



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

Product Status	Obsolete
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	CANbus, I ² C, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 10x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08dv16aclcr

Email: info@E-XFL.COM

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



Section Number

Chapter 14 Serial Communications Interface (S08SCIV4)

Title

14.3.3 Receiver Functional Descr	ription
14.3.4 Interrupts and Status Flag	s
14.3.5 Additional SCI Functions	

Chapter 15 Real-Time Counter (S08RTCV1)

Introduction	
15.1.1 RTC Clock Signal Names	
15.1.2 Features	
15.1.3 Modes of Operation	
15.1.4 Block Diagram	
External Signal Description	
Register Definition	
15.3.1 RTC Status and Control Register (RTCSC)	
15.3.2 RTC Counter Register (RTCCNT)	
15.3.3 RTC Modulo Register (RTCMOD)	
Functional Description	
15.4.1 RTC Operation Example	
Initialization/Application Information	
	Introduction

MC9S08DV60 Series Data Sheet, Rev 3

Page



Section Number

Title

Page

B.3.2	Channel Mode Selection	
B.3.3	Center-Aligned PWM Mode	
TPM In	terrupts	
B.4.1	Clearing Timer Interrupt Flags	
B.4.2	Timer Overflow Interrupt Description	
B.4.3	Channel Event Interrupt Description	
B.4.4	PWM End-of-Duty-Cycle Events	
	B.3.2 B.3.3 TPM In B.4.1 B.4.2 B.4.3 B.4.4	 B.3.2 Channel Mode Selection B.3.3 Center-Aligned PWM Mode TPM Interrupts B.4.1 Clearing Timer Interrupt Flags B.4.2 Timer Overflow Interrupt Description B.4.3 Channel Event Interrupt Description B.4.4 PWM End-of-Duty-Cycle Events

Appendix C Ordering Information and Mechanical Drawings

C.1	Ordering Information	
	C.1.1 MC9S08DV60 Series Devices	
C.2	Mechanical Drawings	



2.2.1 Power

 V_{DD} and V_{SS} are the primary power supply pins for the MCU. This voltage source supplies power to all I/O buffer circuitry and to an internal voltage regulator. The internal voltage regulator provides regulated lower-voltage source to the CPU and other internal circuitry of the MCU.

Typically, application systems have two separate capacitors across the power pins. In this case, there should be a bulk electrolytic capacitor, such as a 10- μ F tantalum capacitor, to provide bulk charge storage for the overall system and a 0.1- μ F ceramic bypass capacitor located as near to the MCU power pins as practical to suppress high-frequency noise. The MC9S08DV60 Series has two V_{DD} pins except on the 32-pin package. Each pin must have a bypass capacitor for best noise suppression.

 V_{DDA} and V_{SSA} are the analog power supply pins for the MCU. This voltage source supplies power to the ADC module. A 0.1- μ F ceramic bypass capacitor should be located as near to the MCU power pins as practical to suppress high-frequency noise.

2.2.2 Oscillator

Immediately after reset, the MCU uses an internally generated clock provided by the multi-purpose clock generator (MCG) module. For more information on the MCG, see Chapter 8, "Multi-Purpose Clock Generator (S08MCGV1)."

The oscillator (XOSC) in this MCU is a Pierce oscillator that can accommodate a crystal or ceramic resonator. Rather than a crystal or ceramic resonator, an external oscillator can be connected to the EXTAL input pin.

Refer to Figure 2-4 for the following discussion. R_S (when used) and R_F should be low-inductance resistors such as carbon composition resistors. Wire-wound resistors and some metal film resistors have too much inductance. C1 and C2 normally should be high-quality ceramic capacitors that are specifically designed for high-frequency applications.

 R_F is used to provide a bias path to keep the EXTAL input in its linear range during crystal startup; its value is not generally critical. Typical systems use 1 M Ω to 10 M Ω . Higher values are sensitive to humidity, and lower values reduce gain and (in extreme cases) could prevent startup.

C1 and C2 are typically in the 5-pF to 25-pF range and are chosen to match the requirements of a specific crystal or resonator. Be sure to take into account printed circuit board (PCB) capacitance and MCU pin capacitance when selecting C1 and C2. The crystal manufacturer typically specifies a load capacitance which is the series combination of C1 and C2 (which are usually the same size). As a first-order approximation, use 10 pF as an estimate of combined pin and PCB capacitance for each oscillator pin (EXTAL and XTAL).

2.2.3 **RESET**

RESET is a dedicated pin with a pull-up device built in. It has input hysteresis, a high current output driver, and no output slew rate control. Internal power-on reset and low-voltage reset circuitry typically make external reset circuitry unnecessary. This pin is normally connected to the standard 6-pin background debug connector so a development system can directly reset the MCU system. If desired, a manual external reset can be added by supplying a simple switch to ground (pull reset pin low to force a reset).



Chapter 3 Modes of Operation

3.1 Introduction

The operating modes of the MC9S08DV60 Series are described in this chapter. Entry into each mode, exit from each mode, and functionality while in each of the modes are described.

3.2 Features

- Active background mode for code development
- Wait mode CPU shuts down to conserve power; system clocks are running and full regulation is maintained
- Stop modes System clocks are stopped and voltage regulator is in standby
 - Stop3 All internal circuits are powered for fast recovery
 - Stop2 Partial power down of internal circuits; RAM content is retained

3.3 Run Mode

This is the normal operating mode for the MC9S08DV60 Series. This mode is selected when the BKGD/MS pin is high at the rising edge of reset. In this mode, the CPU executes code from internal memory with execution beginning at the address fetched from memory at 0xFFFE–0xFFFF after reset.

3.4 Active Background Mode

The active background mode functions are managed through the background debug controller (BDC) in the HCS08 core. The BDC, together with the on-chip debug module (DBG), provide the means for analyzing MCU operation during software development.

Active background mode is entered in any of five ways:

- When the BKGD/MS pin is low at the rising edge of reset
- When a BACKGROUND command is received through the BKGD/MS pin
- When a BGND instruction is executed
- When encountering a BDC breakpoint
- When encountering a DBG breakpoint

After entering active background mode, the CPU is held in a suspended state waiting for serial background commands rather than executing instructions from the user application program.



Chapter 6 Parallel Input/Output Control

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	Internal Pull Enable for Port E Bits — Each of these control bits determines if the internal pull-up device is
PTEPE[7:0]	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.
	0 Internal pull-up device disabled for port E bit n.
	1 Internal pull-up device enabled for port E bit n.

NOTE

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

6.5.5.4 Port E Slew Rate Enable Register (PTESE)

	7	6	5	4	3	2	1	0
R W	PTESE7	PTESE6	PTESE5	PTESE4	PTESE3	PTESE2	PTESE1 ¹	PTESE0
Reset:	0	0	0	0	0	0	0	0

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

¹ PTESE1 has no effect on the input-only PTE1 pin.

Table 6-33. PTESE Register Field Descriptions

Field	Description
7:0 PTESE[7:0]	 Output Slew Rate Enable for Port E Bits — 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. Output slew rate control disabled for port E bit n. Output slew rate control enabled for port E bit n.

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.



6.5.7 Port G Registers

Port G is controlled by the registers listed below.

6.5.7.1 Port G Data Register (PTGD)



Figure 6-42. Port G Data Register (PTGD)

Table 6-40. PTGD Register Field Descriptions

Field	Description
5:0 PTGD[5:0]	Port G Data Register Bits — For port G pins that are inputs, reads return the logic level on the pin. For port G 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 G pins that are configured as outputs, the logic level is driven out the corresponding MCU pin. Reset forces PTGD 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 disabled.

6.5.7.2 Port G Data Direction Register (PTGDD)



Figure 6-43. Port G Data Direction Register (PTGDD)

Table 6-41. PTGDD Register Field Descriptions

Field	Description
5:0 PTGDD[5:0]	 Data Direction for Port G Bits — These read/write bits control the direction of port G pins and what is read for PTGD reads. 0 Input (output driver disabled) and reads return the pin value. 1 Output driver enabled for port G bit n and PTGD reads return the contents of PTGDn.



6.5.7.5 Port G Drive Strength Selection Register (PTGDS)



Figure 6-46. Drive Strength Selection for Port G Register (PTGDS)

Table 6-44. PTGDS Register Field Descriptions

Field	Description
5:0 PTGDS[5:0	 Output Drive Strength Selection for Port G Bits — Each of these control bits selects between low and high output drive for the associated PTG pin. For port G pins that are configured as inputs, these bits have no effect. 0 Low output drive strength selected for port G bit n. 1 High output drive strength selected for port G bit n.



Chapter 7 Central Processor Unit (S08CPUV3)

7.1 Introduction

This section provides summary information about the registers, addressing modes, and instruction set of the CPU of the HCS08 Family. For a more detailed discussion, refer to the *HCS08 Family Reference Manual, volume 1,* Freescale Semiconductor document order number HCS08RMV1/D.

The HCS08 CPU is fully source- and object-code-compatible with the M68HC08 CPU. Several instructions and enhanced addressing modes were added to improve C compiler efficiency and to support a new background debug system which replaces the monitor mode of earlier M68HC08 microcontrollers (MCU).

7.1.1 Features

Features of the HCS08 CPU include:

- Object code fully upward-compatible with M68HC05 and M68HC08 Families
- All registers and memory are mapped to a single 64-Kbyte address space
- 16-bit stack pointer (any size stack anywhere in 64-Kbyte address space)
- 16-bit index register (H:X) with powerful indexed addressing modes
- 8-bit accumulator (A)
- Many instructions treat X as a second general-purpose 8-bit register
- Seven addressing modes:
 - Inherent Operands in internal registers
 - Relative 8-bit signed offset to branch destination
 - Immediate Operand in next object code byte(s)
 - Direct Operand in memory at 0x0000–0x00FF
 - Extended Operand anywhere in 64-Kbyte address space
 - Indexed relative to H:X Five submodes including auto increment
 - Indexed relative to SP Improves C efficiency dramatically
- Memory-to-memory data move instructions with four address mode combinations
- Overflow, half-carry, negative, zero, and carry condition codes support conditional branching on the results of signed, unsigned, and binary-coded decimal (BCD) operations
- Efficient bit manipulation instructions
- Fast 8-bit by 8-bit multiply and 16-bit by 8-bit divide instructions
- STOP and WAIT instructions to invoke low-power operating modes

MC9S08DV60 Series Data Sheet, Rev 3



Source	Operation	dress lode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR	
		PΑ				V 1 1 H	INZC
SUB #opr8i SUB opr8a SUB opr16a SUB oprx16,X SUB oprx8,X SUB ,X SUB oprx16,SP SUB oprx8,SP	Subtract A \leftarrow (A) – (M)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A0 ii B0 dd C0 hh ll D0 ee ff E0 ff F0 9E D0 ee ff 9E E0 ff	2 3 4 3 3 5 4	pp rpp prpp rpp rfp pprpp prpp	↓11-	- ↓ ↓ ↓
SWI	Software Interrupt PC \leftarrow (PC) + \$0001 Push (PCL); SP \leftarrow (SP) - \$0001 Push (PCH); SP \leftarrow (SP) - \$0001 Push (X); SP \leftarrow (SP) - \$0001 Push (A); SP \leftarrow (SP) - \$0001 Push (CCR); SP \leftarrow (SP) - \$0001 I \leftarrow 1; PCH \leftarrow Interrupt Vector High Byte PCL \leftarrow Interrupt Vector Low Byte	INH	83	11	sssssvvfppp	- 1 1 -	1
ТАР	Transfer Accumulator to CCR $CCR \leftarrow (A)$	INH	84	1	р	\$11\$	¢ ¢ ¢ ¢ ¢
ТАХ	Transfer Accumulator to X (Index Register Low) $X \leftarrow (A)$	INH	97	1	р	- 1 1 -	
ТРА	Transfer CCR to Accumulator $A \leftarrow (CCR)$	INH	85	1	q	- 1 1 -	
TST opr8a TSTA TSTX TST oprx8,X TST oprx8,SP	Test for Negative or Zero (M) - \$00 (A) - \$00 (X) - \$00 (M) - \$00 (M) - \$00 (M) - \$00 (M) - \$00	DIR INH INH IX1 IX SP1	3D dd 4D 5D 6D ff 7D 9E 6D ff	4 1 4 3 5	rfpp p rfpp rfp prfpp	011-	- \$ \$ -
TSX	Transfer SP to Index Reg. H:X \leftarrow (SP) + \$0001	INH	95	2	fp	- 1 1 -	
ТХА	Transfer X (Index Reg. Low) to Accumulator $A \leftarrow (X)$	INH	9F	1	q	- 1 1 -	

Table 7-2. Instruction Set Summary (Sheet 8 of 9)



Chapter 8 Multi-Purpose Clock Generator (S08MCGV1)

8.3.3 MCG Trim Register (MCGTRM)



Figure 8-5. MCG Trim Register (MCGTRM)

Table 8-3. MCG Trim	Register Field	Descriptions
---------------------	----------------	--------------

Field	Description
7:0 TRIM	MCG Trim Setting — Controls the internal reference clock frequency by controlling the internal reference clock period. The TRIM bits are binary weighted (i.e., bit 1 will adjust twice as much as bit 0). Increasing the binary value in TRIM will increase the period, and decreasing the value will decrease the period.
	An additional fine trim bit is available in MCGSC as the FTRIM bit.
	If a TRIM[7:0] 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.



Chapter 8 Multi-Purpose Clock Generator (S08MCGV1)

The table below shows MCGOUT frequency calculations using RDIV, BDIV, and VDIV settings for each clock mode. The bus frequency is equal to MCGOUT divided by 2.

Clock Mode	^f мсgouт ¹	Note
FEI (FLL engaged internal)	(f _{int} * 1024) / B	Typical f _{MCGOUT} = 16 MHz immediately after reset. RDIV bits set to %000.
FEE (FLL engaged external)	(f _{ext} / R *1024) / B	f _{ext} / R must be in the range of 31.25 kHz to 39.0625 kHz
FBE (FLL bypassed external)	f _{ext} / B	f _{ext} / R must be in the range of 31.25 kHz to 39.0625 kHz
FBI (FLL bypassed internal)	f _{int} / B	Typical f _{int} = 32 kHz
PEE (PLL engaged external)	[(f _{ext} / R) * M] / B	f _{ext} / R must be in the range of 1 MHz to 2 MHz
PBE (PLL bypassed external)	f _{ext} / B	f _{ext} / R must be in the range of 1 MHz to 2 MHz
BLPI (Bypassed low power internal)	f _{int} / B	
BLPE (Bypassed low power external)	f _{ext} / B	

Table 8-6. MCGOUT Free	quency Calculation Options
------------------------	----------------------------

R is the reference divider selected by the RDIV bits, B is the bus frequency divider selected by the BDIV bits, and M is the multiplier selected by the VDIV bits.

This section will include 3 mode switching examples using a 4 MHz external crystal. If using an external clock source less than 1 MHz, the MCG should not be configured for any of the PLL modes (PEE and PBE).

8.5.2.1 Example # 1: Moving from FEI to PEE Mode: External Crystal = 4 MHz, Bus Frequency = 8 MHz

In this example, the MCG will move through the proper operational modes from FEI to PEE mode until the 4 MHz crystal reference frequency is set to achieve a bus frequency of 8 MHz. Because the MCG is in FEI mode out of reset, this example also shows how to initialize the MCG for PEE mode out of reset. First, the code sequence will be described. Then a flowchart will be included which illustrates the sequence.

- 1. First, FEI must transition to FBE mode:
 - a) MCGC2 = 0x36 (%00110110)
 - BDIV (bits 7 and 6) set to %00, or divide-by-1
 - RANGE (bit 5) set to 1 because the frequency of 4 MHz is within the high frequency range
 - HGO (bit 4) set to 1 to configure external oscillator for high gain operation
 - EREFS (bit 2) set to 1, because a crystal is being used
 - ERCLKEN (bit 1) set to 1 to ensure the external reference clock is active
 - b) Loop until OSCINIT (bit 1) in MCGSC is 1, indicating the crystal selected by the EREFS bit has been initialized.



Conversion Type	ADICLK	ADLSMP	Max Total Conversion Time
Subsequent continuous 10-bit or 12-bit; $f_{BUS} \ge f_{ADCK}/11$	ХХ	1	40 ADCK cycles

The maximum total conversion time is determined by the clock source chosen and the divide ratio selected. The clock source is selectable by the ADICLK bits, and the divide ratio is specified by the ADIV bits. For example, in 10-bit mode, with the bus clock selected as the input clock source, the input clock divide-by-1 ratio selected, and a bus frequency of 8 MHz, then the conversion time for a single conversion is:

Conversion time = $\frac{23 \text{ ADCK Cyc}}{8 \text{ MHz/1}}$ + $\frac{5 \text{ bus Cyc}}{8 \text{ MHz}}$ = 3.5 ms

Number of bus cycles = 3.5 ms x 8 MHz = 28 cycles

NOTE

The ADCK frequency must be between f_{ADCK} minimum and f_{ADCK} maximum to meet ADC specifications.

10.4.5 Automatic Compare Function

The compare function can be configured to check for an upper or lower limit. After the input is sampled and converted, the result is added to the two's complement of the compare value (ADCCVH and ADCCVL). When comparing to an upper limit (ACFGT = 1), if the result is greater-than or equal-to the compare value, COCO is set. When comparing to a lower limit (ACFGT = 0), if the result is less than the compare value, COCO is set. The value generated by the addition of the conversion result and the two's complement of the compare value is transferred to ADCRH and ADCRL.

Upon completion of a conversion while the compare function is enabled, if the compare condition is not true, COCO is not set and no data is transferred to the result registers. An ADC interrupt is generated upon the setting of COCO if the ADC interrupt is enabled (AIEN = 1).

NOTE

The compare function can monitor the voltage on a channel while the MCU is in wait or stop3 mode. The ADC interrupt wakes the MCU when the compare condition is met.

10.4.6 MCU Wait Mode Operation

Wait mode is a lower power-consumption standby mode from which recovery is fast because the clock sources remain active. If a conversion is in progress when the MCU enters wait mode, it continues until completion. Conversions can be initiated while the MCU is in wait mode by means of the hardware trigger or if continuous conversions are enabled.

The bus clock, bus clock divided by two, and ADACK are available as conversion clock sources while in wait mode. The use of ALTCLK as the conversion clock source in wait is dependent on the definition of



10.6.2.3 Noise-Induced Errors

System noise that occurs during the sample or conversion process can affect the accuracy of the conversion. The ADC accuracy numbers are guaranteed as specified only if the following conditions are met:

- There is a 0.1 μ F low-ESR capacitor from V_{REFH} to V_{REFL}.
- There is a 0.1 μ F low-ESR capacitor from V_{DDAD} to V_{SSAD}.
- If inductive isolation is used from the primary supply, an additional 1 μ F capacitor is placed from V_{DDAD} to V_{SSAD}.
- V_{SSAD} (and V_{REFL} , if connected) is connected to V_{SS} at a quiet point in the ground plane.
- Operate the MCU in wait or stop3 mode before initiating (hardware triggered conversions) or immediately after initiating (hardware or software triggered conversions) the ADC conversion.
 - For software triggered conversions, immediately follow the write to ADCSC1 with a wait instruction or stop instruction.
 - For stop3 mode operation, select ADACK as the clock source. Operation in stop3 reduces V_{DD} noise but increases effective conversion time due to stop recovery.
- There is no I/O switching, input or output, on the MCU during the conversion.

There are some situations where external system activity causes radiated or conducted noise emissions or excessive V_{DD} noise is coupled into the ADC. In these situations, or when the MCU cannot be placed in wait or stop3 or I/O activity cannot be halted, these recommended actions may reduce the effect of noise on the accuracy:

- Place a 0.01 μ F capacitor (C_{AS}) on the selected input channel to V_{REFL} or V_{SSAD} (this improves noise issues, but affects the sample rate based on the external analog source resistance).
- Average the result by converting the analog input many times in succession and dividing the sum of the results. Four samples are required to eliminate the effect of a 1LSB, one-time error.
- Reduce the effect of synchronous noise by operating off the asynchronous clock (ADACK) and averaging. Noise that is synchronous to ADCK cannot be averaged out.

10.6.2.4 Code Width and Quantization Error

The ADC quantizes the ideal straight-line transfer function into 4096 steps (in 12-bit mode). Each step ideally has the same height (1 code) and width. The width is defined as the delta between the transition points to one code and the next. The ideal code width for an N bit converter (in this case N can be 8, 10 or 12), defined as 1LSB, is:

1 Isb =
$$(V_{REFH} - V_{REFL}) / 2^N$$
 Eqn. 10-2

There is an inherent quantization error due to the digitization of the result. For 8-bit or 10-bit conversions the code transitions when the voltage is at the midpoint between the points where the straight line transfer function is exactly represented by the actual transfer function. Therefore, the quantization error will be \pm 1/2 lsb in 8- or 10-bit mode. As a consequence, however, the code width of the first (0x000) conversion is only 1/2 lsb and the code width of the last (0xFF or 0x3FF) is 1.5 lsb.



BRP5	BRP4	BRP3	BRP2	BRP1	BRP0	Prescaler value (P)
0	0	0	0	0	0	1
0	0	0	0	0	1	2
0	0	0	0	1	0	3
0	0	0	0	1	1	4
:	:	:	:	:	:	:
1	1	1	1	1	1	64

Table 12-5. Baud Rate Prescaler

12.3.4 MSCAN Bus Timing Register 1 (CANBTR1)

The CANBTR1 register configures various CAN bus timing parameters of the MSCAN module.

	7	6	5	4	3	2	1	0
R W	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
Reset:	0	0	0	0	0	0	0	0

Figure 12-7. MSCAN Bus Timing Register 1 (CANBTR1)

Read: Anytime

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

Table 12-6. CANBTR1 Register Field Descriptions

Field	Description
7 SAMP	 Sampling — This bit determines the number of CAN bus samples taken per bit time. One sample per bit. Three samples per bit¹. If SAMP = 0, the resulting bit value is equal to the value of the single bit positioned at the sample point. If SAMP = 1, the resulting bit value is determined by using majority rule on the three total samples. For higher bit rates, it is recommended that only one sample is taken per bit time (SAMP = 0).
6:4 TSEG2[2:0]	Time Segment 2 — Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point (see Figure 12-43). Time segment 2 (TSEG2) values are programmable as shown in Table 12-7.
3:0 TSEG1[3:0]	Time Segment 1 — Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point (see Figure 12-43). Time segment 1 (TSEG1) values are programmable as shown in Table 12-8.

¹ In this case, PHASE_SEG1 must be at least 2 time quanta (Tq).





Figure 12-10. MSCAN Transmitter Flag Register (CANTFLG)

NOTE

The CANTFLG register is held in the reset state when the initialization mode is active (INITRQ = 1 and INITAK = 1). This register is writable when not in initialization mode (INITRQ = 0 and INITAK = 0).

Read: Anytime

Write: Anytime for TXEx flags when not in initialization mode; write of 1 clears flag, write of 0 is ignored

Field	Description
2:0 TXE[2:0]	Transmitter Buffer Empty — This flag indicates that the associated transmit message buffer is empty, and thus not scheduled for transmission. The CPU must clear the flag after a message is set up in the transmit buffer and is due for transmission. The MSCAN sets the flag after the message is sent successfully. The flag is also set by the MSCAN when the transmission request is successfully aborted due to a pending abort request (see Section 12.3.8, "MSCAN Transmitter Message Abort Request Register (CANTARQ)"). If not masked, a transmit interrupt is pending while this flag is set. Clearing a TXEx flag also clears the corresponding ABTAKx (see Section 12.3.9, "MSCAN Transmitter Message Abort Acknowledge Register (CANTAAK)"). When a TXEx flag is set, the corresponding ABTRQx bit is cleared (see Section 12.3.8, "MSCAN Transmitter Message Abort Request Register (CANTARQ)"). When listen-mode is active (see Section 12.3.2, "MSCAN Control Register 1 (CANCTL1)") the TXEx flags cannot be cleared and no transmission is started. Read and write accesses to the transmit buffer are blocked, if the corresponding TXEx bit is cleared (TXEx = 0) and the buffer is scheduled for transmission. 0 The associated message buffer is full (loaded with a message due for transmission) 1 The associated message buffer is empty (not scheduled)

Table 12-11. CANTFLG Register Field Descriptions

12.3.7 MSCAN Transmitter Interrupt Enable Register (CANTIER)

This register contains the interrupt enable bits for the transmit buffer empty interrupt flags.





MC9S08DV60 Series Data Sheet, Rev 3



Chapter 12 Freescale's Controller Area Network (S08MSCANV1)

12.5.2.1 Message Transmit Background

Modern application layer software is built upon two fundamental assumptions:

- Any CAN node is able to send out a stream of scheduled messages without releasing the CAN bus between the two messages. Such nodes arbitrate for the CAN bus immediately after sending the previous message and only release the CAN bus in case of lost arbitration.
- The internal message queue within any CAN node is organized such that the highest priority message is sent out first, if more than one message is ready to be sent.

The behavior described in the bullets above cannot be achieved with a single transmit buffer. That buffer must be reloaded immediately after the previous message is sent. This loading process lasts a finite amount of time and must be completed within the inter-frame sequence (IFS) to be able to send an uninterrupted stream of messages. Even if this is feasible for limited CAN bus speeds, it requires that the CPU reacts with short latencies to the transmit interrupt.

A double buffer scheme de-couples the reloading of the transmit buffer from the actual message sending and, therefore, reduces the reactiveness requirements of the CPU. Problems can arise if the sending of a message is finished while the CPU re-loads the second buffer. No buffer would then be ready for transmission, and the CAN bus would be released.

At least three transmit buffers are required to meet the first of the above requirements under all circumstances. The MSCAN has three transmit buffers.

The second requirement calls for some sort of internal prioritization which the MSCAN implements with the "local priority" concept described in Section 12.5.2.2, "Transmit Structures."

12.5.2.2 Transmit Structures

The MSCAN triple transmit buffer scheme optimizes real-time performance by allowing multiple messages to be set up in advance. The three buffers are arranged as shown in Figure 12-38.

All three buffers have a 13-byte data structure similar to the outline of the receive buffers (see Section 12.4, "Programmer's Model of Message Storage"). An additional Section 12.4.5, "Transmit Buffer Priority Register (TBPR) contains an 8-bit local priority field (PRIO) (see Section 12.4.5, "Transmit Buffer Priority Register (TBPR)"). The remaining two bytes are used for time stamping of a message, if required (see Section 12.4.6, "Time Stamp Register (TSRH–TSRL)").

To transmit a message, the CPU must identify an available transmit buffer, which is indicated by a set transmitter buffer empty (TXEx) flag (see Section 12.3.6, "MSCAN Transmitter Flag Register (CANTFLG)"). If a transmit buffer is available, the CPU must set a pointer to this buffer by writing to the CANTBSEL register (see Section 12.3.10, "MSCAN Transmit Buffer Selection Register (CANTBSEL)"). This makes the respective buffer accessible within the CANTXFG address space (see Section 12.4, "Programmer's Model of Message Storage"). The algorithmic feature associated with the CANTBSEL register simplifies the transmit buffer selection. In addition, this scheme makes the handler software simpler because only one address area is applicable for the transmit process, and the required address space is minimized.

The CPU then stores the identifier, the control bits, and the data content into one of the transmit buffers. Finally, the buffer is flagged as ready for transmission by clearing the associated TXE flag.



```
Chapter 12 Freescale's Controller Area Network (S08MSCANV1)
```







Chapter 14 Serial Communications Interface (S08SCIV4)



15.3.2 RTC Counter Register (RTCCNT)

RTCCNT is the read-only value of the current RTC count of the 8-bit counter.



Figure 15-4. RTC Counter Register (RTCCNT)

Table 15-4. RTCCNT Field Descriptions

Field	Description
7:0 RTCCNT	RTC Count. These eight read-only bits contain the current value of the 8-bit counter. Writes have no effect to this register. Reset, writing to RTCMOD, or writing different values to RTCLKS and RTCPS clear the count to 0x00.

15.3.3 RTC Modulo Register (RTCMOD)



 Table 15-5. RTCMOD Field Descriptions

Field	Description
7:0 RTCMOD	RTC Modulo. These eight read/write bits contain the modulo value used to reset the count to 0x00 upon a compare match and set the RTIF status bit. A value of 0x00 sets the RTIF bit on each rising edge of the prescaler output. Writing to RTCMOD resets the prescaler and the RTCCNT counters to 0x00. Reset sets the modulo to 0x00.

15.4 Functional Description

The RTC is composed of a main 8-bit up-counter with an 8-bit modulo register, a clock source selector, and a prescaler block with binary-based and decimal-based selectable values. The module also contains software selectable interrupt logic.

After any MCU reset, the counter is stopped and reset to 0x00, the modulus register is set to 0x00, and the prescaler is off. The 1-kHz internal oscillator clock is selected as the default clock source. To start the prescaler, write any value other than zero to the prescaler select bits (RTCPS).

Three clock sources are software selectable: the low power oscillator clock (LPO), the external clock (ERCLK), and the internal clock (IRCLK). The RTC clock select bits (RTCLKS) select the desired clock source. If a different value is written to RTCLKS, the prescaler and RTCCNT counters are reset to 0x00.



Chapter 15 Real-Time Counter (S08RTCV1)