



Welcome to [E-XFL.COM](https://www.e-xfl.com)

### What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Obsolete
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	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	<a href="https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08dv16clc">https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s08dv16clc</a>

Section Number	Title	Page
12.3.15	MSCAN Identifier Acceptance Registers (CANIDAR0-7)	237
12.3.16	MSCAN Identifier Mask Registers (CANIDMR0–CANIDMR7)	238
12.4	Programmer's Model of Message Storage	239
12.4.1	Identifier Registers (IDR0–IDR3)	242
12.4.2	IDR0–IDR3 for Standard Identifier Mapping	244
12.4.3	Data Segment Registers (DSR0-7)	245
12.4.4	Data Length Register (DLR)	246
12.4.5	Transmit Buffer Priority Register (TBPR)	247
12.4.6	Time Stamp Register (TSRH–TSRL)	247
12.5	Functional Description	248
12.5.1	General	248
12.5.2	Message Storage	249
12.5.3	Identifier Acceptance Filter	252
12.5.4	Modes of Operation	259
12.5.5	Low-Power Options	260
12.5.6	Reset Initialization	266
12.5.7	Interrupts	266
12.6	Initialization/Application Information	268
12.6.1	MSCAN initialization	268
12.6.2	Bus-Off Recovery	269

## Chapter 13

### Serial Peripheral Interface (S08SPIV3)

13.1	Introduction	271
13.1.1	Features	273
13.1.2	Block Diagrams	273
13.1.3	SPI Baud Rate Generation	275
13.2	External Signal Description	276
13.2.1	SPSCK — SPI Serial Clock	276
13.2.2	MOSI — Master Data Out, Slave Data In	276
13.2.3	MISO — Master Data In, Slave Data Out	276
13.2.4	$\overline{SS}$ — Slave Select	276
13.3	Modes of Operation	277
13.3.1	SPI in Stop Modes	277
13.4	Register Definition	277
13.4.1	SPI Control Register 1 (SPIC1)	277
13.4.2	SPI Control Register 2 (SPIC2)	278
13.4.3	SPI Baud Rate Register (SPIBR)	279
13.4.4	SPI Status Register (SPIS)	280
13.4.5	SPI Data Register (SPID)	281
13.5	Functional Description	282
13.5.1	SPI Clock Formats	282

Whenever any reset is initiated (whether from an external signal or from an internal system), the  $\overline{\text{RESET}}$  pin is driven low for about 34 bus cycles. The reset circuitry decodes the cause of reset and records it by setting a corresponding bit in the system reset status register (SRS).

### 2.2.4 Background / Mode Select (BKGD/MS)

While in reset, the BKGD/MS pin functions as a mode select pin. Immediately after reset rises, the pin functions as the background pin and can be used for background debug communication. While functioning as a background or mode select pin, the pin includes an internal pull-up device, input hysteresis, a standard output driver, and no output slew rate control.

If nothing is connected to this pin, the MCU will enter normal operating mode at the rising edge of reset. If a debug system is connected to the 6-pin standard background debug header, it can hold BKGD low during the rising edge of reset which forces the MCU to active background mode.

The BKGD/MS pin is used primarily for background debug controller (BDC) communications using a custom protocol that uses 16 clock cycles of the target MCU's BDC clock per bit time. The target MCU's BDC clock could be as fast as the bus clock rate, so there should never be any significant capacitance connected to the BKGD/MS pin that could interfere with background serial communications.

Although the BKGD/MS pin is a pseudo open-drain pin, the background debug communication protocol provides brief, actively driven, high speedup pulses to ensure fast rise times. Small capacitances from cables and the absolute value of the internal pull-up device play almost no role in determining rise and fall times on the BKGD/MS pin.

### 2.2.5 ADC Reference Pins ( $V_{\text{REFH}}$ , $V_{\text{REFL}}$ )

The  $V_{\text{REFH}}$  and  $V_{\text{REFL}}$  pins are the voltage reference high and voltage reference low inputs, respectively, for the ADC module.

### 2.2.6 General-Purpose I/O and Peripheral Ports

The MC9S08DV60 Series series of MCUs support up to 53 general-purpose I/O pins and 1 input-only pin, which are shared with on-chip peripheral functions (timers, serial I/O, ADC, MSCAN, etc.).

When a port pin is configured as a general-purpose output or a peripheral uses the port pin as an output, software can select one of two drive strengths and enable or disable slew rate control. When a port pin is configured as a general-purpose input or a peripheral uses the port pin as an input, software can enable a pull-up device. Immediately after reset, all of these pins are configured as high-impedance general-purpose inputs with internal pull-up devices disabled.

When an on-chip peripheral system is controlling a pin, data direction control bits still determine what is read from port data registers even though the peripheral module controls the pin direction by controlling the enable for the pin's output buffer. For information about controlling these pins as general-purpose I/O pins, see [Chapter 6, "Parallel Input/Output Control."](#)

**NOTE**

To avoid extra current drain from floating input pins, the reset initialization routine in the application program should either enable on-chip pull-up devices or change the direction of unused or non-bonded pins to outputs so they do not float.

## Chapter 6

# Parallel Input/Output Control

This section explains software controls related to parallel input/output (I/O) and pin control. The MC9S08DV60 Series has seven parallel I/O ports which include a total of up to 53 I/O pins and one input-only pin. See [Chapter 2, “Pins and Connections,”](#) for more information about pin assignments and external hardware considerations of these pins.

Many of these pins are shared with on-chip peripherals such as timer systems, communication systems, or pin interrupts as shown in [Table 2-1](#). The peripheral modules have priority over the general-purpose I/O functions so that when a peripheral is enabled, the I/O functions associated with the shared pins are disabled.

After reset, the shared peripheral functions are disabled and the pins are configured as inputs ( $PTxDDn = 0$ ). The pin control functions for each pin are configured as follows: slew rate control enabled ( $PTxSEn = 1$ ), low drive strength selected ( $PTxDSn = 0$ ), and internal pull-ups disabled ( $PTxPEn = 0$ ).

### NOTE

- Not all general-purpose I/O pins are available on all packages. To avoid extra current drain from floating input pins, the user's reset initialization routine in the application program must either enable on-chip pull-up devices or change the direction of unconnected pins to outputs so the pins do not float.
- The PTE1 pin does not contain a clamp diode to  $V_{DD}$  and should not be driven above  $V_{DD}$ . The voltage measured on the internally pulled up PTE1 pin may be as low as  $V_{DD} - 0.7$  V. The internal gates connected to this pin are pulled all the way to  $V_{DD}$ .

## 6.1 Port Data and Data Direction

Reading and writing of parallel I/Os are performed through the port data registers. The direction, either input or output, is controlled through the port data direction registers. The parallel I/O port function for an individual pin is illustrated in the block diagram shown in [Figure 6-1](#).

The data direction control bit ( $PTxDDn$ ) determines whether the output buffer for the associated pin is enabled, and also controls the source for port data register reads. The input buffer for the associated pin is always enabled unless the pin is enabled as an analog function or is an output-only pin.

When a shared digital function is enabled for a pin, the output buffer is controlled by the shared function. However, the data direction register bit will continue to control the source for reads of the port data register.

When a shared analog function is enabled for a pin, both the input and output buffers are disabled. A value of 0 is read for any port data bit where the bit is an input ( $PTxDDn = 0$ ) and the input buffer is disabled.

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

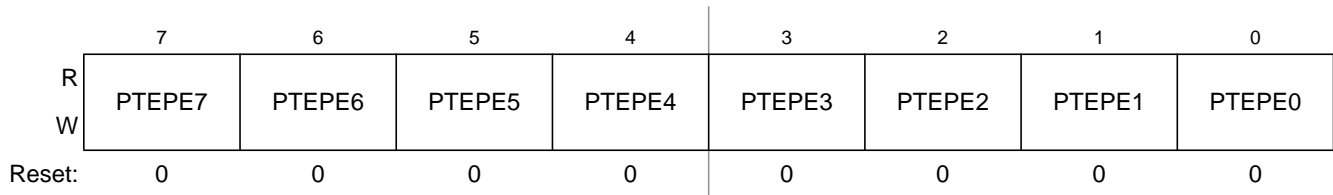


Figure 6-34. Internal Pull Enable for Port E Register (PTEPE)

Table 6-32. PTEPE Register Field Descriptions

Field	Description
7:0 PTEPE[7:0]	<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. 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)

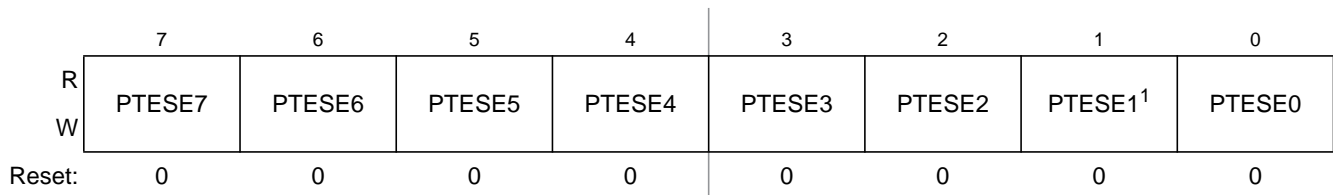


Figure 6-35. Slew Rate Enable for Port E Register (PTESE)

<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]	<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. 0 Output slew rate control disabled for port E bit n. 1 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.

### 7.2.3 Stack Pointer (SP)

This 16-bit address pointer register points at the next available location on the automatic last-in-first-out (LIFO) stack. The stack may be located anywhere in the 64-Kbyte address space that has RAM and can be any size up to the amount of available RAM. The stack is used to automatically save the return address for subroutine calls, the return address and CPU registers during interrupts, and for local variables. The AIS (add immediate to stack pointer) instruction adds an 8-bit signed immediate value to SP. This is most often used to allocate or deallocate space for local variables on the stack.

SP is forced to 0x00FF at reset for compatibility with the earlier M68HC05 Family. HCS08 programs normally change the value in SP to the address of the last location (highest address) in on-chip RAM during reset initialization to free up direct page RAM (from the end of the on-chip registers to 0x00FF).

The RSP (reset stack pointer) instruction was included for compatibility with the M68HC05 Family and is seldom used in new HCS08 programs because it only affects the low-order half of the stack pointer.

### 7.2.4 Program Counter (PC)

The program counter is a 16-bit register that contains the address of the next instruction or operand to be fetched.

During normal program execution, the program counter automatically increments to the next sequential memory location every time an instruction or operand is fetched. Jump, branch, interrupt, and return operations load the program counter with an address other than that of the next sequential location. This is called a change-of-flow.

During reset, the program counter is loaded with the reset vector that is located at 0xFFFFE and 0xFFFF. The vector stored there is the address of the first instruction that will be executed after exiting the reset state.

### 7.2.5 Condition Code Register (CCR)

The 8-bit condition code register contains the interrupt mask (I) and five flags that indicate the results of the instruction just executed. Bits 6 and 5 are set permanently to 1. The following paragraphs describe the functions of the condition code bits in general terms. For a more detailed explanation of how each instruction sets the CCR bits, refer to the *HCS08 Family Reference Manual, volume 1*, Freescale Semiconductor document order number HCS08RMv1.

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

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR	
						V 1 1 H	I N Z C
BCC <i>rel</i>	Branch if Carry Bit Clear (if C = 0)	REL	24 rr	3	ppp	- 1 1 -	- - - -
BCLR <i>n,opr8a</i>	Clear Bit n in Memory (Mn ← 0)	DIR (b0)	11 dd	5	r fwpp	- 1 1 -	- - - -
		DIR (b1)	13 dd	5	r fwpp		
		DIR (b2)	15 dd	5	r fwpp		
		DIR (b3)	17 dd	5	r fwpp		
		DIR (b4)	19 dd	5	r fwpp		
		DIR (b5)	1B dd	5	r fwpp		
		DIR (b6)	1D dd	5	r fwpp		
		DIR (b7)	1F dd	5	r fwpp		
BCS <i>rel</i>	Branch if Carry Bit Set (if C = 1) (Same as BLO)	REL	25 rr	3	ppp	- 1 1 -	- - - -
BEQ <i>rel</i>	Branch if Equal (if Z = 1)	REL	27 rr	3	ppp	- 1 1 -	- - - -
BGE <i>rel</i>	Branch if Greater Than or Equal To (if N ⊕ V = 0) (Signed)	REL	90 rr	3	ppp	- 1 1 -	- - - -
BGND	Enter active background if ENBDM=1 Waits for and processes BDM commands until GO, TRACE1, or TAGGO	INH	82	5+	f p . . . ppp	- 1 1 -	- - - -
BGT <i>rel</i>	Branch if Greater Than (if Z   (N ⊕ V) = 0) (Signed)	REL	92 rr	3	ppp	- 1 1 -	- - - -
BHCC <i>rel</i>	Branch if Half Carry Bit Clear (if H = 0)	REL	28 rr	3	ppp	- 1 1 -	- - - -
BHCS <i>rel</i>	Branch if Half Carry Bit Set (if H = 1)	REL	29 rr	3	ppp	- 1 1 -	- - - -
BHI <i>rel</i>	Branch if Higher (if C   Z = 0)	REL	22 rr	3	ppp	- 1 1 -	- - - -
BHS <i>rel</i>	Branch if Higher or Same (if C = 0) (Same as BCC)	REL	24 rr	3	ppp	- 1 1 -	- - - -
BIH <i>rel</i>	Branch if IRQ Pin High (if IRQ pin = 1)	REL	2F rr	3	ppp	- 1 1 -	- - - -
BIL <i>rel</i>	Branch if IRQ Pin Low (if IRQ pin = 0)	REL	2E rr	3	ppp	- 1 1 -	- - - -
BIT # <i>opr8i</i> BIT <i>opr8a</i> BIT <i>opr16a</i> BIT <i>opr16,X</i> BIT <i>opr8,X</i> BIT <i>,X</i> BIT <i>opr16,SP</i> BIT <i>opr8,SP</i>	Bit Test (A) & (M) (CCR Updated but Operands Not Changed)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A5 ii B5 dd C5 hh ll D5 ee ff E5 ff F5 9E D5 ee ff 9E E5 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	0 1 1 -	- ↑ ↓ -
BLE <i>rel</i>	Branch if Less Than or Equal To (if Z   (N ⊕ V) = 1) (Signed)	REL	93 rr	3	ppp	- 1 1 -	- - - -
BLO <i>rel</i>	Branch if Lower (if C = 1) (Same as BCS)	REL	25 rr	3	ppp	- 1 1 -	- - - -
BLS <i>rel</i>	Branch if Lower or Same (if C   Z = 1)	REL	23 rr	3	ppp	- 1 1 -	- - - -
BLT <i>rel</i>	Branch if Less Than (if N ⊕ V = 1) (Signed)	REL	91 rr	3	ppp	- 1 1 -	- - - -
BMC <i>rel</i>	Branch if Interrupt Mask Clear (if I = 0)	REL	2C rr	3	ppp	- 1 1 -	- - - -
BMI <i>rel</i>	Branch if Minus (if N = 1)	REL	2B rr	3	ppp	- 1 1 -	- - - -
BMS <i>rel</i>	Branch if Interrupt Mask Set (if I = 1)	REL	2D rr	3	ppp	- 1 1 -	- - - -
BNE <i>rel</i>	Branch if Not Equal (if Z = 0)	REL	26 rr	3	ppp	- 1 1 -	- - - -



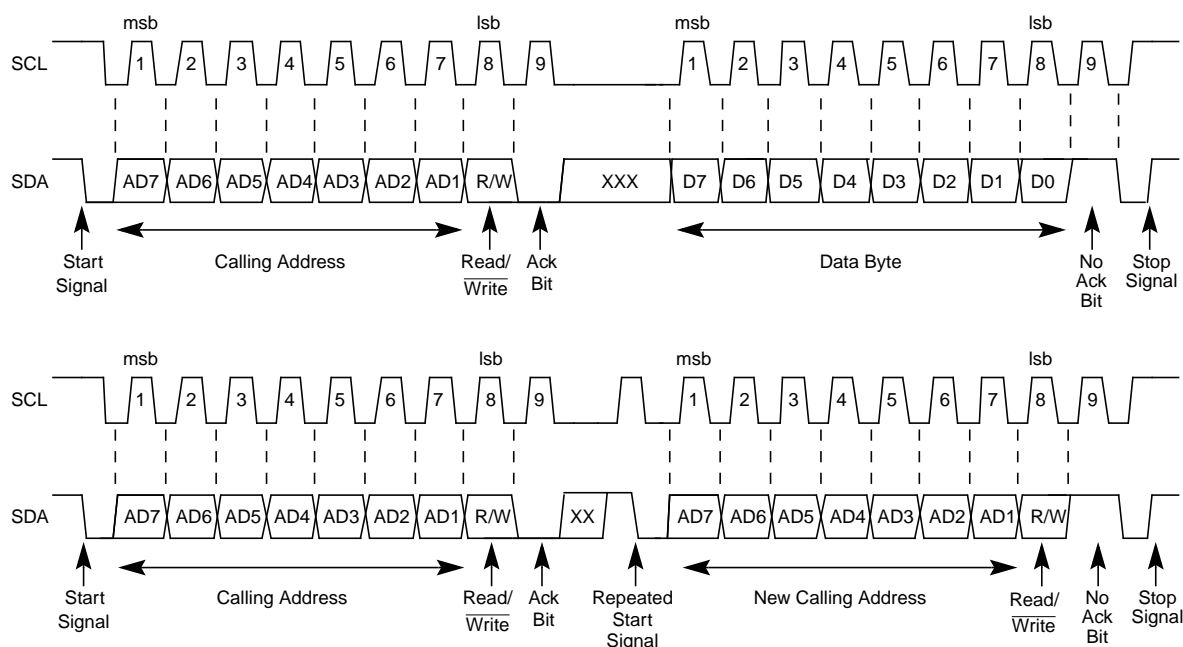


Figure 11-9. IIC Bus Transmission Signals

#### 11.4.1.1 Start Signal

When the bus is free, no master device is engaging the bus (SCL and SDA lines are at logical high), a master may initiate communication by sending a start signal. As shown in Figure 11-9, a start signal is defined as a high-to-low transition of SDA while SCL is high. This signal denotes the beginning of a new data transfer (each data transfer may contain several bytes of data) and brings all slaves out of their idle states.

#### 11.4.1.2 Slave Address Transmission

The first byte of data transferred immediately after the start signal is the slave address transmitted by the master. This is a seven-bit calling address followed by a  $R/\overline{W}$  bit. The  $R/\overline{W}$  bit tells the slave the desired direction of data transfer.

- 1 = Read transfer, the slave transmits data to the master.
- 0 = Write transfer, the master transmits data to the slave.

Only the slave with a calling address that matches the one transmitted by the master responds by sending back an acknowledge bit. This is done by pulling the SDA low at the ninth clock (see Figure 11-9).

No two slaves in the system may have the same address. If the IIC module is the master, it must not transmit an address equal to its own slave address. The IIC cannot be master and slave at the same time. However, if arbitration is lost during an address cycle, the IIC reverts to slave mode and operates correctly even if it is being addressed by another master.

**Table 12-1. CANCTL0 Register Field Descriptions**

Field	Description
7 RXFRM <sup>1</sup>	<b>Received Frame Flag</b> — This bit is read and clear only. It is set when a receiver has received a valid message correctly, independently of the filter configuration. After it is set, it remains set until cleared by software or reset. Clearing is done by writing a 1. Writing a 0 is ignored. This bit is not valid in loopback mode. 0 No valid message was received since last clearing this flag 1 A valid message was received since last clearing of this flag
6 RXACT	<b>Receiver Active Status</b> — This read-only flag indicates the MSCAN is receiving a message. The flag is controlled by the receiver front end. This bit is not valid in loopback mode. 0 MSCAN is transmitting or idle <sup>2</sup> 1 MSCAN is receiving a message (including when arbitration is lost) <sup>2</sup>
5 CSWAI <sup>3</sup>	<b>CAN Stops in Wait Mode</b> — Enabling this bit allows for lower power consumption in wait mode by disabling all the clocks at the CPU bus interface to the MSCAN module. 0 The module is not affected during wait mode 1 The module ceases to be clocked during wait mode
4 SYNCH	<b>Synchronized Status</b> — This read-only flag indicates whether the MSCAN is synchronized to the CAN bus and able to participate in the communication process. It is set and cleared by the MSCAN. 0 MSCAN is not synchronized to the CAN bus 1 MSCAN is synchronized to the CAN bus
3 TIME	<b>Timer Enable</b> — This bit activates an internal 16-bit wide free running timer which is clocked by the bit clock rate. If the timer is enabled, a 16-bit time stamp will be assigned to each transmitted/received message within the active TX/RX buffer. As soon as a message is acknowledged on the CAN bus, the time stamp will be written to the highest bytes (0x000E, 0x000F) in the appropriate buffer (see <a href="#">Section 12.4, “Programmer's Model of Message Storage”</a> ). The internal timer is reset (all bits set to 0) when disabled. This bit is held low in initialization mode. 0 Disable internal MSCAN timer 1 Enable internal MSCAN timer
2 WUPE <sup>4</sup>	<b>Wake-Up Enable</b> — This configuration bit allows the MSCAN to restart from sleep mode when traffic on CAN is detected (see <a href="#">Section 12.5.5.4, “MSCAN Sleep Mode”</a> ). This bit must be configured before sleep mode entry for the selected function to take effect. 0 Wake-up disabled — The MSCAN ignores traffic on CAN 1 Wake-up enabled — The MSCAN is able to restart

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

Field	Description
1 SLPAK	<b>Sleep Mode Acknowledge</b> — This flag indicates whether the MSCAN module has entered sleep mode (see <a href="#">Section 12.5.5.4, “MSCAN Sleep Mode”</a> ). 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. 0 Running — The MSCAN operates normally 1 Sleep mode active — The MSCAN has entered sleep mode
0 INITAK	<b>Initialization Mode Acknowledge</b> — This flag indicates whether the MSCAN module is in initialization mode (see <a href="#">Section 12.5.5.5, “MSCAN Initialization Mode”</a> ). 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

### 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 <a href="#">Table 12-4</a> ).
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 <a href="#">Table 12-5</a> ).

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

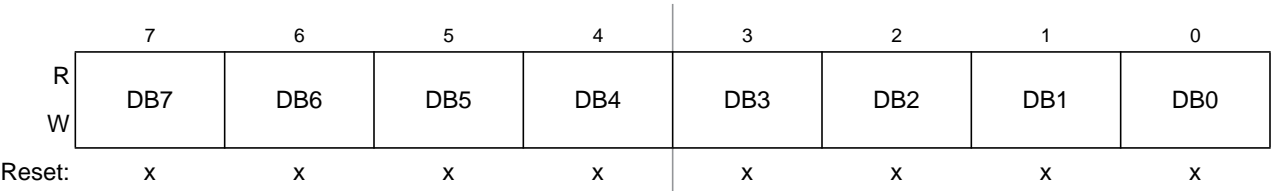


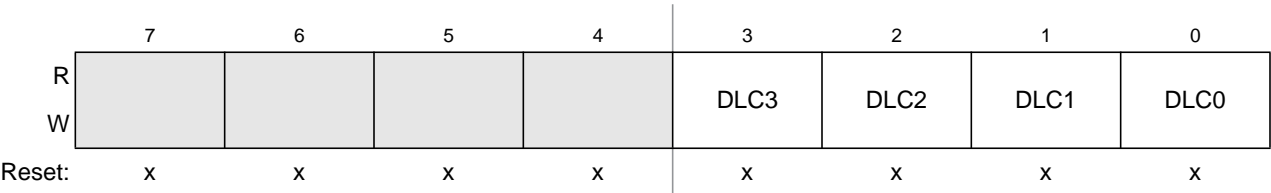
Figure 12-33. Data Segment Registers (DSR0–DSR7) — Extended Identifier Mapping

Table 12-31. DSR0–DSR7 Register Field Descriptions

Field	Description
7:0 DB[7:0]	Data bits 7:0

### 12.4.4 Data Length Register (DLR)

This register keeps the data length field of the CAN frame.




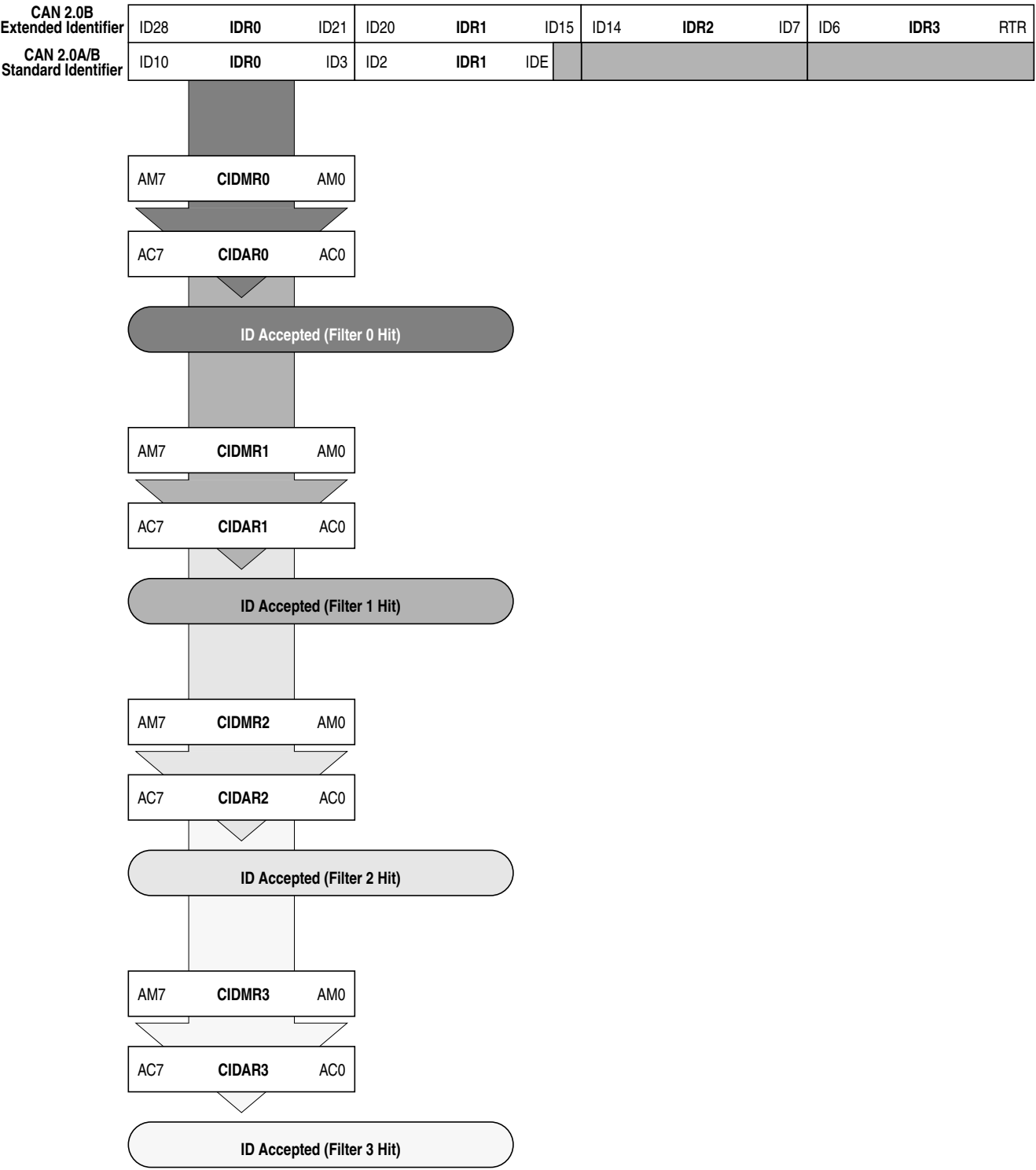
 = Unused; always read “x”

Figure 12-34. Data Length Register (DLR) — Extended Identifier Mapping

Table 12-32. DLR Register Field Descriptions

Field	Description
3:0 DLC[3:0]	<b>Data Length Code Bits</b> — The data length code contains the number of bytes (data byte count) of the respective message. During the transmission of a remote frame, the data length code is transmitted as programmed while the number of transmitted data bytes is always 0. The data byte count ranges from 0 to 8 for a data frame. <a href="#">Table 12-33</a> shows the effect of setting the DLC bits.



**Figure 12-41. 8-bit Maskable Identifier Acceptance Filters**

MSCAN filter uses three sets of registers to provide the filter configuration. Firstly, the CANIDAC register determines the configuration of the banks into filter sizes and number of filters. Secondly, registers CANIDMR0/1/2/3 determine those bits on which the filter will operate by placing a '0' at the appropriate

**Table 12-34. Time Segment Syntax**

Syntax	Description
SYNC_SEG	System expects transitions to occur on the CAN bus during this period.
Transmit Point	A node in transmit mode transfers a new value to the CAN bus at this point.
Sample Point	A node in receive mode samples the CAN bus at this point. If the three samples per bit option is selected, then this point marks the position of the third sample.

The synchronization jump width (see the Bosch CAN specification for details) can be programmed in a range of 1 to 4 time quanta by setting the SJW parameter.

The SYNC\_SEG, TSEG1, TSEG2, and SJW parameters are set by programming the MSCAN bus timing registers (CANBTR0, CANBTR1) (see [Section 12.3.3, “MSCAN Bus Timing Register 0 \(CANBTR0\)”](#) and [Section 12.3.4, “MSCAN Bus Timing Register 1 \(CANBTR1\)”](#)).

[Table 12-35](#) gives an overview of the CAN compliant segment settings and the related parameter values.

#### NOTE

It is the user's responsibility to ensure the bit time settings are in compliance with the CAN standard.

**Table 12-35. CAN Standard Compliant Bit Time Segment Settings**

Time Segment 1	TSEG1	Time Segment 2	TSEG2	Synchronization Jump Width	SJW
5 .. 10	4 .. 9	2	1	1 .. 2	0 .. 1
4 .. 11	3 .. 10	3	2	1 .. 3	0 .. 2
5 .. 12	4 .. 11	4	3	1 .. 4	0 .. 3
6 .. 13	5 .. 12	5	4	1 .. 4	0 .. 3
7 .. 14	6 .. 13	6	5	1 .. 4	0 .. 3
8 .. 15	7 .. 14	7	6	1 .. 4	0 .. 3
9 .. 16	8 .. 15	8	7	1 .. 4	0 .. 3

## 12.5.4 Modes of Operation

### 12.5.4.1 Normal Modes

The MSCAN module behaves as described within this specification in all normal system operation modes.

### 12.5.4.2 Special Modes

The MSCAN module behaves as described within this specification in all special system operation modes.

### 13.1.1 Features

Features of the SPI module include:

- Master or slave mode operation
- Full-duplex or single-wire bidirectional option
- Programmable transmit bit rate
- Double-buffered transmit and receive
- Serial clock phase and polarity options
- Slave select output
- Selectable MSB-first or LSB-first shifting

### 13.1.2 Block Diagrams

This section includes block diagrams showing SPI system connections, the internal organization of the SPI module, and the SPI clock dividers that control the master mode bit rate.

#### 13.1.2.1 SPI System Block Diagram

Figure 13-2 shows the SPI modules of two MCUs connected in a master-slave arrangement. The master device initiates all SPI data transfers. During a transfer, the master shifts data out (on the MOSI pin) to the slave while simultaneously shifting data in (on the MISO pin) from the slave. The transfer effectively exchanges the data that was in the SPI shift registers of the two SPI systems. The SPSCCK signal is a clock output from the master and an input to the slave. The slave device must be selected by a low level on the slave select input ( $\overline{SS}$  pin). In this system, the master device has configured its  $\overline{SS}$  pin as an optional slave select output.

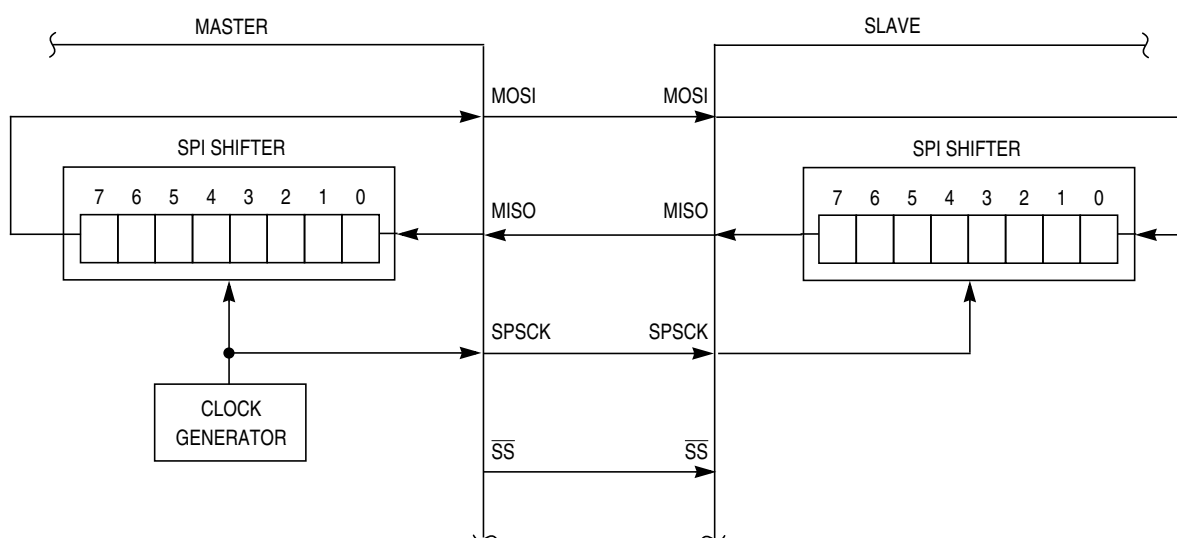


Figure 13-2. SPI System Connections

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 one-half SPSCK cycle before the start of the transfer and goes back high at the end of the eighth bit time of the transfer. The  $\overline{SS}$  IN waveform applies to the slave select input of a slave.

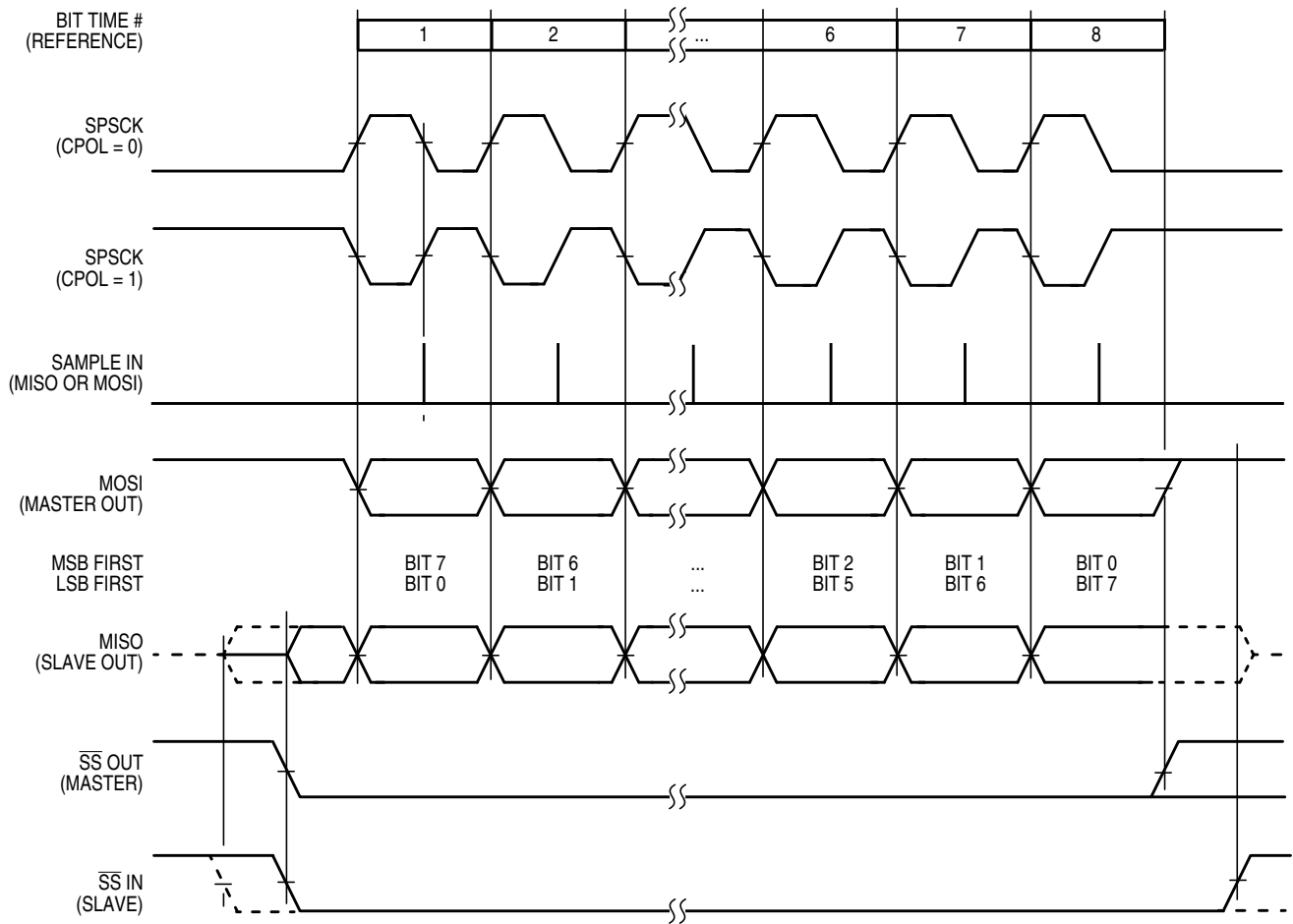


Figure 13-10. SPI Clock Formats (CPHA = 1)

When CPHA = 1, the slave begins to drive its MISO output when  $\overline{SS}$  goes to active low, but the data is not defined until the first SPSCK edge. The first SPSCK edge shifts the first bit of data from the shifter onto the MOSI output of the master and the MISO output of the slave. The next SPSCK edge causes both the master and the slave to sample the data bit values on their MISO and MOSI inputs, respectively. At the third SPSCK 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 = 1, the slave's  $\overline{SS}$  input is not required to go to its inactive high level between transfers.

Figure 13-11 shows the clock formats when CPHA = 0. At the top of the figure, the eight bit times are shown for reference with bit 1 starting as the slave is selected ( $\overline{SS}$  IN goes low), and bit 8 ends at the last SPSCK edge. The MSB first and LSB first lines show the order of SPI data bits depending on the setting



## Chapter 17

# Development Support

### 17.1 Introduction

Development support systems in the HCS08 include the background debug controller (BDC) and the on-chip debug module (DBG). The BDC provides a single-wire debug interface to the target MCU that provides a convenient interface for programming the on-chip Flash and other nonvolatile memories. The BDC is also the primary debug interface for development and allows non-intrusive access to memory data and traditional debug features such as CPU register modify, breakpoints, and single instruction trace commands.

In the HCS08 Family, address and data bus signals are not available on external pins (not even in test modes). Debug is done through commands fed into the target MCU via the single-wire background debug interface. The debug module provides a means to selectively trigger and capture bus information so an external development system can reconstruct what happened inside the MCU on a cycle-by-cycle basis without having external access to the address and data signals.

#### 17.1.1 Forcing Active Background

The method for forcing active background mode depends on the specific HCS08 derivative. For the MC9S08DV60, you can force active background after a power-on reset by holding the BKGD pin low as the device exits the reset condition. You can also force active background by driving BKGD low immediately after a serial background command that writes a one to the BDFR bit in the SBDIFR register. If no debug pod is connected to the BKGD pin, the MCU will always reset into normal operating mode.

When no debugger pod is connected to the 6-pin BDM interface connector, the internal pullup on BKGD chooses normal operating mode. When a debug pod is connected to BKGD it is possible to force the MCU into active background mode after reset. The specific conditions for forcing active background depend upon the HCS08 derivative (refer to the introduction to this Development Support section). It is not necessary to reset the target MCU to communicate with it through the background debug interface.

## 17.2.2 Communication Details

The BDC serial interface requires the external controller to generate a falling edge on the BKGD pin to indicate the start of each bit time. The external controller provides this falling edge whether data is transmitted or received.

BKGD is a pseudo-open-drain pin that can be driven either by an external controller or by the MCU. Data is transferred MSB first at 16 BDC clock cycles per bit (nominal speed). The interface times out if 512 BDC clock cycles occur between falling edges from the host. Any BDC command that was in progress when this timeout occurs is aborted without affecting the memory or operating mode of the target MCU system.

The custom serial protocol requires the debug pod to know the target BDC communication clock speed.

The clock switch (CLKSW) control bit in the BDC status and control register allows the user to select the BDC clock source. The BDC clock source can either be the bus or the alternate BDC clock source.

The BKGD pin can receive a high or low level or transmit a high or low level. The following diagrams show timing for each of these cases. Interface timing is synchronous to clocks in the target BDC, but asynchronous to the external host. The internal BDC clock signal is shown for reference in counting cycles.

## 17.3 On-Chip Debug System (DBG)

Because HCS08 devices do not have external address and data buses, the most important functions of an in-circuit emulator have been built onto the chip with the MCU. The debug system consists of an 8-stage FIFO that can store address or data bus information, and a flexible trigger system to decide when to capture bus information and what information to capture. The system relies on the single-wire background debug system to access debug control registers and to read results out of the eight stage FIFO.

The debug module includes control and status registers that are accessible in the user's memory map. These registers are located in the high register space to avoid using valuable direct page memory space.

Most of the debug module's functions are used during development, and user programs rarely access any of the control and status registers for the debug module. The one exception is that the debug system can provide the means to implement a form of ROM patching. This topic is discussed in greater detail in [Section 17.3.6, "Hardware Breakpoints."](#)

### 17.3.1 Comparators A and B

Two 16-bit comparators (A and B) can optionally be qualified with the R/W signal and an opcode tracking circuit. Separate control bits allow you to ignore R/W for each comparator. The opcode tracking circuitry optionally allows you to specify that a trigger will occur only if the opcode at the specified address is actually executed as opposed to only being read from memory into the instruction queue. The comparators are also capable of magnitude comparisons to support the inside range and outside range trigger modes. Comparators are disabled temporarily during all BDC accesses.

The A comparator is always associated with the 16-bit CPU address. The B comparator compares to the CPU address or the 8-bit CPU data bus, depending on the trigger mode selected. Because the CPU data bus is separated into a read data bus and a write data bus, the RWAEN and RWA control bits have an additional purpose, in full address plus data comparisons they are used to decide which of these buses to use in the comparator B data bus comparisons. If RWAEN = 1 (enabled) and RWA = 0 (write), the CPU's write data bus is used. Otherwise, the CPU's read data bus is used.

The currently selected trigger mode determines what the debugger logic does when a comparator detects a qualified match condition. A match can cause:

- Generation of a breakpoint to the CPU
- Storage of data bus values into the FIFO
- Starting to store change-of-flow addresses into the FIFO (begin type trace)
- Stopping the storage of change-of-flow addresses into the FIFO (end type trace)

### 17.3.2 Bus Capture Information and FIFO Operation

The usual way to use the FIFO is to setup the trigger mode and other control options, then arm the debugger. When the FIFO has filled or the debugger has stopped storing data into the FIFO, you would read the information out of it in the order it was stored into the FIFO. Status bits indicate the number of words of valid information that are in the FIFO as data is stored into it. If a trace run is manually halted by writing 0 to ARM before the FIFO is full (CNT = 1:0:0:0), the information is shifted by one position and

**Table A-12. MCG Frequency Specifications (Temperature Range = –40 to 125°C Ambient) (continued)**

Num	C	Rating	Symbol	Min	Typical	Max	Unit
18	T	RMS frequency variation of a single clock cycle measured 625 ns after reference edge. <sup>6</sup>	$f_{pll\_cycjit\_625ns}$	—	0.566 <sup>4</sup>	—	% $f_{pll}$
19	T	Maximum frequency variation averaged over 625 ns window.	$f_{pll\_maxjit\_625ns}$	—	0.113	—	% $f_{pll}$
20	D	Lock entry frequency tolerance <sup>7</sup>	$D_{lock}$	$\pm 1.49$	—	$\pm 2.98$	%
21	D	Lock exit frequency tolerance <sup>8</sup>	$D_{unl}$	$\pm 4.47$	—	$\pm 5.97$	%
22	D	Lock time - FLL	$t_{fll\_lock}$	—	—	$t_{fll\_acquire+}$ $1075(1/f_{int\_t})$	s
23	D	Lock time - PLL	$t_{pll\_lock}$	—	—	$t_{pll\_acquire+}$ $1075(1/f_{pll\_ref})$	s
24	D	Loss of external clock minimum frequency - RANGE = 0	$f_{loc\_low}$	$(3/5) \times f_{int}$	—	—	kHz
25	D	Loss of external clock minimum frequency - RANGE = 1	$f_{loc\_high}$	$(16/5) \times f_{int}$	—	—	kHz

<sup>1</sup> TRIM register at default value (0x80) and FTRIM control bit at default value (0x0).

<sup>2</sup> This specification applies to any time the FLL reference source or reference divider is changed, trim value changed or changing from FLL disabled (BLPE, BLPI) to FLL enabled (FEI, FEE, FBE, FBI). If a crystal/resonator is being used as the reference, this specification assumes it is already running.

<sup>3</sup> This specification applies to any time the PLL VCO divider or reference divider is changed, or changing from PLL disabled (BLPE, BLPI) to PLL enabled (PBE, PEE). If a crystal/resonator is being used as the reference, this specification assumes it is already running.

<sup>4</sup> Jitter is the average deviation from the programmed frequency measured over the specified interval at maximum  $f_{BUS}$ . Measurements are made with the device powered by filtered supplies and clocked by a stable external clock signal. Noise injected into the FLL circuitry via  $V_{DD}$  and  $V_{SS}$  and variation in crystal oscillator frequency increase the  $C_{Jitter}$  percentage for a given interval. Jitter measurements are based upon a 40MHz MCGOUT clock frequency.

<sup>5</sup> In some specifications, this value is described as “long term accuracy of PLL output clock (averaged over 2 ms)” with symbol “ $f_{pll\_jitter\_2ms}$ .” The parameter is unchanged, but the description has been changed for clarification purposes.

<sup>6</sup> In some specifications, this value is described as “Jitter of PLL output clock measured over 625 ns” with symbol “ $f_{pll\_jitter\_625ns}$ .” The parameter is unchanged, but the description has been changed for clarification purposes.

<sup>7</sup> Below  $D_{lock}$  minimum, the MCG is guaranteed to enter lock. Above  $D_{lock}$  maximum, the MCG will not enter lock. But if the MCG is already in lock, then the MCG may stay in lock.

<sup>8</sup> Below  $D_{unl}$  minimum, the MCG will not exit lock if already in lock. Above  $D_{unl}$  maximum, the MCG is guaranteed to exit lock.

## A.14.1 Radiated Emissions

Microcontroller radiated RF emissions are measured from 150 kHz to 1 GHz using the TEM/GTEM Cell method in accordance with the IEC 61967-2 and SAE J1752/3 standards. The measurement is performed with the microcontroller installed on a custom EMC evaluation board while running specialized EMC test software. The radiated emissions from the microcontroller are measured in a TEM cell in two package orientations (North and East). For more detailed information concerning the evaluation results, conditions and setup, please refer to the EMC Evaluation Report for this device.

The maximum radiated RF emissions of the tested configuration in all orientations are less than or equal to the reported emissions levels.

**Table A-18. Radiated Emissions for 3M05C Mask Set**

Parameter	Symbol	Conditions	Frequency	$f_{osc}/f_{CPU}$	Level <sup>1</sup> (Max)	Unit
Radiated emissions, electric field — Conditions -	$V_{RE\_TEM}$	$V_{DD} = 5$ $T_A = +25^{\circ}C$ 64 LQFP	0.15 – 50 MHz	16 MHz Crystal 20 MHz Bus	18	dB $\mu$ V
			50 – 150 MHz		18	
			150 – 500 MHz		13	
			500 – 1000 MHz		7	
			IEC Level		L	—
			SAE Level		2	—

<sup>1</sup> Data based on qualification test results.