....	200
11.2.2	SDA — Serial Data Line .....	200
11.3	Register Definition .....	200



V<sub>REFH</sub> and V<sub>REFL</sub> are internally connected to V<sub>DDA</sub> and V<sub>SSA</sub>, respectively.

**Figure 2-2. 48-Pin LQFP**

- <sup>5</sup> ERCLKEN and EREFSTEN set in MCGC2 for, else in standby. For high frequency range (RANGE in MCGC2 set) requires the LVD to also be enabled in stop3.
- <sup>6</sup> If ENBDM is set when entering stop2, the MCU will actually enter stop3.
- <sup>7</sup> If LVDSE is set when entering stop2, the MCU will actually enter stop3.


**Table 5-3. SRS Register Field Descriptions**

Field	Description
2 LOC	<b>Loss of Clock</b> — Reset was caused by a loss of external clock. 0 Reset not caused by loss of external clock 1 Reset caused by loss of external clock
1 LVD	<b>Low-Voltage Detect</b> — If the LVDRE bit is set and the supply drops below the LVD trip voltage, an LVD reset will occur. This bit is also set by POR. 0 Reset not caused by LVD trip or POR. 1 Reset caused by LVD trip or POR.

### 5.8.3 System Background Debug Force Reset Register (SBDFR)

This high page register contains a single write-only control bit. A serial background command such as WRITE\_BYTE must be used to write to SBDFR. Attempts to write this register from a user program are ignored. Reads always return 0x00.

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								BDFR <sup>1</sup>
Reset:	0	0	0	0	0	0	0	0

 = Unimplemented or Reserved

<sup>1</sup> BDFR is writable only through serial background debug commands, not from user programs.

**Figure 5-4. System Background Debug Force Reset Register (SBDFR)**
**Table 5-4. SBDFR Register Field Descriptions**

Field	Description
0 BDFR	<b>Background Debug Force Reset</b> — A serial background command such as WRITE_BYTE can be used to allow an external debug host to force a target system reset. Writing 1 to this bit forces an MCU reset. This bit cannot be written from a user program.

## 6.5.1 Port A Registers

Port A is controlled by the registers listed below.

### 6.5.1.1 Port A Data Register (PTAD)

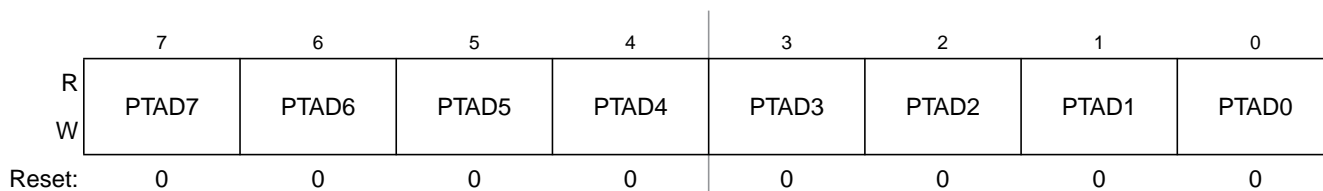


Figure 6-3. Port A Data Register (PTAD)

Table 6-1. PTAD Register Field Descriptions

Field	Description
7:0 PTAD[7:0]	<b>Port A Data Register Bits</b> — For port A pins that are inputs, reads return the logic level on the pin. For port A pins that are configured as outputs, reads return the last value written to this register. Writes are latched into all bits of this register. For port A pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTAD to all 0s, but these 0s are not driven out the corresponding pins because reset also configures all port pins as high-impedance inputs with pull-ups/pull-downs disabled.

### 6.5.1.2 Port A Data Direction Register (PTADD)

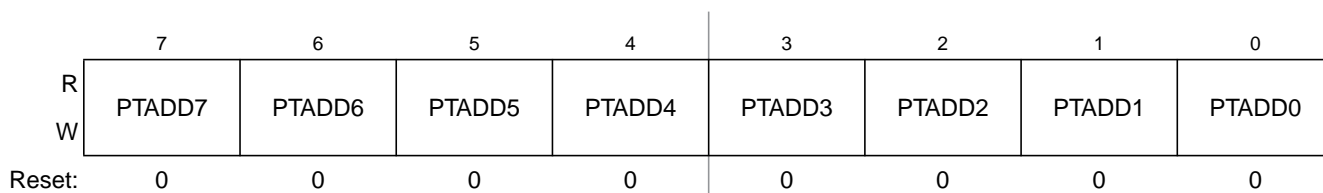


Figure 6-4. Port A Data Direction Register (PTADD)

Table 6-2. PTADD Register Field Descriptions

Field	Description
7:0 PTADD[7:0]	<b>Data Direction for Port A Bits</b> — These read/write bits control the direction of port A pins and what is read for PTAD reads. 0 Input (output driver disabled) and reads return the pin value. 1 Output driver enabled for port A bit n and PTAD reads return the contents of PTADn.

### 6.5.5.3 Port E Pull Enable Register (PTEPE)

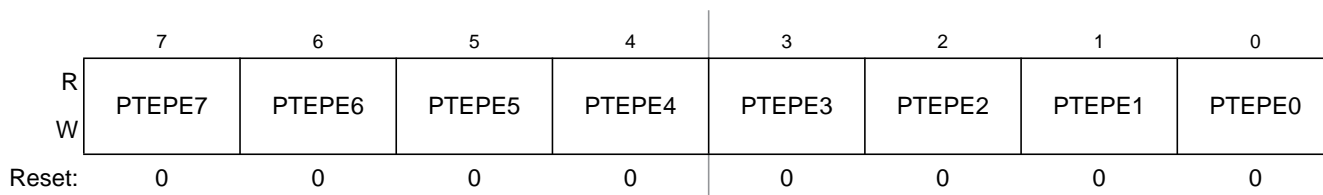


Figure 6-34. Internal Pull Enable for Port E Register (PTEPE)

Table 6-32. PTEPE Register Field Descriptions

Field	Description
7:0 PTEPE[7:0]	<p><b>Internal Pull Enable for Port E Bits</b> — Each of these control bits determines if the internal pull-up device is enabled for the associated PTE pin. For port E pins that are configured as outputs, these bits have no effect and the internal pull devices are disabled.</p> <p>0 Internal pull-up device disabled for port E bit n. 1 Internal pull-up device enabled for port E bit n.</p>

**NOTE**

Pull-down devices only apply when using pin interrupt functions, when corresponding edge select and pin select functions are configured.

### 6.5.5.4 Port E Slew Rate Enable Register (PTESE)

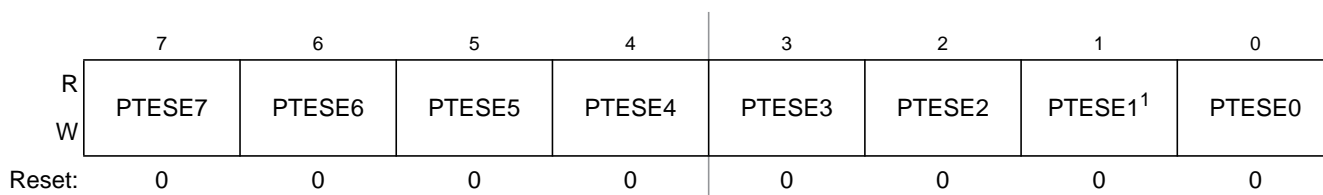


Figure 6-35. Slew Rate Enable for Port E Register (PTESE)

<sup>1</sup> PTESE1 has no effect on the input-only PTE1 pin.

Table 6-33. PTESE Register Field Descriptions

Field	Description
7:0 PTESE[7:0]	<p><b>Output Slew Rate Enable for Port E Bits</b> — Each of these control bits determines if the output slew rate control is enabled for the associated PTE pin. For port E pins that are configured as inputs, these bits have no effect.</p> <p>0 Output slew rate control disabled for port E bit n. 1 Output slew rate control enabled for port E bit n.</p>

**Note:** Slew rate reset default values may differ between engineering samples and final production parts. Always initialize slew rate control to the desired value to ensure correct operation.



Table 7-3. Opcode Map (Sheet 2 of 2)

Bit-Manipulation	Branch	Read-Modify-Write				Control				Register/Memory												
				9E60	6 NEG 3 SP1							9ED0	5 SUB 4 SP2	9EE0	4 SUB 3 SP1							
				9E61	6 CBEQ 4 SP1							9ED1	5 CMP 4 SP2	9EE1	4 CMP 3 SP1							
												9ED2	5 SBC 4 SP2	9EE2	4 SBC 3 SP1							
				9E63	6 COM 3 SP1							9ED3	5 CPX 4 SP2	9EE3	4 CPX 3 SP1	9EF3	6 CPHX 3 SP1					
				9E64	6 LSR 3 SP1							9ED4	5 AND 4 SP2	9EE4	4 AND 3 SP1							
												9ED5	5 BIT 4 SP2	9EE5	4 BIT 3 SP1							
				9E66	6 ROR 3 SP1							9ED6	5 LDA 4 SP2	9EE6	4 LDA 3 SP1							
				9E67	6 ASR 3 SP1							9ED7	5 STA 4 SP2	9EE7	4 STA 3 SP1							
				9E68	6 LSL 3 SP1							9ED8	5 EOR 4 SP2	9EE8	4 EOR 3 SP1							
				9E69	6 ROL 3 SP1							9ED9	5 ADC 4 SP2	9EE9	4 ADC 3 SP1							
				9E6A	6 DEC 3 SP1							9EDA	5 ORA 4 SP2	9EEA	4 ORA 3 SP1							
				9E6B	8 DBNZ 4 SP1							9EDB	5 ADD 4 SP2	9EEB	4 ADD 3 SP1							
				9E6C	6 INC 3 SP1																	
				9E6D	5 TST 3 SP1																	
											9EAE	5 LDHX 2 IX	9EBE	6 LDHX 4 IX2	9ECE	5 LDHX 3 IX1	9EDE	5 LDX 4 SP2	9EEE	4 LDX 3 SP1	9EFE	5 LDHX 3 SP1
				9E6F	6 CLR 3 SP1								9EDF	5 STX 4 SP2	9EEF	4 STX 3 SP1	9EFF	5 STHX 3 SP1				

INH Inherent      REL Relative      SP1 Stack Pointer, 8-Bit Offset  
 IMM Immediate    IX Indexed, No Offset    SP2 Stack Pointer, 16-Bit Offset  
 DIR Direct        IX1 Indexed, 8-Bit Offset    IX+ Indexed, No Offset with Post Increment  
 EXT Extended     IX2 Indexed, 16-Bit Offset    IX1+ Indexed, 1-Byte Offset with Post Increment  
 DD DIR to DIR    IMD IMM to DIR  
 IX+D IX+ to DIR    DIX+ DIR to IX+

Note: All Sheet 2 Opcodes are Preceded by the Page 2 Prebyte (9E)

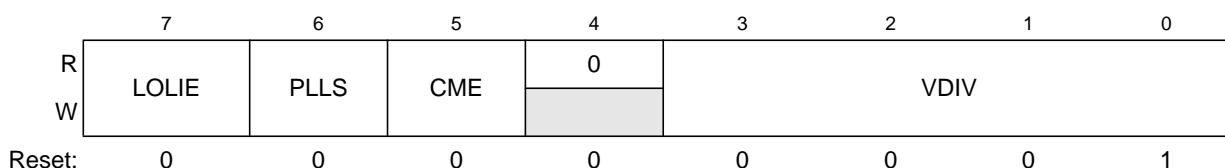
Prebyte (9E) and Opcode in Hexadecimal	9E60	6	NEG	3	SP1	HCS08 Cycles
Number of Bytes						Instruction Mnemonic
						Addressing Mode



**Table 8-4. MCG Status and Control Register Field Descriptions (continued)**

Field	Description
1 OSCINIT	<b>OSC Initialization</b> — If the external reference clock is selected by ERCLKEN or by the MCG being in FEE, FBE, PEE, PBE, or BLPE mode, and if EREFS is set, then this bit is set after the initialization cycles of the external oscillator clock have completed. This bit is only cleared when either EREFS is cleared or when the MCG is in either FEI, FBI, or BLPI mode and ERCLKEN is cleared.
0 FTRIM	<b>MCG Fine Trim</b> — Controls the smallest adjustment of the internal reference clock frequency. Setting FTRIM will increase the period and clearing FTRIM will decrease the period by the smallest amount possible.  If an FTRIM value stored in nonvolatile memory is to be used, it's the user's responsibility to copy that value from the nonvolatile memory location to this register's FTRIM bit.

### 8.3.5 MCG Control Register 3 (MCGC3)



**Figure 8-7. MCG PLL Register (MCGPLL)**

**Table 8-5. MCG PLL Register Field Descriptions**

Field	Description
7 LOLIE	<b>Loss of Lock Interrupt Enable</b> — Determines if an interrupt request is made following a loss of lock indication. The LOLIE bit only has an effect when LOLS is set. 0 No request on loss of lock. 1 Generate an interrupt request on loss of lock.
6 PLLS	<b>PLL Select</b> — Controls whether the PLL or FLL is selected. If the PLLS bit is clear, the PLL is disabled in all modes. If the PLLS is set, the FLL is disabled in all modes. 1 PLL is selected 0 FLL is selected

# 8.4 Functional Description

## 8.4.1 Operational Modes

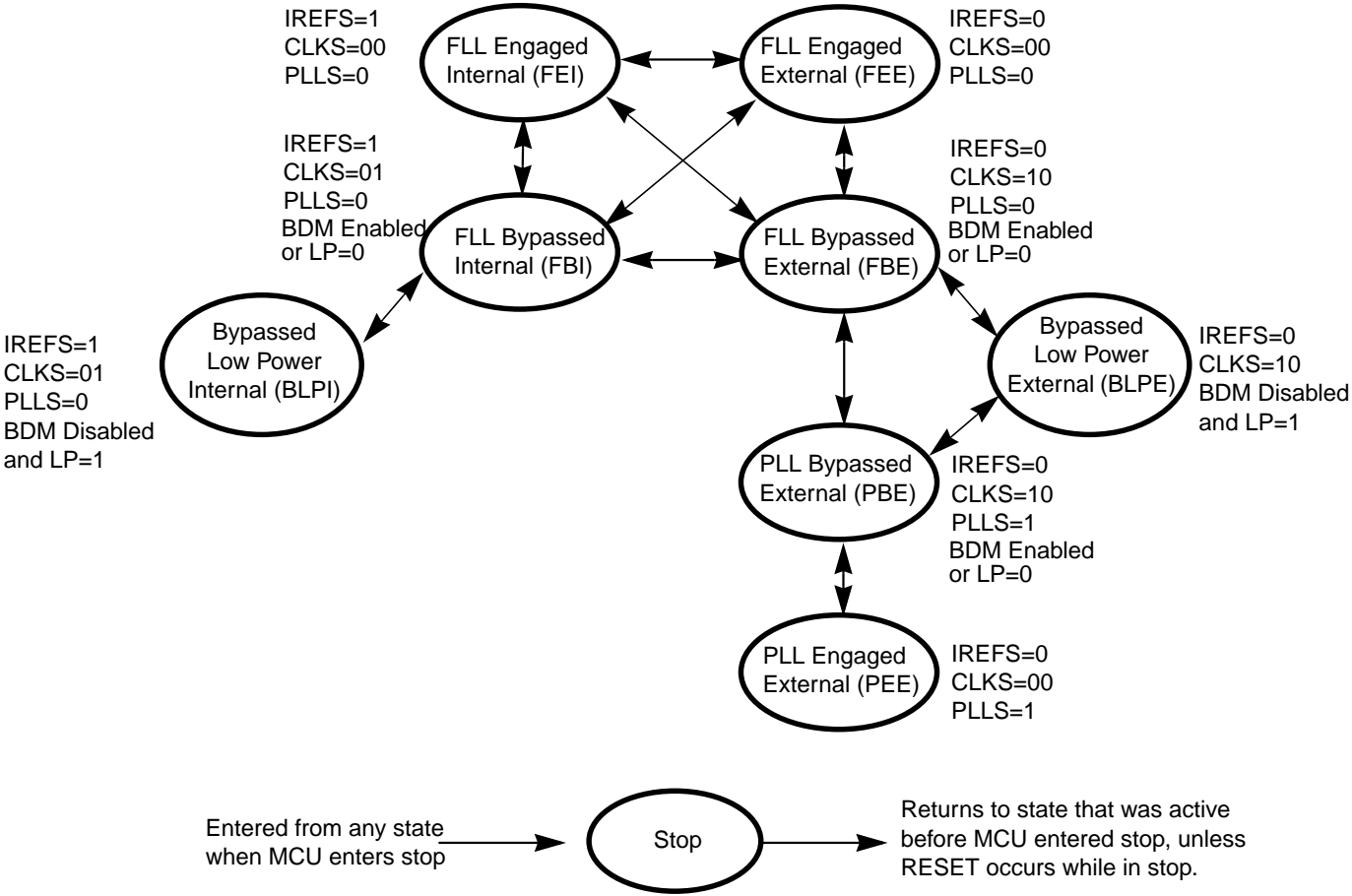


Figure 8-8. Clock Switching Modes

multiplication factor, as selected by the VDIV bits, times the reference frequency, as selected by the RDIV bits. If BDM is enabled then the MCGLCLK is derived from the DCO (open-loop mode) divided by two. If BDM is not enabled then the FLL is disabled in a low power state.

#### 8.4.1.6 PLL Bypassed External (PBE)

In PLL bypassed external (PBE) mode, the MCGOUT clock is derived from the external reference clock and the PLL is operational but its output clock is not used. This mode is useful to allow the PLL to acquire its target frequency while the MCGOUT clock is driven from the external reference clock.

The PLL bypassed external mode is entered when all the following conditions occur:

- CLKS bits are written to 10
- IREFS bit is written to 0
- PLLS bit is written to 1
- RDIV bits are written to divide reference clock to be within the range of 1 MHz to 2 MHz
- LP bit is written to 0

In PLL bypassed external mode, the MCGOUT clock is derived from the external reference clock. The external reference clock which is enabled can be an external crystal/resonator or it can be another external clock source. The PLL clock frequency locks to a multiplication factor, as selected by the VDIV bits, times the reference frequency, as selected by the RDIV bits. If BDM is enabled then the MCGLCLK is derived from the DCO (open-loop mode) divided by two. If BDM is not enabled then the FLL is disabled in a low power state.

#### 8.4.1.7 Bypassed Low Power Internal (BLPI)

The bypassed low power internal (BLPI) mode is entered when all the following conditions occur:

- CLKS bits are written to 01
- IREFS bit is written to 1
- PLLS bit is written to 0
- LP bit is written to 1
- BDM mode is not active

In bypassed low power internal mode, the MCGOUT clock is derived from the internal reference clock.

The PLL and the FLL are disabled at all times in BLPI mode and the MCGLCLK will not be available for BDC communications. If the BDM becomes active the mode will switch to FLL bypassed internal (FBI) mode.

#### 8.4.1.8 Bypassed Low Power External (BLPE)

The bypassed low power external (BLPE) mode is entered when all the following conditions occur:

- CLKS bits are written to 10
- IREFS bit is written to 0
- PLLS bit is written to 0 or 1

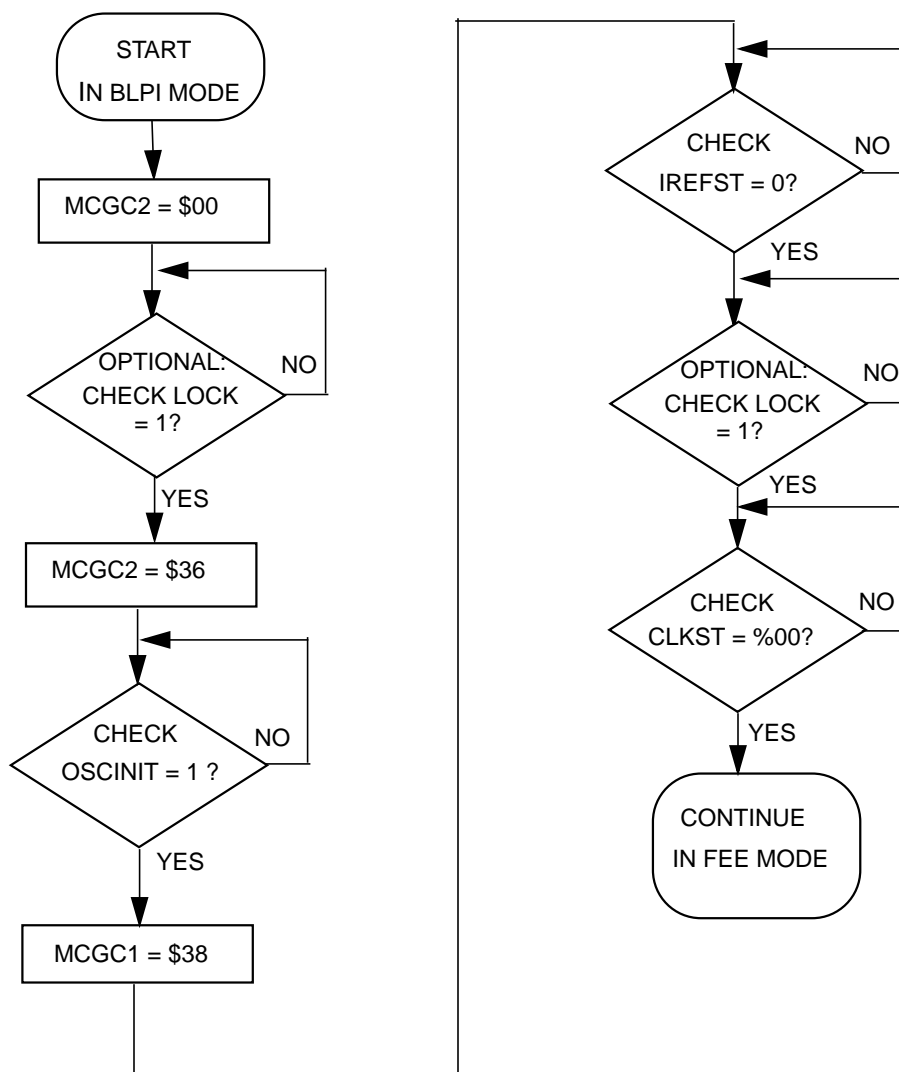


Figure 8-11. Flowchart of BLPI to FEE Mode Transition using a 4 MHz crystal

### 8.5.2.4 Example # 4: Moving from FEI to PEE Mode: External Crystal = 8 MHz, Bus Frequency = 8 MHz

In this example, the MCG will move through the proper operational modes from FEI to PEE mode until the 8 MHz crystal reference frequency is set to achieve a bus frequency of 8 MHz.

This example is similar to example number one except that in this case the frequency of the external crystal is 8 MHz instead of 4 MHz. Special consideration must be taken with this case since there is a period of time along the way from FEI mode to PEE mode where the FLL operates based on a reference clock with a frequency that is greater than the maximum allowed for the FLL. This occurs because with an 8 MHz

ALTCLK for this MCU. Consult the module introduction for information on ALTCLK specific to this MCU.

A conversion complete event sets the COCO and generates an ADC interrupt to wake the MCU from wait mode if the ADC interrupt is enabled (AIEN = 1).

## 10.4.7 MCU Stop3 Mode Operation

Stop mode is a low power-consumption standby mode during which most or all clock sources on the MCU are disabled.

### 10.4.7.1 Stop3 Mode With ADACK Disabled

If the asynchronous clock, ADACK, is not selected as the conversion clock, executing a stop instruction aborts the current conversion and places the ADC in its idle state. The contents of ADCRH and ADCRL are unaffected by stop3 mode. After exiting from stop3 mode, a software or hardware trigger is required to resume conversions.

### 10.4.7.2 Stop3 Mode With ADACK Enabled

If ADACK is selected as the conversion clock, the ADC continues operation during stop3 mode. For guaranteed ADC operation, the MCU's voltage regulator must remain active during stop3 mode. Consult the module introduction for configuration information for this MCU.

If a conversion is in progress when the MCU enters stop3 mode, it continues until completion. Conversions can be initiated while the MCU is in stop3 mode by means of the hardware trigger or if continuous conversions are enabled.

A conversion complete event sets the COCO and generates an ADC interrupt to wake the MCU from stop3 mode if the ADC interrupt is enabled (AIEN = 1).

#### NOTE

The ADC module can wake the system from low-power stop and cause the MCU to begin consuming run-level currents without generating a system level interrupt. To prevent this scenario, software should ensure the data transfer blocking mechanism (discussed in [Section 10.4.4.2, “Completing Conversions”](#)) is cleared when entering stop3 and continuing ADC conversions.

## 10.4.8 MCU Stop2 Mode Operation

The ADC module is automatically disabled when the MCU enters stop2 mode. All module registers contain their reset values following exit from stop2. Therefore, the module must be re-enabled and re-configured following exit from stop2.

**Table 11-2. IICF Field Descriptions**

Field	Description
7–6 MULT	<p><b>IIC Multiplier Factor.</b> The MULT bits define the multiplier factor, mul. This factor, along with the SCL divider, generates the IIC baud rate. The multiplier factor mul as defined by the MULT bits is provided below.</p> <p>00 mul = 01 01 mul = 02 10 mul = 04 11 Reserved</p>
5–0 ICR	<p><b>IIC Clock Rate.</b> The ICR bits are used to prescale the bus clock for bit rate selection. These bits and the MULT bits determine the IIC baud rate, the SDA hold time, the SCL Start hold time, and the SCL Stop hold time. <a href="#">Table 11-4</a> provides the SCL divider and hold values for corresponding values of the ICR.</p> <p>The SCL divider multiplied by multiplier factor mul generates IIC baud rate.</p> $\text{IIC baud rate} = \frac{\text{bus speed (Hz)}}{\text{mul} \times \text{SCLdivider}} \quad \text{Eqn. 11-1}$ <p>SDA hold time is the delay from the falling edge of SCL (IIC clock) to the changing of SDA (IIC data).</p> $\text{SDA hold time} = \text{bus period (s)} \times \text{mul} \times \text{SDA hold value} \quad \text{Eqn. 11-2}$ <p>SCL start hold time is the delay from the falling edge of SDA (IIC data) while SCL is high (Start condition) to the falling edge of SCL (IIC clock).</p> $\text{SCL Start hold time} = \text{bus period (s)} \times \text{mul} \times \text{SCL Start hold value} \quad \text{Eqn. 11-3}$ <p>SCL stop hold time is the delay from the rising edge of SCL (IIC clock) to the rising edge of SDA (IIC data) while SCL is high (Stop condition).</p> $\text{SCL Stop hold time} = \text{bus period (s)} \times \text{mul} \times \text{SCL Stop hold value} \quad \text{Eqn. 11-4}$

For example, if the bus speed is 8 MHz, the table below shows the possible hold time values with different ICR and MULT selections to achieve an IIC baud rate of 100kbps.

**Table 11-3. Hold Time Values for 8 MHz Bus Speed**

MULT	ICR	Hold Times (μs)		
		SDA	SCL Start	SCL Stop
0x2	0x00	3.500	3.000	5.500
0x1	0x07	2.500	4.000	5.250
0x1	0x0B	2.250	4.000	5.250
0x0	0x14	2.125	4.250	5.125
0x0	0x18	1.125	4.750	5.125

**Table 12-33. Data Length Codes**

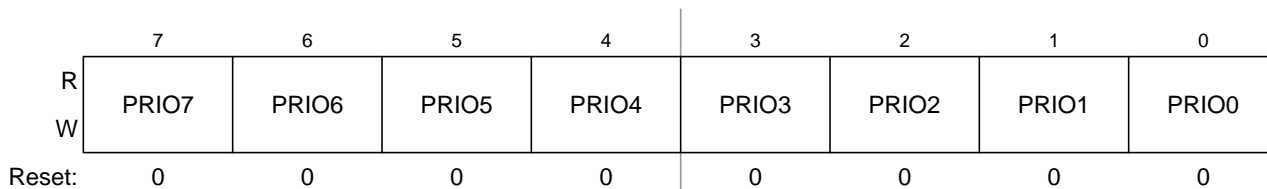
Data Length Code				Data Byte Count
DLC3	DLC2	DLC1	DLC0	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8

### 12.4.5 Transmit Buffer Priority Register (TBPR)

This register defines the local priority of the associated message transmit buffer. The local priority is used for the internal prioritization process of the MSCAN and is defined to be highest for the smallest binary number. The MSCAN implements the following internal prioritization mechanisms:

- All transmission buffers with a cleared TXEx flag participate in the prioritization immediately before the SOF (start of frame) is sent.
- The transmission buffer with the lowest local priority field wins the prioritization.

In cases of more than one buffer having the same lowest priority, the message buffer with the lower index number wins.


**Figure 12-35. Transmit Buffer Priority Register (TBPR)**

**Read:** Anytime when TXEx flag is set (see Section 12.3.6, “MSCAN Transmitter Flag Register (CANTFLG)”) and the corresponding transmit buffer is selected in CANTBSEL (see Section 12.3.10, “MSCAN Transmit Buffer Selection Register (CANTBSEL)”).

**Write:** Anytime when TXEx flag is set (see Section 12.3.6, “MSCAN Transmitter Flag Register (CANTFLG)”) and the corresponding transmit buffer is selected in CANTBSEL (see Section 12.3.10, “MSCAN Transmit Buffer Selection Register (CANTBSEL)”).

### 12.4.6 Time Stamp Register (TSRH–TSRL)

If the TIME bit is enabled, the MSCAN will write a time stamp to the respective registers in the active transmit or receive buffer as soon as a message has been acknowledged on the CAN bus (see

### 12.5.4.3 Emulation Modes

In all emulation modes, the MSCAN module behaves just like normal system operation modes as described within this specification.

### 12.5.4.4 Listen-Only Mode

In an optional CAN bus monitoring mode (listen-only), the CAN node is able to receive valid data frames and valid remote frames, but it sends only “recessive” bits on the CAN bus. In addition, it cannot start a transmission. If the MAC sub-layer is required to send a “dominant” bit (ACK bit, overload flag, or active error flag), the bit is rerouted internally so that the MAC sub-layer monitors this “dominant” bit, although the CAN bus may remain in recessive state externally.

### 12.5.4.5 Security Modes

The MSCAN module has no security features.

### 12.5.4.6 Loopback Self Test Mode

Loopback self test mode is sometimes used to check software, independent of connections in the external system, to help isolate system problems. In this mode, the transmitter output is internally connected to the receiver input. The RXCAN input pin is ignored and the TXCAN output goes to the recessive state (logic 1). The MSCAN behaves as it does normally when transmitting and treats its own transmitted message as a message received from a remote node. In this state, the MSCAN ignores the bit sent during the ACK slot in the CAN frame acknowledge field to ensure proper reception of its own message. Both transmit and receive interrupts are generated.

## 12.5.5 Low-Power Options

If the MSCAN is disabled ( $CANE = 0$ ), the MSCAN clocks are stopped for power saving.

If the MSCAN is enabled ( $CANE = 1$ ), the MSCAN has two additional modes with reduced power consumption, compared to normal mode: sleep and power down mode. In sleep mode, power consumption is reduced by stopping all clocks except those to access the registers from the CPU side. In power down mode, all clocks are stopped and no power is consumed.

[Table 12-36](#) summarizes the combinations of MSCAN and CPU modes. A particular combination of modes is entered by the given settings on the CSWAI and SLPRQ/SLPAK bits.

For all modes, an MSCAN wake-up interrupt can occur only if the MSCAN is in sleep mode ( $SLPRQ = 1$  and  $SLPAK = 1$ ), wake-up functionality is enabled ( $WUPE = 1$ ), and the wake-up interrupt is enabled ( $WUPIE = 1$ ).



## 13.5 Functional Description

An SPI transfer is initiated by checking for the SPI transmit buffer empty flag (SPTEF = 1) and then writing a byte of data to the SPI data register (SPID) in the master SPI device. When the SPI shift register is available, this byte of data is moved from the transmit data buffer to the shifter, SPTEF is set to indicate there is room in the buffer to queue another transmit character if desired, and the SPI serial transfer starts.

During the SPI transfer, data is sampled (read) on the MISO pin at one SPSCCK edge and shifted, changing the bit value on the MOSI pin, one-half SPSCCK cycle later. After eight SPSCCK cycles, the data that was in the shift register of the master has been shifted out the MOSI pin to the slave while eight bits of data were shifted in the MISO pin into the master's shift register. At the end of this transfer, the received data byte is moved from the shifter into the receive data buffer and SPRF is set to indicate the data can be read by reading SPID. If another byte of data is waiting in the transmit buffer at the end of a transfer, it is moved into the shifter, SPTEF is set, and a new transfer is started.

Normally, SPI data is transferred most significant bit (MSB) first. If the least significant bit first enable (LSBFE) bit is set, SPI data is shifted LSB first.

When the SPI is configured as a slave, its  $\overline{SS}$  pin must be driven low before a transfer starts and  $\overline{SS}$  must stay low throughout the transfer. If a clock format where CPHA = 0 is selected,  $\overline{SS}$  must be driven to a logic 1 between successive transfers. If CPHA = 1,  $\overline{SS}$  may remain low between successive transfers. See [Section 13.5.1, "SPI Clock Formats"](#) for more details.

Because the transmitter and receiver are double buffered, a second byte, in addition to the byte currently being shifted out, can be queued into the transmit data buffer, and a previously received character can be in the receive data buffer while a new character is being shifted in. The SPTEF flag indicates when the transmit buffer has room for a new character. The SPRF flag indicates when a received character is available in the receive data buffer. The received character must be read out of the receive buffer (read SPID) before the next transfer is finished or a receive overrun error results.

In the case of a receive overrun, the new data is lost because the receive buffer still held the previous character and was not ready to accept the new data. There is no indication for such an overrun condition so the application system designer must ensure that previous data has been read from the receive buffer before a new transfer is initiated.

### 13.5.1 SPI Clock Formats

To accommodate a wide variety of synchronous serial peripherals from different manufacturers, the SPI system has a clock polarity (CPOL) bit and a clock phase (CPHA) control bit to select one of four clock formats for data transfers. CPOL selectively inserts an inverter in series with the clock. CPHA chooses between two different clock phase relationships between the clock and data.

[Figure 13-10](#) shows the clock formats when CPHA = 1. At the top of the figure, the eight bit times are shown for reference with bit 1 starting at the first SPSCCK edge and bit 8 ending one-half SPSCCK cycle after the sixteenth SPSCCK edge. The MSB first and LSB first lines show the order of SPI data bits depending on the setting in LSBFE. Both variations of SPSCCK polarity are shown, but only one of these waveforms applies for a specific transfer, depending on the value in CPOL. The SAMPLE IN waveform applies to the MOSI input of a slave or the MISO input of a master. The MOSI waveform applies to the MOSI output

in LSBFE. Both variations of SPSCCK polarity are shown, but only one of these waveforms applies for a specific transfer, depending on the value in CPOL. The SAMPLE IN waveform applies to the MOSI input of a slave or the MISO input of a master. The MOSI waveform applies to the MOSI output pin from a master and the MISO waveform applies to the MISO output from a slave. The  $\overline{SS}$  OUT waveform applies to the slave select output from a master (provided MODFEN and SSOE = 1). The master  $\overline{SS}$  output goes to active low at the start of the first bit time of the transfer and goes back high one-half SPSCCK cycle after the end of the eighth bit time of the transfer. The  $\overline{SS}$  IN waveform applies to the slave select input of a slave.

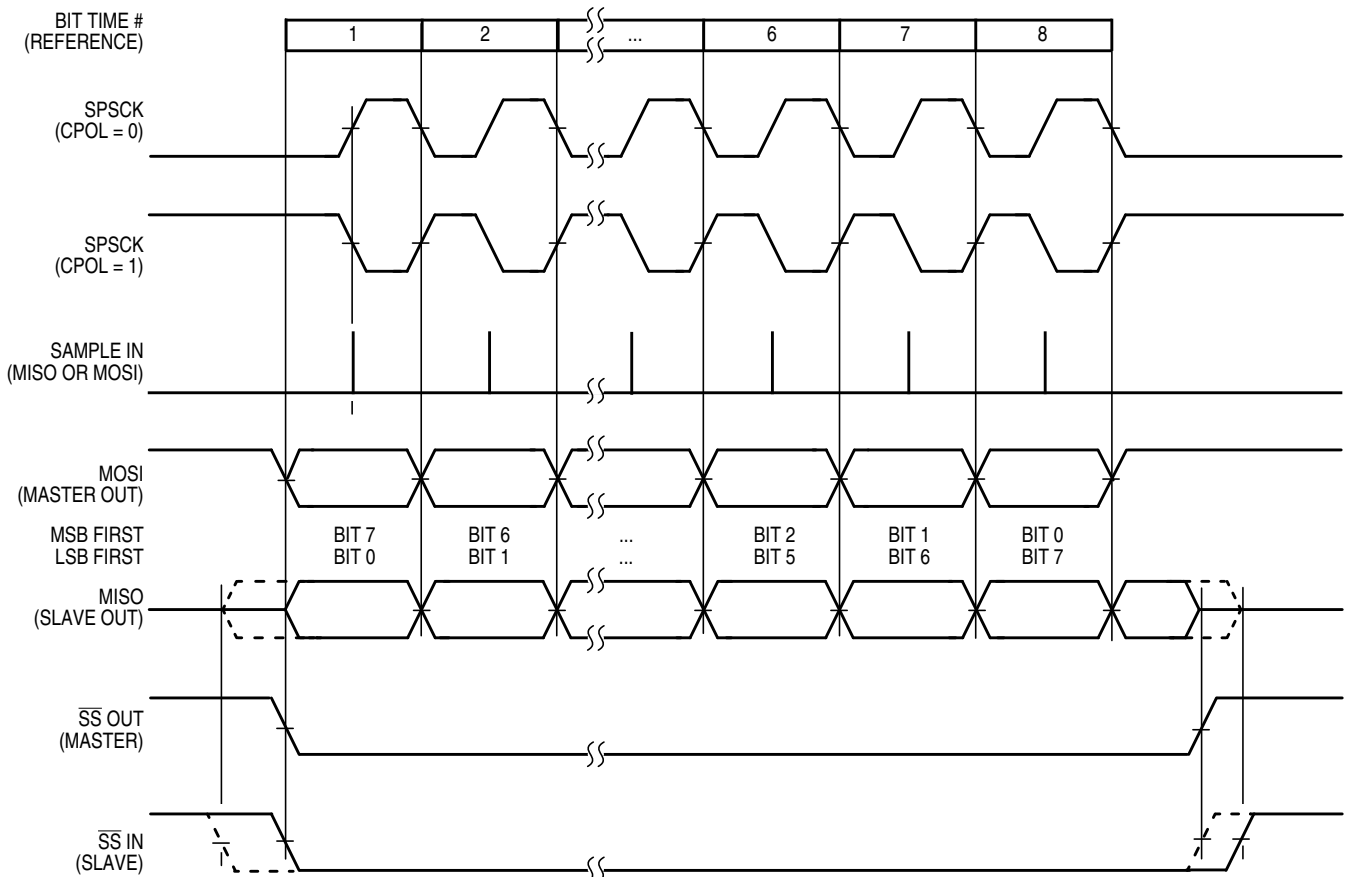


Figure 13-11. SPI Clock Formats (CPHA = 0)

When CPHA = 0, the slave begins to drive its MISO output with the first data bit value (MSB or LSB depending on LSBFE) when  $\overline{SS}$  goes to active low. The first SPSCCK edge causes both the master and the slave to sample the data bit values on their MISO and MOSI inputs, respectively. At the second SPSCCK edge, the SPI shifter shifts one bit position which shifts in the bit value that was just sampled and shifts the second data bit value out the other end of the shifter to the MOSI and MISO outputs of the master and slave, respectively. When CPHA = 0, the slave's  $\overline{SS}$  input must go to its inactive high level between transfers.



When the TPM is configured for center-aligned PWM (and ELSnB:ELSnA not = 0:0), the data direction for all channels in this TPM are overridden, the TPMxCHn pins are forced to be outputs controlled by the TPM, and the ELSnA bits control the polarity of each TPMxCHn output. If ELSnB:ELSnA=1:0, the corresponding TPMxCHn pin is cleared when the timer counter is counting up, and the channel value register matches the timer counter; the TPMxCHn pin is set when the timer counter is counting down, and the channel value register matches the timer counter. If ELSnA=1, the corresponding TPMxCHn pin is set when the timer counter is counting up and the channel value register matches the timer counter; the TPMxCHn pin is cleared when the timer counter is counting down and the channel value register matches the timer counter.

TPMxMODH:TPMxMODL = 0x0008  
 TPMxCnVH:TPMxCnVL = 0x0005

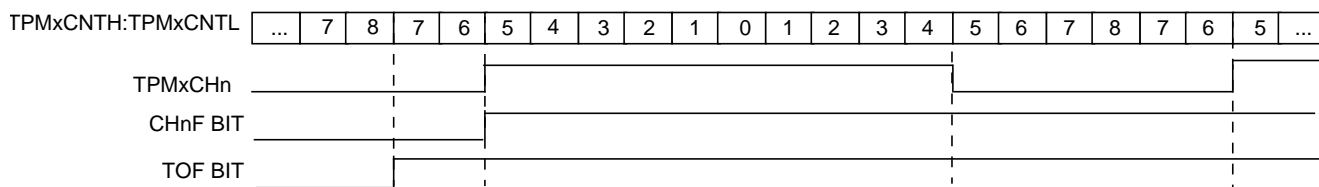


Figure 16-5. High-True Pulse of a Center-Aligned PWM

TPMxMODH:TPMxMODL = 0x0008  
 TPMxCnVH:TPMxCnVL = 0x0005

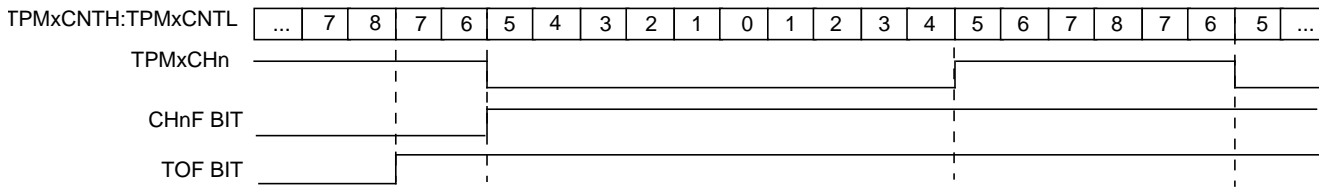


Figure 16-6. Low-True Pulse of a Center-Aligned PWM

## B.2.4 Timer Channel n Status and Control Register (TPMxCnSC)

TPMxCnSC contains the channel interrupt status flag and control bits that are used to configure the interrupt enable, channel configuration, and pin function.

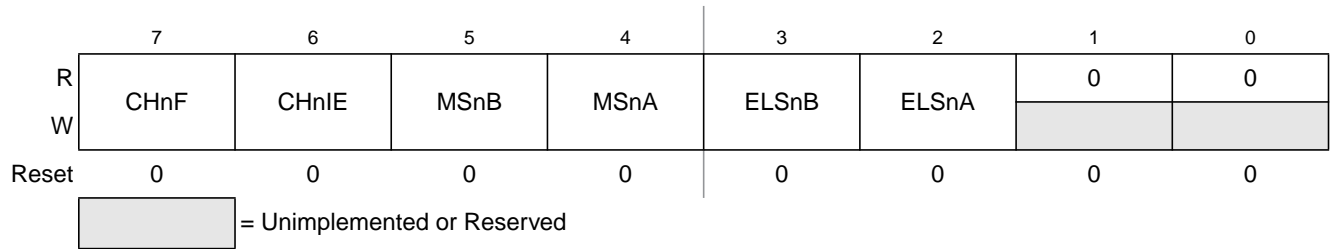


Figure B-7. Timer Channel n Status and Control Register (TPMxCnSC)

Table B-4. TPMxCnSC Register Field Descriptions

Field	Description
7 CHnF	<p><b>Channel n Flag</b> — When channel n is configured for input capture, this flag bit is set when an active edge occurs on the channel n pin. When channel n is an output compare or edge-aligned PWM channel, CHnF is set when the value in the TPM counter registers matches the value in the TPM channel n value registers. This flag is seldom used with center-aligned PWMs because it is set every time the counter matches the channel value register, which correspond to both edges of the active duty cycle period.</p> <p>A corresponding interrupt is requested when CHnF is set and interrupts are enabled (CHnIE = 1). Clear CHnF by reading TPMxCnSC while CHnF is set and then writing a 0 to CHnF. If another interrupt request occurs before the clearing sequence is complete, the sequence is reset so CHnF would remain set after the clear sequence was completed for the earlier CHnF. This is done so a CHnF interrupt request cannot be lost by clearing a previous CHnF. Reset clears CHnF. Writing a 1 to CHnF has no effect.</p> <p>0 No input capture or output compare event occurred on channel n 1 Input capture or output compare event occurred on channel n</p>
6 CHnIE	<p><b>Channel n Interrupt Enable</b> — This read/write bit enables interrupts from channel n. Reset clears CHnIE.</p> <p>0 Channel n interrupt requests disabled (use software polling) 1 Channel n interrupt requests enabled</p>
5 MSnB	<p><b>Mode Select B for TPM Channel n</b> — When CPWMS = 0, MSnB = 1 configures TPM channel n for edge-aligned PWM mode. For a summary of channel mode and setup controls, refer to <a href="#">Table B-5</a>.</p>
4 MSnA	<p><b>Mode Select A for TPM Channel n</b> — When CPWMS = 0 and MSnB = 0, MSnA configures TPM channel n for input capture mode or output compare mode. Refer to <a href="#">Table B-5</a> for a summary of channel mode and setup controls.</p>
3:2 ELSn[B:A]	<p><b>Edge/Level Select Bits</b> — Depending on the operating mode for the timer channel as set by CPWMS:MSnB:MSnA and shown in <a href="#">Table B-5</a>, these bits select the polarity of the input edge that triggers an input capture event, select the level that will be driven in response to an output compare match, or select the polarity of the PWM output.</p> <p>Setting ELSnB:ELSnA to 0:0 configures the related timer pin as a general-purpose I/O pin unrelated to any timer channel functions. This function is typically used to temporarily disable an input capture channel or to make the timer pin available as a general-purpose I/O pin when the associated timer channel is set up as a software timer that does not require the use of a pin.</p>