NXP USA Inc. - MC9S08DV16CLCR Datasheet





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/mc9s08dv16clcr

Email: info@E-XFL.COM

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



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	 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. 0 Command buffer is full (not ready for additional commands). 1 A new burst program command can be written to the command buffer.
6 FCCF	 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. Command in progress All commands complete
5 FPVIOL	 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. 0 No protection violation. 1 An attempt was made to erase or program a protected location.



Chapter 5 Resets, Interrupts, and General System Control

5.8.2 System Reset Status Register (SRS)

This high page register includes read-only status flags to indicate the source of the most recent reset. When a debug host forces reset by writing 1 to BDFR in the SBDFR register, none of the status bits in SRS will be set. Writing any value to this register address causes a COP reset when the COP is enabled except the values 0x55 and 0xAA. Writing a 0x55-0xAA sequence to this address clears the COP watchdog timer without affecting the contents of this register. The reset state of these bits depends on what caused the MCU to reset.

	7	6	5	4	3	2	1	0
R	POR	PIN	COP	ILOP	ILAD	LOC	LVD	0
W		Wr	iting 0x55, 0xA	A to SRS addr	ess clears CO	P watchdog tim	ier.	
POR:	1	0	0	0	0	0	1	0
LVD:	u	0	0	0	0	0	1	0
Any other reset:	0	Note ⁽¹⁾	Note ⁽¹⁾	Note ⁽¹⁾	Note ⁽¹⁾	0	0	0

¹ Any of these reset sources that are active at the time of reset entry will cause the corresponding bit(s) to be set; bits corresponding to sources that are not active at the time of reset entry will be cleared.

Figure 5-3. System Reset Status (SRS)

Table 5-3. SRS Register Field Descriptions

Field	Description
7 POR	 Power-On Reset — Reset was caused by the power-on detection logic. Because the internal supply voltage was ramping up at the time, the low-voltage reset (LVD) status bit is also set to indicate that the reset occurred while the internal supply was below the LVD threshold. 0 Reset not caused by POR. 1 POR caused reset.
6 PIN	 External Reset Pin — Reset was caused by an active-low level on the external reset pin. 0 Reset not caused by external reset pin. 1 Reset came from external reset pin.
5 COP	 Computer Operating Properly (COP) Watchdog — Reset was caused by the COP watchdog timer timing out. This reset source can be blocked by COPE = 0. 0 Reset not caused by COP timeout. 1 Reset caused by COP timeout.
4 ILOP	 Illegal Opcode — Reset was caused by an attempt to execute an unimplemented or illegal opcode. The STOP instruction is considered illegal if stop is disabled by STOPE = 0 in the SOPT register. The BGND instruction is considered illegal if active background mode is disabled by ENBDM = 0 in the BDCSC register. 0 Reset not caused by an illegal opcode. 1 Reset caused by an illegal opcode.
3 ILAD	 Illegal Address — Reset was caused by an attempt to access either data or an instruction at an unimplemented memory address. 0 Reset not caused by an illegal address. 1 Reset caused by an illegal address.



An output pin can be selected to have high output drive strength by setting the corresponding bit in the drive strength select register (PTxDSn). When high drive is selected, a pin is capable of sourcing and sinking greater current. Even though every I/O pin can be selected as high drive, the user must ensure that the total current source and sink limits for the MCU are not exceeded. Drive strength selection is intended to affect the DC behavior of I/O pins. However, the AC behavior is also affected. High drive allows a pin to drive a greater load with the same switching speed as a low drive enabled pin into a smaller load. Because of this, the EMC emissions may be affected by enabling pins as high drive.

6.3 Pin Interrupts

Port A, port B, and port D pins can be configured as external interrupt inputs and as an external means of waking the MCU from stop or wait low-power modes.



The block diagram for each port interrupt logic is shown Figure 6-2.

Figure 6-2. Port Interrupt Block Diagram

Writing to the PTxPSn bits in the port interrupt pin select register (PTxPS) independently enables or disables each port pin. Each port can be configured as edge sensitive or edge and level sensitive based on the PTxMOD bit in the port interrupt status and control register (PTxSC). Edge sensitivity can be software programmed to be either falling or rising; the level can be either low or high. The polarity of the edge or edge and level sensitivity is selected using the PTxESn bits in the port interrupt edge select register (PTxSC).

Synchronous logic is used to detect edges. Prior to detecting an edge, enabled port inputs must be at the deasserted logic level. A falling edge is detected when an enabled port input signal is seen as a logic 1 (the deasserted level) during one bus cycle and then a logic 0 (the asserted level) during the next cycle. A rising edge is detected when the input signal is seen as a logic 0 during one bus cycle and then a logic 1 during the next cycle.

6.3.1 Edge Only Sensitivity

A valid edge on an enabled port pin will set PTxIF in PTxSC. If PTxIE in PTxSC is set, an interrupt request will be presented to the CPU. Clearing of PTxIF is accomplished by writing a 1 to PTxACK in PTxSC.



Chapter 7 Central Processor Unit (S08CPUV3)

Source	Operation	dress ode	Object Code	cles	Cyc-by-Cyc	Affect on CCR		
1 Onn		βq V		රි	Details	V 1 1 H	INZC	
RSP	Reset Stack Pointer (Low Byte) SPL ← \$FF (High Byte Not Affected)	INH	9C	1	ą	- 1 1 -		
RTI	Return from Interrupt SP \leftarrow (SP) + \$0001; Pull (CCR) SP \leftarrow (SP) + \$0001; Pull (A) SP \leftarrow (SP) + \$0001; Pull (X) SP \leftarrow (SP) + \$0001; Pull (PCH) SP \leftarrow (SP) + \$0001; Pull (PCL)	INH	80	9	uuuuufppp	↓11↓	↓↓↓↓	
RTS	Return from Subroutine SP \leftarrow SP + \$0001; Pull (PCH) SP \leftarrow SP + \$0001; Pull (PCL)	INH	81	5	ufppp	- 1 1 -		
SBC #opr8i SBC opr8a SBC opr16a SBC oprx16,X SBC oprx8,X SBC ,X SBC oprx16,SP SBC oprx8,SP	Subtract with Carry A \leftarrow (A) – (M) – (C)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A2 ii B2 dd C2 hh 11 D2 ee ff E2 ff F2 9E D2 ee ff 9E E2 ff	2 3 4 3 3 5 4	pp rpp prpp rpp rfp pprpp prpp	↓11-	- ↓ ↓ ↓	
SEC	Set Carry Bit $(C \leftarrow 1)$	INH	99	1	p	- 1 1 -	1	
SEI	Set Interrupt Mask Bit $(I \leftarrow 1)$	INH	9B	1	q	- 1 1 -	1 – – –	
STA opr8a STA opr16a STA oprx16,X STA oprx8,X STA ,X STA oprx16,SP STA oprx8,SP	Store Accumulator in Memory $M \leftarrow (A)$	DIR EXT IX2 IX1 IX SP2 SP1	B7 dd C7 hh 11 D7 ee ff E7 ff F7 9E D7 ee ff 9E E7 ff	3 4 3 2 5 4	БмББ ББмББ мБ БмББ БмББ БмББ АмББ	011-	- \$ \$ -	
STHX opr8a STHX opr16a STHX oprx8,SP	Store H:X (Index Reg.) (M:M + \$0001) ← (H:X)	DIR EXT SP1	35 dd 96 hh 11 9E FF ff	4 5 5	bambb bambb bambb	011-	- ‡ ‡ -	
STOP	Enable Interrupts: Stop Processing Refer to MCU Documentation I bit \leftarrow 0; Stop Processing	INH	8E	2	fp	- 1 1 -	0	
STX opr8a STX opr16a STX oprx16,X STX oprx8,X STX ,X STX oprx16,SP STX oprx8,SP	Store X (Low 8 Bits of Index Register) in Memory $M \leftarrow (X)$	DIR EXT IX2 IX1 IX SP2 SP1	BF dd CF hh ll DF ee ff EF ff FF 9E DF ee ff 9E EF ff	3 4 3 2 5 4	БмББ ББмББ мБ БмББ БмББ БмББ АмББ	011-	- \$ \$ -	



Chapter 7 Central Processor Unit (S08CPUV3)

Bit-Manip	oulation	Branch	Rea	ad-Modify-Write	Contro	1		Register/	lemory		
				9E60 6 NEG 3 SP1				S 2	9ED0 5 SUB 4 SP2	9EE0 4 SUB 3 SP1	
				9E61 6 CBEQ 4 SP1				S 2	9ED1 5 CMP 4 SP2	9EE1 4 CMP 3 SP1	
								2	9ED2 5 SBC 4 SP2	9EE2 4 SBC 3 SP1	
				9E63 6 COM 3 SP1				2	9ED3 5 CPX 4 SP2	9EE3 4 CPX 3 SP1	9EF3 6 CPHX 3 SP1
				9E64 6 LSR 3 SP1				2	9ED4 5 AND 4 SP2	9EE4 4 AND 3 SP1	
								2	9ED5 5 BIT 4 SP2	9EE5 4 BIT 3 SP1	
				9E66 6 ROR 3 SP1				2	DED6 5 LDA 4 SP2	9EE6 4 LDA 3 SP1	
				9E67 6 ASR 3 SP1				2	9ED7 5 STA 4 SP2	9EE7 4 STA 3 SP1	
				9E68 6 LSL 3 SP1				2	ED8 5 EOR SP2	9EE8 4 EOR 3 SP1	
				9E69 6 ROL 3 SP1				2	DED9 5 ADC 4 SP2	9EE9 4 ADC 3 SP1	
				9E6A 6 DEC 3 SP1				2	OEDA 5 ORA 4 SP2	9EEA 4 ORA 3 SP1	
				9E6B 8 DBNZ 4 SP1				2	DEDB 5 ADD 4 SP2	9EEB 4 ADD 3 SP1	
				9E6C 6 INC 3 SP1							
				9E6D 5 TST 3 SP1							
						9EAE 5 LDHX 2 IX	9EBE 6 LDHX 4 IX2	9ECE 5 LDHX 3 IX1 4	DEDE 5 LDX 4 SP2	9EEE 4 LDX 3 SP1	9EFE 5 LDHX 3 SP1
				9E6F 6 CLR 3 SP1					9EDF 5 STX 4 SP2	9EEF 4 STX 3 SP1	9EFF 5 STHX 3 SP1

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

Inherent Immediate Direct Extended DIR to DIR IX+ to DIR INH IMM DIR EXT DD IX+D

REL IX IX1 IX2 IMD DIX+ Relative Indexed, No Offset Indexed, 8-Bit Offset Indexed, 16-Bit Offset IMM to DIR DIR to IX+ Stack Pointer, 8-Bit Offset Stack Pointer, 16-Bit Offset Indexed, No Offset with Post Increment Indexed, 1-Byte Offset with Post Increment

SP1 SP2 IX+

IX1+

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

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



Chapter 8 Multi-Purpose Clock Generator (S08MCGV1)



10.2.1 Analog Power (V_{DDAD})

The ADC analog portion uses V_{DDAD} as its power connection. In some packages, V_{DDAD} is connected internally to V_{DD} . If externally available, connect the V_{DDAD} pin to the same voltage potential as V_{DD} . External filtering may be necessary to ensure clean V_{DDAD} for good results.

10.2.2 Analog Ground (V_{SSAD})

The ADC analog portion uses V_{SSAD} as its ground connection. In some packages, V_{SSAD} is connected internally to V_{SS} . If externally available, connect the V_{SSAD} pin to the same voltage potential as V_{SS} .

10.2.3 Voltage Reference High (V_{REFH})

 V_{REFH} is the high reference voltage for the converter. In some packages, V_{REFH} is connected internally to V_{DDAD} . If externally available, V_{REFH} may be connected to the same potential as V_{DDAD} or may be driven by an external source between the minimum V_{DDAD} spec and the V_{DDAD} potential (V_{REFH} must never exceed V_{DDAD}).

10.2.4 Voltage Reference Low (V_{REFL})

 V_{REFL} is the low-reference voltage for the converter. In some packages, V_{REFL} is connected internally to V_{SSAD} . If externally available, connect the V_{REFL} pin to the same voltage potential as V_{SSAD} .

10.2.5 Analog Channel Inputs (ADx)

The ADC module supports up to 28 separate analog inputs. An input is selected for conversion through the ADCH channel select bits.

10.3 Register Definition

These memory-mapped registers control and monitor operation of the ADC:

- Status and control register, ADCSC1
- Status and control register, ADCSC2
- Data result registers, ADCRH and ADCRL
- Compare value registers, ADCCVH and ADCCVL
- Configuration register, ADCCFG
- Pin control registers, APCTL1, APCTL2, APCTL3

10.3.1 Status and Control Register 1 (ADCSC1)

This section describes the function of the ADC status and control register (ADCSC1). Writing ADCSC1 aborts the current conversion and initiates a new conversion (if the ADCH bits are equal to a value other than all 1s).



10.5 Initialization Information

This section gives an example that provides some basic direction on how to initialize and configure the ADC module. You can configure the module for 8-, 10-, or 12-bit resolution, single or continuous conversion, and a polled or interrupt approach, among many other options. Refer to Table 10-7, Table 10-8, and Table 10-9 for information used in this example.

NOTE

Hexadecimal values designated by a preceding 0x, binary values designated by a preceding %, and decimal values have no preceding character.

10.5.1 ADC Module Initialization Example

10.5.1.1 Initialization Sequence

Before the ADC module can be used to complete conversions, an initialization procedure must be performed. A typical sequence is as follows:

- 1. Update the configuration register (ADCCFG) to select the input clock source and the divide ratio used to generate the internal clock, ADCK. This register is also used for selecting sample time and low-power configuration.
- 2. Update status and control register 2 (ADCSC2) to select the conversion trigger (hardware or software) and compare function options, if enabled.
- 3. Update status and control register 1 (ADCSC1) to select whether conversions will be continuous or completed only once, and to enable or disable conversion complete interrupts. The input channel on which conversions will be performed is also selected here.

10.5.1.2 Pseudo-Code Example

In this example, the ADC module is set up with interrupts enabled to perform a single 10-bit conversion at low power with a long sample time on input channel 1, where the internal ADCK clock is derived from the bus clock divided by 1.

ADCCFG = 0x98 (%10011000)

Bit 7	ADLPC	1	Configures for low power (lowers maximum clock speed)
Bit 6:5	ADIV	00	Sets the ADCK to the input clock \div 1
Bit 4	ADLSMP	1	Configures for long sample time
Bit 3:2	MODE	10	Sets mode at 10-bit conversions
Bit 1:0	ADICLK	00	Selects bus clock as input clock source

ADCSC2 = 0x00 (%00000000)

Bit	7	ADACT	0
Bit	6	ADTRG	0
Bit	5	ACFE	0
Bit	4	ACFGT	0
Bit	3:2		00
Bit	1:0		00

Flag indicates if a conversion is in progress
Software trigger selected
Compare function disabled
Not used in this example
Reserved, always reads zero
Reserved for Freescale's internal use; always write zero



Field	Description
1 SLPRQ⁵	Sleep Mode Request — This bit requests the MSCAN to enter sleep mode, which is an internal power saving mode (see Section 12.5.5.4, "MSCAN Sleep Mode"). The sleep mode request is serviced when the CAN bus is idle, i.e., the module is not receiving a message and all transmit buffers are empty. The module indicates entry to sleep mode by setting SLPAK = 1 (see Section 12.3.2, "MSCAN Control Register 1 (CANCTL1)"). SLPRQ cannot be set while the WUPIF flag is set (see Section 12.3.4.1, "MSCAN Receiver Flag Register (CANRFLG)"). Sleep mode will be active until SLPRQ is cleared by the CPU or, depending on the setting of WUPE, the MSCAN detects activity on the CAN bus and clears SLPRQ itself. 0 Running — The MSCAN functions normally 1 Sleep mode request — The MSCAN enters sleep mode when CAN bus idle
0 INITRQ ^{6,7}	Initialization Mode Request — When this bit is set by the CPU, the MSCAN skips to initialization mode (see Section 12.5.5.5, "MSCAN Initialization Mode"). Any ongoing transmission or reception is aborted and synchronization to the CAN bus is lost. The module indicates entry to initialization mode by setting INITAK = 1 (Section 12.3.2, "MSCAN Control Register 1 (CANCTL1)"). The following registers enter their hard reset state and restore their default values: CANCTL0 ⁸ , CANRFLG ⁹ , CANRIER ¹⁰ , CANTFLG, CANTIER, CANTARQ, CANTAAK, and CANTBSEL. The registers CANCTL1, CANBTR0, CANBTR1, CANIDAC, CANIDAR0-7, and CANIDMR0-7 can only be written by the CPU when the MSCAN is in initialization mode. (INITRQ = 1 and INITAK = 1). The values of the error counters are not affected by initialization mode. When this bit is cleared by the CPU, the MSCAN restarts and then tries to synchronize to the CAN bus. If the MSCAN is not in bus-off state, it synchronizes after 11 consecutive recessive bits on the CAN bus; if the MSCAN is in initialization mode is exited, which is INITRQ = 0 and INITAK = 0. 0 Normal operation 1 MSCAN in initialization mode

Table 12-1. CANCTL0 Register Field Descriptions (continued)

¹ The MSCAN must be in normal mode for this bit to become set.

² See the Bosch CAN 2.0A/B specification for a detailed definition of transmitter and receiver states.

- ³ In order to protect from accidentally violating the CAN protocol, the TXCAN pin is immediately forced to a recessive state when the CPU enters wait (CSWAI = 1) or stop mode (see Section 12.5.5.2, "Operation in Wait Mode" and Section 12.5.5.3, "Operation in Stop Mode").
- ⁴ The CPU has to make sure that the WUPE bit and the WUPIE wake-up interrupt enable bit (see Section 12.3.5, "MSCAN Receiver Interrupt Enable Register (CANRIER)) is enabled, if the recovery mechanism from stop or wait is required.
- ⁵ The CPU cannot clear SLPRQ before the MSCAN has entered sleep mode (SLPRQ = 1 and SLPAK = 1).
- ⁶ The CPU cannot clear INITRQ before the MSCAN has entered initialization mode (INITRQ = 1 and INITAK = 1).
- ⁷ In order to protect from accidentally violating the CAN protocol, the TXCAN pin is immediately forced to a recessive state when the initialization mode is requested by the CPU. Thus, the recommended procedure is to bring the MSCAN into sleep mode (SLPRQ = 1 and SLPAK = 1) before requesting initialization mode.
- ⁸ Not including WUPE, INITRQ, and SLPRQ.
- ⁹ TSTAT1 and TSTAT0 are not affected by initialization mode.

¹⁰ RSTAT1 and RSTAT0 are not affected by initialization mode.



NOTE

The CANTIER 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 when not in initialization mode

Table 12-12. CANTIER Register Field Descriptions

Field	Description
2:0	Transmitter Empty Interrupt Enable
TXEIE[2:0]	0 No interrupt request is generated from this event.
	1 A transmitter empty (transmit buffer available for transmission) event causes a transmitter empty interrupt
	request. See Section 12.5.2.2, "Transmit Structures" for details.

12.3.8 MSCAN Transmitter Message Abort Request Register (CANTARQ)

The CANTARQ register allows abort request of messages queued for transmission.



Figure 12-12. MSCAN Transmitter Message Abort Request Register (CANTARQ)

NOTE

The CANTARQ 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 when not in initialization mode

Field	Description
2:0 ABTRQ[2:0]	 Abort Request — The CPU sets the ABTRQx bit to request that a scheduled message buffer (TXEx = 0) be aborted. The MSCAN grants the request if the message has not already started transmission, or if the transmission is not successful (lost arbitration or error). When a message is aborted, the associated TXE (see Section 12.3.6, "MSCAN Transmitter Flag Register (CANTFLG)") and abort acknowledge flags (ABTAK, see Section 12.3.9, "MSCAN Transmitter Message Abort Acknowledge Register (CANTAAK)") are set and a transmit interrupt occurs if enabled. The CPU cannot reset ABTRQx. ABTRQx is reset whenever the associated TXE flag is set. 0 No abort request 1 Abort request pending

Offset Address	Register	Access
0x00X0	Identifier Register 0	
0x00X1	Identifier Register 1	
0x00X2	Identifier Register 2	
0x00X3	Identifier Register 3	
0x00X4	Data Segment Register 0	
0x00X5	Data Segment Register 1	
0x00X6	Data Segment Register 2	
0x00X7	Data Segment Register 3	
0x00X8	Data Segment Register 4	
0x00X9	Data Segment Register 5	
0x00XA	Data Segment Register 6	
0x00XB	Data Segment Register 7	
0x00XC	Data Length Register	
0x00XD	Transmit Buffer Priority Register ¹	
0x00XE	Time Stamp Register (High Byte) ²	
0x00XF	Time Stamp Register (Low Byte) ³	

Table 12-24. Message Buffer Organization

¹ Not applicable for receive buffers

² Read-only for CPU

³ Read-only for CPU

Figure 12-23 shows the common 13-byte data structure of receive and transmit buffers for extended identifiers. The mapping of standard identifiers into the IDR registers is shown in Figure 12-24.

All bits of the receive and transmit buffers are 'x' out of reset because of RAM-based implementation¹. All reserved or unused bits of the receive and transmit buffers always read 'x'.

^{1.} Exception: The transmit priority registers are 0 out of reset.



Chapter 13 Serial Peripheral Interface (S08SPIV3)

13.1 Introduction

The serial peripheral interface (SPI) module provides for full-duplex, synchronous, serial communication between the MCU and peripheral devices. These peripheral devices can include other microcontrollers, analog-to-digital converters, shift registers, sensors, memories, etc.

The SPI runs at a baud rate up to the bus clock divided by two in master mode and bus clock divided by four in slave mode.

All devices in the MC9S08DV60 Series MCUs contain one SPI module, as shown in the following block diagram.

NOTE

Ensure that the SPI should not be disabled (SPE=0) at the same time as a bit change to the CPHA bit. These changes should be performed as separate operations or unexpected behavior may occur.

Chapter 13 Serial Peripheral Interface (S08SPIV3)



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 SPSCK 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



Field	Description
7 TDRE	Transmit Data Register Empty Flag — TDRE is set out of reset and when a transmit data value transfers from the transmit data buffer to the transmit shifter, leaving room for a new character in the buffer. To clear TDRE, read SCIxS1 with TDRE = 1 and then write to the SCI data register (SCIxD). 0 Transmit data register (buffer) full. 1 Transmit data register (buffer) empty.
6 TC	 Transmission Complete Flag — TC is set out of reset and when TDRE = 1 and no data, preamble, or break character is being transmitted. 0 Transmitter active (sending data, a preamble, or a break). 1 Transmitter idle (transmission activity complete). TC is cleared automatically by reading SCIxS1 with TC = 1 and then doing one of the following three things: Write to the SCI data register (SCIxD) to transmit new data Queue a preamble by changing TE from 0 to 1 Queue a break character by writing 1 to SBK in SCIxC2
5 RDRF	 Receive Data Register Full Flag — RDRF becomes set when a character transfers from the receive shifter into the receive data register (SCIxD). To clear RDRF, read SCIxS1 with RDRF = 1 and then read the SCI data register (SCIxD). 0 Receive data register empty. 1 Receive data register full.
4 IDLE	Idle Line Flag — IDLE is set when the SCI receive line becomes idle for a full character time after a period of activity. When ILT = 0, the receiver starts counting idle bit times after the start bit. So if the receive character is all 1s, these bit times and the stop bit time count toward the full character time of logic high (10 or 11 bit times depending on the M control bit) needed for the receiver to detect an idle line. When ILT = 1, the receiver doesn't start counting idle bit times until after the stop bit. So the stop bit and any logic high bit times at the end of the previous character do not count toward the full character time of logic high needed for the receiver to detect an idle line. To clear IDLE, read SCIxS1 with IDLE = 1 and then read the SCI data register (SCIxD). After IDLE has been cleared, it cannot become set again until after a new character has been received and RDRF has been set. IDLE will get set only once even if the receive line remains idle for an extended period. 0 No idle line was detected.
3 OR	 Receiver Overrun Flag — OR is set when a new serial character is ready to be transferred to the receive data register (buffer), but the previously received character has not been read from SCIxD yet. In this case, the new character (and all associated error information) is lost because there is no room to move it into SCIxD. To clear OR, read SCIxS1 with OR = 1 and then read the SCI data register (SCIxD). 0 No overrun. 1 Receive overrun (new SCI data lost).
2 NF	 Noise Flag — The advanced sampling technique used in the receiver takes seven samples during the start bit and three samples in each data bit and the stop bit. If any of these samples disagrees with the rest of the samples within any bit time in the frame, the flag NF will be set at the same time as the flag RDRF gets set for the character. To clear NF, read SCIxS1 and then read the SCI data register (SCIxD). 0 No noise detected. 1 Noise detected in the received character in SCIxD.

Table 14-6. SCIxS1 Field Descriptions



Field	Description
1 LBKDE	 LIN Break Detection Enable— LBKDE is used to select a longer break character detection length. While LBKDE is set, framing error (FE) and receive data register full (RDRF) flags are prevented from setting. 0 Break character is detected at length of 10 bit times (11 if M = 1). 1 Break character is detected at length of 11 bit times (12 if M = 1).
0 RAF	 Receiver Active Flag — RAF is set when the SCI receiver detects the beginning of a valid start bit, and RAF is cleared automatically when the receiver detects an idle line. This status flag can be used to check whether an SCI character is being received before instructing the MCU to go to stop mode. 0 SCI receiver idle waiting for a start bit. 1 SCI receiver active (RxD input not idle).

Table 14-7. SCIxS2 Field Descriptions (continued)

¹ Setting RXINV inverts the RxD input for all cases: data bits, start and stop bits, break, and idle.

When using an internal oscillator in a LIN system, it is necessary to raise the break detection threshold by one bit time. Under the worst case timing conditions allowed in LIN, it is possible that a 0x00 data character can appear to be 10.26 bit times long at a slave which is running 14% faster than the master. This would trigger normal break detection circuitry which is designed to detect a 10 bit break symbol. When the LBKDE bit is set, framing errors are inhibited and the break detection threshold changes from 10 bits to 11 bits, preventing false detection of a 0x00 data character as a LIN break symbol.

14.2.6 SCI Control Register 3 (SCIxC3)



Figure 14-10. SCI Control Register 3 (SCIxC3)

Field	Description
7 R8	Ninth Data Bit for Receiver — When the SCI is configured for 9-bit data ($M = 1$), R8 can be thought of as a ninth receive data bit to the left of the MSB of the buffered data in the SCIxD register. When reading 9-bit data, read R8 before reading SCIxD because reading SCIxD completes automatic flag clearing sequences which could allow R8 and SCIxD to be overwritten with new data.
6 T8	Ninth Data Bit for Transmitter — When the SCI is configured for 9-bit data (M = 1), T8 may be thought of as a ninth transmit data bit to the left of the MSB of the data in the SCIxD register. When writing 9-bit data, the entire 9-bit value is transferred to the SCI shift register after SCIxD is written so T8 should be written (if it needs to change from its previous value) before SCIxD is written. If T8 does not need to change in the new value (such as when it is used to generate mark or space parity), it need not be written each time SCIxD is written.
5 TXDIR	TxD Pin Direction in Single-Wire Mode — When the SCI is configured for single-wire half-duplex operation(LOOPS = RSRC = 1), this bit determines the direction of data at the TxD pin.0TxD pin is an input in single-wire mode.1TxD pin is an output in single-wire mode.



Chapter 17 Development Support

The SYNC command is unlike other BDC commands because the host does not necessarily know the correct communications speed to use for BDC communications until after it has analyzed the response to the SYNC command.

To issue a SYNC command, the host:

- Drives the BKGD pin low for at least 128 cycles of the slowest possible BDC clock (The slowest clock is normally the reference oscillator/64 or the self-clocked rate/64.)
- Drives BKGD high for a brief speedup pulse to get a fast rise time (This speedup pulse is typically one cycle of the fastest clock in the system.)
- Removes all drive to the BKGD pin so it reverts to high impedance
- Monitors the BKGD pin for the sync response pulse

The target, upon detecting the SYNC request from the host (which is a much longer low time than would ever occur during normal BDC communications):

- Waits for BKGD to return to a logic high
- Delays 16 cycles to allow the host to stop driving the high speedup pulse
- Drives BKGD low for 128 BDC clock cycles
- Drives a 1-cycle high speedup pulse to force a fast rise time on BKGD
- Removes all drive to the BKGD pin so it reverts to high impedance

The host measures the low time of this 128-cycle sync response pulse and determines the correct speed for subsequent BDC communications. Typically, the host can determine the correct communication speed within a few percent of the actual target speed and the communication protocol can easily tolerate speed errors of several percent.

17.2.4 BDC Hardware Breakpoint

The BDC includes one relatively simple hardware breakpoint that compares the CPU address bus to a 16-bit match value in the BDCBKPT register. This breakpoint can generate a forced breakpoint or a tagged breakpoint. A forced breakpoint causes the CPU to enter active background mode at the first instruction boundary following any access to the breakpoint address. The tagged breakpoint causes the instruction opcode at the breakpoint address to be tagged so that the CPU will enter active background mode rather than executing that instruction if and when it reaches the end of the instruction queue. This implies that tagged breakpoints can only be placed at the address of an instruction opcode while forced breakpoints can be set at any address.

The breakpoint enable (BKPTEN) control bit in the BDC status and control register (BDCSCR) is used to enable the breakpoint logic (BKPTEN = 1). When BKPTEN = 0, its default value after reset, the breakpoint logic is disabled and no BDC breakpoints are requested regardless of the values in other BDC breakpoint registers and control bits. The force/tag select (FTS) control bit in BDCSCR is used to select forced (FTS = 1) or tagged (FTS = 0) type breakpoints.

The on-chip debug module (DBG) includes circuitry for two additional hardware breakpoints that are more flexible than the simple breakpoint in the BDC module.



Num	с	Rating	Symbol	Value	Unit	Temp. Code
1	D	Operating temperature range (packaged)	T _A	-40 to 125 -40 to 105 -40 to 85	°C	M V C
2	Т	Maximum Junction Temperature ¹	Т _Ј	135	°C	_
3	D	Thermal resistance ² Single-layer board				
		64-pin LQFP	θ_{JA}	69	°C/W	
		48-pin LQFP	θ_{JA}	75	°C/W	
		32-pin LQFP	θ_{JA}	80	°C/W	
		Four-Layer board				
		64-pin LQFP	θ_{JA}	51	°C/W	
		48-pin LQFP	θ_{JA}	51	°C/W	
		32-pin LQFP	θ_{JA}	52	°C/W	

 Table A-3. Thermal Characteristics

¹ Junction temperature is a function of die size, on-chip power dissipation, package thermal resistance, mounting site (board) temperature, ambient temperature, air flow, power dissipation of other components on the board, and board thermal resistance.

² Junction to Ambient Natural Convection

The average chip-junction temperature (T_I) in °C can be obtained from:

$$T_{J} = T_{A} + (P_{D} \times \theta_{JA})$$
 Eqn. A-1

where:

 T_A = Ambient temperature, °C θ_{JA} = Package thermal resistance, junction-to-ambient, °C/W $P_D = P_{int} + P_{I/O}$ $P_{int} = I_{DD} \times V_{DD}$, Watts — chip internal power $P_{I/O}$ = Power dissipation on input and output pins — user determined

For most applications, $P_{I/O} \ll P_{int}$ and can be neglected. An approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected) is:

$$P_{D} = K \div (T_{J} + 273^{\circ}C) \qquad \qquad Eqn. A-2$$

Solving equations 1 and 2 for K gives:

$$K = P_D \times (T_A + 273^{\circ}C) + \theta_{JA} \times (P_D)^2$$
 Eqn. A-3



Appendix A Electrical Characteristics







Figure A-4. Pin Interrupt Timing

A.12.2 Timer/PWM

Synchronizer circuits determine the shortest input pulses that can be recognized or the fastest clock that can be used as the optional external source to the timer counter. These synchronizers operate from the current bus rate clock.

Num	С	Rating	Symbol	Min	Мах	Unit
1	_	External clock frequency	f _{TCLK}	dc	f _{Bus} /4	MHz
2	_	External clock period	t _{TCLK}	4	_	t _{cyc}
3	D	External clock high time	t _{clkh}	1.5	_	t _{cyc}
4	D	External clock low time	t _{clkl}	1.5	_	t _{cyc}
5	D	Input capture pulse width	t _{ICPW}	1.5	_	t _{cyc}

Table A-14. TPM Input Timing



Appendix B Timer Pulse-Width Modulator (TPMV2)

NOTE

This chapter refers to S08TPM version 2, which applies to the 3M05C and older mask sets of this device.)M74K and newer mask set devices use S08TPM version 3. If your device uses mask 0M74K or newer, please refer to Chapter 16, "Timer Pulse-Width Modulator (S08TPMV3) for information pertaining to that module.

The TPM uses one input/output (I/O) pin per channel, TPMxCHn where x is the TPM number (for example, 1 or 2) and n is the channel number (for example, 0–4). The TPM shares its I/O pins with general-purpose I/O port pins (refer to the Pins and Connections chapter for more information).

B.0.1 Features

The TPM has the following features:

- Each TPM may be configured for buffered, center-aligned pulse-width modulation (CPWM) on all channels
- Clock sources independently selectable per TPM (multiple TPMs device)
- Selectable clock sources (device dependent): bus clock, fixed system clock, external pin
- Clock prescaler taps for divide by 1, 2, 4, 8, 16, 32, 64, or 128
- 16-bit free-running or up/down (CPWM) count operation
- 16-bit modulus register to control counter range
- Timer system enable
- One interrupt per channel plus a terminal count interrupt for each TPM module (multiple TPMs device)
- Channel features:
 - Each channel may be input capture, output compare, or buffered edge-aligned PWM
 - Rising-edge, falling-edge, or any-edge input capture trigger
 - Set, clear, or toggle output compare action
 - Selectable polarity on PWM outputs

B.0.2 Block Diagram

Figure B-1 shows the structure of a TPM. Some MCUs include more than one TPM, with various numbers of channels.

