

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

E·XFI

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 ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s08dv32amlf

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong



**Section Number** 

## Title

Page

# Chapter 16 Timer Pulse-Width Modulator (S08TPMV3)

16.1	Introduction	
	16.1.1 Features	
	16.1.2 Modes of Operation	
	16.1.3 Block Diagram	
16.2	Signal Description	
	16.2.1 Detailed Signal Descriptions	
16.3	Register Definition	
	16.3.1 TPM Status and Control Register (TPMxSC)	
	16.3.2 TPM-Counter Registers (TPMxCNTH:TPMxCNTL)	
	16.3.3 TPM Counter Modulo Registers (TPMxMODH:TPMxMODL)	
	16.3.4 TPM Channel n Status and Control Register (TPMxCnSC)	
	16.3.5 TPM Channel Value Registers (TPMxCnVH:TPMxCnVL)	
16.4	Functional Description	
	16.4.1 Counter	
	16.4.2 Channel Mode Selection	
16.5	Reset Overview	
	16.5.1 General	
	16.5.2 Description of Reset Operation	
16.6	Interrupts	
	16.6.1 General	
	16.6.2 Description of Interrupt Operation	
16.7	The Differences from TPM v2 to TPM v3	

# Chapter 17 Development Support

Introduction	
17.1.1 Forcing Active Background	
17.1.2 Features	
Background Debug Controller (BDC)	
17.2.1 BKGD Pin Description	
17.2.2 Communication Details	
17.2.3 BDC Commands	
17.2.4 BDC Hardware Breakpoint	354
On-Chip Debug System (DBG)	355
17.3.1 Comparators A and B	355
17.3.2 Bus Capture Information and FIFO Operation	355
17.3.3 Change-of-Flow Information	
17.3.4 Tag vs. Force Breakpoints and Triggers	
17.3.5 Trigger Modes	
17.3.6 Hardware Breakpoints	
	Introduction 17.1.1 Forcing Active Background 17.1.2 Features Background Debug Controller (BDC) 17.2.1 BKGD Pin Description 17.2.2 Communication Details 17.2.3 BDC Commands 17.2.4 BDC Hardware Breakpoint On-Chip Debug System (DBG) 17.3.1 Comparators A and B 17.3.2 Bus Capture Information and FIFO Operation 17.3.3 Change-of-Flow Information 17.3.4 Tag vs. Force Breakpoints and Triggers 17.3.6 Hardware Breakpoints

MC9S08DV60 Series Data Sheet, Rev 3



#### Chapter 3 Modes of Operation

- <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.



# 4.4 RAM

The MC9S08DV60 Series includes static RAM. The locations in RAM below 0x0100 can be accessed using the more efficient direct addressing mode, and any single bit in this area can be accessed with the bit manipulation instructions (BCLR, BSET, BRCLR, and BRSET). Locating the most frequently accessed program variables in this area of RAM is preferred.

The RAM retains data while the MCU is in low-power wait, stop2, or stop3 mode. At power-on the contents of RAM are uninitialized. RAM data is unaffected by any reset if the supply voltage does not drop below the minimum value for RAM retention ( $V_{RAM}$ ).

For compatibility with M68HC05 MCUs, the HCS08 resets the stack pointer to 0x00FF. In the MC9S08DV60 Series, it is usually best to reinitialize the stack pointer to the top of the RAM so the direct page RAM can be used for frequently accessed RAM variables and bit-addressable program variables. Include the following 2-instruction sequence in your reset initialization routine (where RamLast is equated to the highest address of the RAM in the Freescale Semiconductor equate file).

LDHX #RamLast+1 ;point one past RAM TXS ;SP<-(H:X-1)

When security is enabled, the RAM is considered a secure memory resource and is not accessible through BDM or code executing from non-secure memory. See Section 4.5.9, "Security", for a detailed description of the security feature.

# 4.5 Flash

MC9S08DV60 Series devices include Flash memory intended primarily for program and data storage. In-circuit programming allows the operating program and data to be loaded into Flash after final assembly of the application product. It is possible to program the arrays through the single-wire background debug interface. Because no special voltages are needed for erase and programming operations, in-application programming is also possible through other software-controlled communication paths. For a more detailed discussion of in-circuit and in-application programming, refer to the *HCS08 Family Reference Manual*, *Volume I*, Freescale Semiconductor document order number HCS08RMv1.

# 4.5.1 Features

Features of the Flash include:

- Array size (see Table 1-1 for exact array sizes)
- Flash sector size: 768 bytes
- Single power supply program and erase
- Command interface for fast program and erase operation
- Up to 100,000 program/erase cycles at typical voltage and temperature
- Flexible block protection and vector redirection
- Security feature for Flash and RAM
- Burst programming capability
- Sector erase abort

MC9S08DV60 Series Data Sheet, Rev 3



**Chapter 4 Memory** 



Figure 4-3. Burst Program Flowchart



FPS	Address Area Protected	Memory Size Protected (bytes)	Number of Sectors Protected
0x1B	0x2800-0xFFFF	54K	72
0x1A	0x2200-0xFFFF	55.5K	74
0x19	0x1C00-0xFFFF	57K	76
0x18–0x00	0x0000-0xFFFF	64K	86

Table 4-13. Flash Block Protection (continued)

## 4.5.10.5 Flash Status Register (FSTAT)



### Figure 4-9. Flash Status Register (FSTAT)

### Table 4-14. FSTAT Register Field Descriptions

Field	Description
7 FCBEF	<ul> <li>Command Buffer Empty Flag — The FCBEF bit is used to launch commands. It also indicates that the command buffer is empty so that a new command sequence can be executed when performing burst programming. The FCBEF bit is cleared by writing a 1 to it or when a burst program command is transferred to the array for programming. Only burst program commands can be buffered.</li> <li>0 Command buffer is full (not ready for additional commands).</li> <li>1 A new burst program command can be written to the command buffer.</li> </ul>
6 FCCF	<ul> <li>Command Complete Flag — FCCF is set automatically when the command buffer is empty and no command is being processed. FCCF is cleared automatically when a new command is started (by writing 1 to FCBEF to register a command). Writing to FCCF has no meaning or effect.</li> <li>Command in progress</li> <li>All commands complete</li> </ul>
5 FPVIOL	<ul> <li>Protection Violation Flag — FPVIOL is set automatically when a command that attempts to erase or program a location in a protected block is launched (the erroneous command is ignored). FPVIOL is cleared by writing a 1 to FPVIOL.</li> <li>0 No protection violation.</li> <li>1 An attempt was made to erase or program a protected location.</li> </ul>



Chapter 5 Resets, Interrupts, and General System Control

# 5.4 Computer Operating Properly (COP) Watchdog

The COP watchdog is intended to force a system reset when the application software fails to execute as expected. To prevent a system reset from the COP timer (when it is enabled), application software must reset the COP counter periodically. If the application program gets lost and fails to reset the COP counter before it times out, a system reset is generated to force the system back to a known starting point.

After any reset, the COP watchdog is enabled (see Section 5.8.4, "System Options Register 1 (SOPT1)," for additional information). If the COP watchdog is not used in an application, it can be disabled by clearing COPT bits in SOPT1.

The COP counter is reset by writing 0x55 and 0xAA (in this order) to the address of SRS during the selected timeout period. Writes do not affect the data in the read-only SRS. As soon as the write sequence is done, the COP timeout period is restarted. If the program fails to do this during the time-out period, the MCU will reset. Also, if any value other than 0x55 or 0xAA is written to SRS, the MCU is immediately reset.

The COPCLKS bit in SOPT2 (see Section 5.8.5, "System Options Register 2 (SOPT2)," for additional information) selects the clock source used for the COP timer. The clock source options are either the bus clock or an internal 1-kHz clock source. With each clock source, there are three associated time-outs controlled by the COPT bits in SOPT1. Table 5-6 summaries the control functions of the COPCLKS and COPT bits. The COP watchdog defaults to operation from the 1-kHz clock source and the longest time-out (2<sup>10</sup> cycles).

When the bus clock source is selected, windowed COP operation is available by setting COPW in the SOPT2 register. In this mode, writes to the SRS register to clear the COP timer must occur in the last 25% of the selected timeout period. A premature write immediately resets the MCU. When the 1-kHz clock source is selected, windowed COP operation is not available.

The COP counter is initialized by the first writes to the SOPT1 and SOPT2 registers and after any system reset. Subsequent writes to SOPT1 and SOPT2 have no effect on COP operation. Even if the application will use the reset default settings of COPT, COPCLKS, and COPW bits, the user should write to the write-once SOPT1 and SOPT2 registers during reset initialization to lock in the settings. This will prevent accidental changes if the application program gets lost.

The write to SRS that services (clears) the COP counter should not be placed in an interrupt service routine (ISR) because the ISR could continue to be executed periodically even if the main application program fails.

If the bus clock source is selected, the COP counter does not increment while the MCU is in background debug mode or while the system is in stop mode. The COP counter resumes when the MCU exits background debug mode or stop mode.

If the 1-kHz clock source is selected, the COP counter is re-initialized to zero upon entry to either background debug mode or stop mode and begins from zero upon exit from background debug mode or stop mode.



Chapter 6 Parallel Input/Output Control

# 6.5.2.5 Port B Drive Strength Selection Register (PTBDS)



Figure 6-15. Drive Strength Selection for Port B Register (PTBDS)

### Table 6-13. PTBDS Register Field Descriptions

Field	Description
7:0 PTBDS[7:0]	<ul> <li>Output Drive Strength Selection for Port B Bits — Each of these control bits selects between low and high output drive for the associated PTB pin. For port B pins that are configured as inputs, these bits have no effect.</li> <li>0 Low output drive strength selected for port B bit n.</li> <li>1 High output drive strength selected for port B bit n.</li> </ul>

## 6.5.2.6 Port B Interrupt Status and Control Register (PTBSC)



## Figure 6-16. Port B Interrupt Status and Control Register (PTBSC)

### Table 6-14. PTBSC Register Field Descriptions

Field	Description
3 PTBIF	<ul> <li>Port B Interrupt Flag — PTBIF indicates when a Port B interrupt is detected. Writes have no effect on PTBIF.</li> <li>0 No Port B interrupt detected.</li> <li>1 Port B interrupt detected.</li> </ul>
2 PTBACK	<b>Port B Interrupt Acknowledge</b> — Writing a 1 to PTBACK is part of the flag clearing mechanism. PTBACK always reads as 0.
1 PTBIE	<ul> <li>Port B Interrupt Enable — PTBIE determines whether a port B interrupt is requested.</li> <li>0 Port B interrupt request not enabled.</li> <li>1 Port B interrupt request enabled.</li> </ul>
0 PTBMOD	<ul> <li>Port B Detection Mode — PTBMOD (along with the PTBES bits) controls the detection mode of the port B interrupt pins.</li> <li>0 Port B pins detect edges only.</li> <li>1 Port B pins detect both edges and levels.</li> </ul>



## 6.5.4 Port D Registers

Port D is controlled by the registers listed below.

## 6.5.4.1 Port D Data Register (PTDD)



### Figure 6-24. Port D Data Register (PTDD)

#### Table 6-22. PTDD Register Field Descriptions

Field	Description
7:0 PTDD[7:0]	Port D Data Register Bits — For port D pins that are inputs, reads return the logic level on the pin. For port D 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 D pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTDD 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.

## 6.5.4.2 Port D Data Direction Register (PTDDD)

	7	6	5	4	3	2	1	0
R	PTDDD7	PTDDD6	PTDDD5	PTDDD4	PTDDD3	PTDDD2	PTDDD1	PTDDD0
W			_		-			_
Reset:	0	0	0	0	0	0	0	0

### Figure 6-25. Port D Data Direction Register (PTDDD)

### Table 6-23. PTDDD Register Field Descriptions

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



# 6.5.5.5 Port E Drive Strength Selection Register (PTEDS)

_	7	6	5	4	3	2	1	0
R W	PTEDS7	PTEDS6	PTEDS5	PTEDS4	PTEDS3	PTEDS2	PTEDS1 <sup>1</sup>	PTEDS0
Reset:	0	0	0	0	0	0	0	0

Figure 6-36. Drive Strength Selection for Port E Register (PTEDS)

<sup>1</sup> PTEDS1 has no effect on the input-only PTE1 pin.

Field	Description
7:0 PTEDS[7:0]	<ul> <li>Output Drive Strength Selection for Port E Bits — Each of these control bits selects between low and high output drive for the associated PTE pin. For port E pins that are configured as inputs, these bits have no effect.</li> <li>0 Low output drive strength selected for port E bit n.</li> <li>1 High output drive strength selected for port E bit n.</li> </ul>



Chapter 6 Parallel Input/Output Control

# 6.5.7.3 Port G Pull Enable Register (PTGPE)



### Figure 6-44. Internal Pull Enable for Port G Register (PTGPE)

### Table 6-42. PTGPE Register Field Descriptions

Field	Description
5:0 PTGPE[5:0]	Internal Pull Enable for Port G Bits — Each of these control bits determines if the internal pull-up device is enabled for the associated PTG pin. For port G pins that are configured as outputs, these bits have no effect and the internal pull devices are disabled. 0 Internal pull-up device disabled for port G bit n.

## NOTE

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

## 6.5.7.4 Port G Slew Rate Enable Register (PTGSE)



### Figure 6-45. Slew Rate Enable for Port G Register (PTGSE)

### Table 6-43. PTGSE Register Field Descriptions

Field	Description
5:0 PTGSE[5:0]	<ul> <li>Output Slew Rate Enable for Port G Bits — Each of these control bits determines if the output slew rate control is enabled for the associated PTG pin. For port G pins that are configured as inputs, these bits have no effect.</li> <li>Output slew rate control disabled for port G bit n.</li> <li>Output slew rate control enabled for port G bit n.</li> </ul>

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.



Chapter 8 Multi-Purpose Clock Generator (S08MCGV1)



Field	Description
1 SLPAK	<ul> <li>Sleep Mode Acknowledge — This flag indicates whether the MSCAN module has entered sleep mode (see Section 12.5.5.4, "MSCAN Sleep Mode"). It is used as a handshake flag for the SLPRQ sleep mode request. Sleep mode is active when SLPRQ = 1 and SLPAK = 1. Depending on the setting of WUPE, the MSCAN will clear the flag if it detects activity on the CAN bus while in sleep mode.CPU clearing the SLPRQ bit will also reset the SLPAK bit.</li> <li>0 Running — The MSCAN operates normally</li> <li>1 Sleep mode active — The MSCAN has entered sleep mode</li> </ul>
0 INITAK	Initialization Mode Acknowledge — This flag indicates whether the MSCAN module is in initialization mode (see Section 12.5.5.5, "MSCAN Initialization Mode"). It is used as a handshake flag for the INITRQ initialization mode request. Initialization mode is active when INITRQ = 1 and INITAK = 1. The registers CANCTL1, CANBTR0, CANBTR1, CANIDAC, CANIDAR0–CANIDAR7, and CANIDMR0–CANIDMR7 can be written only by the CPU when the MSCAN is in initialization mode. 0 Running — The MSCAN operates normally 1 Initialization mode active — The MSCAN is in initialization mode

### Table 12-2. CANCTL1 Register Field Descriptions (continued)

## 12.3.3 MSCAN Bus Timing Register 0 (CANBTR0)

The CANBTR0 register configures various CAN bus timing parameters of the MSCAN module.

	7	6	5	4	3	2	1	0
R W	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
Reset:	0	0	0	0	0	0	0	0

Figure 12-6. MSCAN Bus Timing Register 0 (CANBTR0)

### Read: Anytime Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1)

### Table 12-3. CANBTR0 Register Field Descriptions

Field	Description
7:6 SJW[1:0]	<b>Synchronization Jump Width</b> — The synchronization jump width defines the maximum number of time quanta (Tq) clock cycles a bit can be shortened or lengthened to achieve resynchronization to data transitions on the CAN bus (see Table 12-4).
5:0 BRP[5:0]	<b>Baud Rate Prescaler</b> — These bits determine the time quanta (Tq) clock which is used to build up the bit timing (see Table 12-5).

#### Table 12-4. Synchronization Jump Width

SJW1	SJW0	Synchronization Jump Width
0	0	1 Tq clock cycle
0	1	2 Tq clock cycles
1	0	3 Tq clock cycles
1	1	4 Tq clock cycles



Chapter 12 Freescale's Controller Area Network (S08MSCANV1)

field of the CAN frame, is received into the next available RxBG. If the MSCAN receives an invalid message in its RxBG (wrong identifier, transmission errors, etc.) the actual contents of the buffer will be over-written by the next message. The buffer will then not be shifted into the FIFO.

When the MSCAN module is transmitting, the MSCAN receives its own transmitted messages into the background receive buffer, RxBG, but does not shift it into the receiver FIFO, generate a receive interrupt, or acknowledge its own messages on the CAN bus. The exception to this rule is in loopback mode (see Section 12.3.2, "MSCAN Control Register 1 (CANCTL1)") where the MSCAN treats its own messages exactly like all other incoming messages. The MSCAN receives its own transmitted messages in the event that it loses arbitration. If arbitration is lost, the MSCAN must be prepared to become a receiver.

An overrun condition occurs when all receive message buffers in the FIFO are filled with correctly received messages with accepted identifiers and another message is correctly received from the CAN bus with an accepted identifier. The latter message is discarded and an error interrupt with overrun indication is generated if enabled (see Section 12.5.7.5, "Error Interrupt"). The MSCAN remains able to transmit messages while the receiver FIFO is full, but all incoming messages are discarded. As soon as a receive buffer in the FIFO is available again, new valid messages will be accepted.

# 12.5.3 Identifier Acceptance Filter

The MSCAN identifier acceptance registers (see Section 12.3.11, "MSCAN Identifier Acceptance Control Register (CANIDAC)") define the acceptable patterns of the standard or extended identifier (ID[10:0] or ID[28:0]). Any of these bits can be marked 'don't care' in the MSCAN identifier mask registers (see Section 12.3.16, "MSCAN Identifier Mask Registers (CANIDMR0–CANIDMR7)").

A filter hit is indicated to the application software by a set receive buffer full flag (RXF = 1) and three bits in the CANIDAC register (see Section 12.3.11, "MSCAN Identifier Acceptance Control Register (CANIDAC)"). These identifier hit flags (IDHIT[2:0]) clearly identify the filter section that caused the acceptance. They simplify the application software's task to identify the cause of the receiver interrupt. If more than one hit occurs (two or more filters match), the lower hit has priority.

A very flexible programmable generic identifier acceptance filter has been introduced to reduce the CPU interrupt loading. The filter is programmable to operate in four different modes (see Bosch CAN 2.0A/B protocol specification):

- Two identifier acceptance filters, each to be applied to:
  - The full 29 bits of the extended identifier and to the following bits of the CAN 2.0B frame:
    - Remote transmission request (RTR)
    - Identifier extension (IDE)
    - Substitute remote request (SRR)
  - The 11 bits of the standard identifier plus the RTR and IDE bits of the CAN 2.0A/B messages<sup>1</sup>. This mode implements two filters for a full length CAN 2.0B compliant extended identifier. Figure 12-39 shows how the first 32-bit filter bank (CANIDAR0–CANIDAR3, CANIDMR0–CANIDMR3) produces a filter 0 hit. Similarly, the second filter bank (CANIDAR4–CANIDAR7, CANIDMR4–CANIDMR7) produces a filter 1 hit.

<sup>1.</sup> Although this mode can be used for standard identifiers, it is recommended to use the four or eight identifier acceptance filters for standard identifiers



Chapter 13 Serial Peripheral Interface (S08SPIV3)



Figure 13-3. SPI Module Block Diagram

## 13.1.3 SPI Baud Rate Generation

As shown in Figure 13-4, the clock source for the SPI baud rate generator is the bus clock. The three prescale bits (SPPR2:SPPR1:SPPR0) choose a prescale divisor of 1, 2, 3, 4, 5, 6, 7, or 8. The three rate select bits (SPR2:SPR1:SPR0) divide the output of the prescaler stage by 2, 4, 8, 16, 32, 64, 128, or 256 to get the internal SPI master mode bit-rate clock.



Writing 0 to TE does not immediately release the pin to be a general-purpose I/O pin. Any transmit activity that is in progress must first be completed. This includes data characters in progress, queued idle characters, and queued break characters.

## 14.3.2.1 Send Break and Queued Idle

The SBK control bit in SCIxC2 is used to send break characters which were originally used to gain the attention of old teletype receivers. Break characters are a full character time of logic 0 (10 bit times including the start and stop bits). A longer break of 13 bit times can be enabled by setting BRK13 = 1. Normally, a program would wait for TDRE to become set to indicate the last character of a message has moved to the transmit shifter, then write 1 and then write 0 to the SBK bit. This action queues a break character to be sent as soon as the shifter is available. If SBK is still 1 when the queued break moves into the shifter (synchronized to the baud rate clock), an additional break character is queued. If the receiving device is another Freescale Semiconductor SCI, the break characters will be received as 0s in all eight data bits and a framing error (FE = 1) occurs.

When idle-line wakeup is used, a full character time of idle (logic 1) is needed between messages to wake up any sleeping receivers. Normally, a program would wait for TDRE to become set to indicate the last character of a message has moved to the transmit shifter, then write 0 and then write 1 to the TE bit. This action queues an idle character to be sent as soon as the shifter is available. As long as the character in the shifter does not finish while TE = 0, the SCI transmitter never actually releases control of the TxD pin. If there is a possibility of the shifter finishing while TE = 0, set the general-purpose I/O controls so the pin that is shared with TxD is an output driving a logic 1. This ensures that the TxD line will look like a normal idle line even if the SCI loses control of the port pin between writing 0 and then 1 to TE.

The length of the break character is affected by the BRK13 and M bits as shown below.

BRK13	М	Break Character Length
0	0	10 bit times
0	1	11 bit times
1	0	13 bit times
1	1	14 bit times

Table 14-9. Break Character Length

## 14.3.3 Receiver Functional Description

In this section, the receiver block diagram (Figure 14-3) is used as a guide for the overall receiver functional description. Next, the data sampling technique used to reconstruct receiver data is described in more detail. Finally, two variations of the receiver wakeup function are explained.

The receiver input is inverted by setting RXINV = 1. The receiver is enabled by setting the RE bit in SCIxC2. Character frames consist of a start bit of logic 0, eight (or nine) data bits (LSB first), and a stop bit of logic 1. For information about 9-bit data mode, refer to Section 14.3.5.1, "8- and 9-Bit Data Modes." For the remainder of this discussion, we assume the SCI is configured for normal 8-bit data mode.

After receiving the stop bit into the receive shifter, and provided the receive data register is not already full, the data character is transferred to the receive data register and the receive data register full (RDRF) status



#### Chapter 16 Timer Pulse-Width Modulator (S08TPMV3)



o - VDD and VSS pins are each internally connected to two pads in 32-pin package

Pin not connected in 48-pin and 32-pin packages □ - Pin not connected in 32-pin package

Figure 16-1. MC9S08DV60 Block Diagram

MC9S08DV60 Series Data Sheet, Rev 3



## 16.2.1.1 EXTCLK — External Clock Source

Control bits in the timer status and control register allow the user to select nothing (timer disable), the bus-rate clock (the normal default source), a crystal-related clock, or an external clock as the clock which drives the TPM prescaler and subsequently the 16-bit TPM counter. The external clock source is synchronized in the TPM. The bus clock clocks the synchronizer; the frequency of the external source must be no more than one-fourth the frequency of the bus-rate clock, to meet Nyquist criteria and allowing for jitter.

The external clock signal shares the same pin as a channel I/O pin, so the channel pin will not be usable for channel I/O function when selected as the external clock source. It is the user's responsibility to avoid such settings. If this pin is used as an external clock source (CLKSB:CLKSA = 1:1), the channel can still be used in output compare mode as a software timer (ELSnB:ELSnA = 0:0).

## 16.2.1.2 TPMxCHn — TPM Channel n I/O Pin(s)

Each TPM channel is associated with an I/O pin on the MCU. The function of this pin depends on the channel configuration. The TPM pins share with general purpose I/O pins, where each pin has a port data register bit, and a data direction control bit, and the port has optional passive pullups which may be enabled whenever a port pin is acting as an input.

The TPM channel does not control the I/O pin when (ELSnB:ELSnA = 0:0) or when (CLKSB:CLKSA = 0:0) so it normally reverts to general purpose I/O control. When CPWMS = 1 (and ELSnB:ELSnA not = 0:0), all channels within the TPM are configured for center-aligned PWM and the TPMxCHn pins are all controlled by the TPM system. When CPWMS=0, the MSnB:MSnA control bits determine whether the channel is configured for input capture, output compare, or edge-aligned PWM.

When a channel is configured for input capture (CPWMS=0, MSnB:MSnA = 0:0 and ELSnB:ELSnA not = 0:0), the TPMxCHn pin is forced to act as an edge-sensitive input to the TPM. ELSnB:ELSnA control bits determine what polarity edge or edges will trigger input-capture events. A synchronizer based on the bus clock is used to synchronize input edges to the bus clock. This implies the minimum pulse width—that can be reliably detected—on an input capture pin is four bus clock periods (with ideal clock pulses as near as two bus clocks can be detected). TPM uses this pin as an input capture input to override the port data and data direction controls for the same pin.

When a channel is configured for output compare (CPWMS=0, MSnB:MSnA = 0:1 and ELSnB:ELSnA not = 0:0), the associated data direction control is overridden, the TPMxCHn pin is considered an output controlled by the TPM, and the ELSnB:ELSnA control bits determine how the pin is controlled. The remaining three combinations of ELSnB:ELSnA determine whether the TPMxCHn pin is toggled, cleared, or set each time the 16-bit channel value register matches the timer counter.

When the output compare toggle mode is initially selected, the previous value on the pin is driven out until the next output compare event—then the pin is toggled.



Chapter 16 Timer/PWM Module (S08TPMV3)





When BDM is active, the timer counter is frozen (this is the value that will be read by user); the coherency mechanism is frozen such that the buffer latches remain in the state they were in when the BDM became active, even if one or both counter halves are read while BDM is active. This assures that if the user was in the middle of reading a 16-bit register when BDM became active, it will read the appropriate value from the other half of the 16-bit value after returning to normal execution.

In BDM mode, writing any value to TPMxSC, TPMxCNTH or TPMxCNTL registers resets the read coherency mechanism of the TPMxCNTH:L registers, regardless of the data involved in the write.

# 16.3.3 TPM Counter Modulo Registers (TPMxMODH:TPMxMODL)

The read/write TPM modulo registers contain the modulo value for the TPM counter. After the TPM counter reaches the modulo value, the TPM counter resumes counting from 0x0000 at the next clock, and the overflow flag (TOF) becomes set. Writing to TPMxMODH or TPMxMODL inhibits the TOF bit and overflow interrupts until the other byte is written. Reset sets the TPM counter modulo registers to 0x0000 which results in a free running timer counter (modulo disabled).

Writing to either byte (TPMxMODH or TPMxMODL) latches the value into a buffer and the registers are updated with the value of their write buffer according to the value of CLKSB:CLKSA bits, so:

- If (CLKSB:CLKSA = 0:0), then the registers are updated when the second byte is written
- If (CLKSB:CLKSA not = 0:0), then the registers are updated after both bytes were written, and the TPM counter changes from (TPMxMODH:TPMxMODL 1) to (TPMxMODH:TPMxMODL). If the TPM counter is a free-running counter, the update is made when the TPM counter changes from 0xFFFE to 0xFFFF

The latching mechanism may be manually reset by writing to the TPMxSC address (whether BDM is active or not).

When BDM is active, the coherency mechanism is frozen (unless reset by writing to TPMxSC register) such that the buffer latches remain in the state they were in when the BDM became active, even if one or both halves of the modulo register are written while BDM is active. Any write to the modulo registers bypasses the buffer latches and directly writes to the modulo register while BDM is active.

	7	6	5	4	3	2	1	0
R W	Bit 15	14	13	12	11	10	9	Bit 8
Reset	0	0	0	0	0	0	0	0



MC9S08DV60 Series Data Sheet, Rev 3



Command Mnemonic	Active BDM/ Non-intrusive	Coding Structure	Description
SYNC	Non-intrusive	n/a <sup>1</sup>	Request a timed reference pulse to determine target BDC communication speed
ACK_ENABLE	Non-intrusive	D5/d	Enable acknowledge protocol. Refer to Freescale document order no. HCS08RMv1/D.
ACK_DISABLE	Non-intrusive	D6/d	Disable acknowledge protocol. Refer to Freescale document order no. HCS08RMv1/D.
BACKGROUND	Non-intrusive	90/d	Enter active background mode if enabled (ignore if ENBDM bit equals 0)
READ_STATUS	Non-intrusive	E4/SS	Read BDC status from BDCSCR
WRITE_CONTROL	Non-intrusive	C4/CC	Write BDC controls in BDCSCR
READ_BYTE	Non-intrusive	E0/AAAA/d/RD	Read a byte from target memory
READ_BYTE_WS	Non-intrusive	E1/AAAA/d/SS/RD	Read a byte and report status
READ_LAST	Non-intrusive	E8/SS/RD	Re-read byte from address just read and report status
WRITE_BYTE	Non-intrusive	C0/AAAA/WD/d	Write a byte to target memory
WRITE_BYTE_WS	Non-intrusive	C1/AAAA/WD/d/SS	Write a byte and report status
READ_BKPT	Non-intrusive	E2/RBKP	Read BDCBKPT breakpoint register
WRITE_BKPT	Non-intrusive	C2/WBKP	Write BDCBKPT breakpoint register
GO	Active BDM	08/d	Go to execute the user application program starting at the address currently in the PC
TRACE1	Active BDM	10/d	Trace 1 user instruction at the address in the PC, then return to active background mode
TAGGO	Active BDM	18/d	Same as GO but enable external tagging (HCS08 devices have no external tagging pin)
READ_A	Active BDM	68/d/RD	Read accumulator (A)
READ_CCR	Active BDM	69/d/RD	Read condition code register (CCR)
READ_PC	Active BDM	6B/d/RD16	Read program counter (PC)
READ_HX	Active BDM	6C/d/RD16	Read H and X register pair (H:X)
READ_SP	Active BDM	6F/d/RD16	Read stack pointer (SP)
READ_NEXT	Active BDM	70/d/RD	Increment H:X by one then read memory byte located at H:X
READ_NEXT_WS	Active BDM	71/d/SS/RD	Increment H:X by one then read memory byte located at H:X. Report status and data.
WRITE_A	Active BDM	48/WD/d	Write accumulator (A)
WRITE_CCR	Active BDM	49/WD/d	Write condition code register (CCR)
WRITE_PC	Active BDM	4B/WD16/d	Write program counter (PC)
WRITE_HX	Active BDM	4C/WD16/d	Write H and X register pair (H:X)
WRITE_SP	Active BDM	4F/WD16/d	Write stack pointer (SP)
WRITE_NEXT	Active BDM	50/WD/d	Increment H:X by one, then write memory byte located at H:X
WRITE_NEXT_WS	Active BDM	51/WD/d/SS	Increment H:X by one, then write memory byte located at H:X. Also report status.

<sup>1</sup> The SYNC command is a special operation that does not have a command code.







Figure A-5. Timer External Clock





## A.12.3 MSCAN

Table A-15. MSCAN Wake-up Pulse Characteristi
---

Num	С	Rating	Symbol	Min	Тур	Мах	Unit
1	D	MSCAN Wake-up dominant pulse filtered	t <sub>WUP</sub>			2	μs
2	D	MSCAN Wake-up dominant pulse pass	t <sub>WUP</sub>	5	—	—	μs