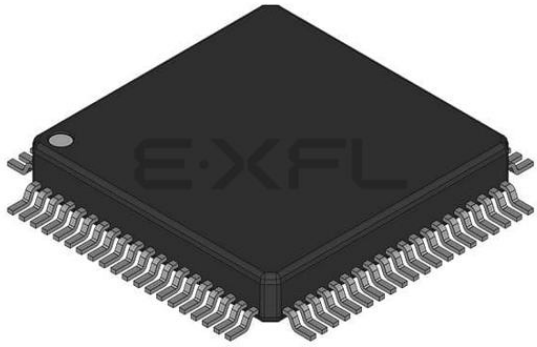

Welcome to E-XFL.COM

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 "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	60
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-QFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12gc128cfue

0x0140–0x017F CAN (Scalable Controller Area Network — MSCAN)[1] (continued)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0146	CANTFLG	Read:	0	0	0	0	0	TXE2	TXE1	TXE0
		Write:								
0x0147	CANTIER	Read:	0	0	0	0	0	TXEIE2	TXEIE1	TXEIE0
		Write:								
0x0148	CANTARQ	Read:	0	0	0	0	0	ABTRQ2	ABTRQ1	ABTRQ0
		Write:								
0x0149	CANTAAK	Read:	0	0	0	0	0	ABTAK2	ABTAK1	ABTAK0
		Write:								
0x014A	CANTBSEL	Read:	0	0	0	0	0	TX2	TX1	TX0
		Write:								
0x014B	CANIDAC	Read:	0	0	IDAM1	IDAM0	0	IDHIT2	IDHIT1	IDHIT0
		Write:								
0x014C	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								
0x014D	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								
0x014E	CANRXERR	Read:	RXERR7	RXERR6	RXERR5	RXERR4	RXERR3	RXERR2	RXERR1	RXERR0
		Write:								
0x014F	CANTXERR	Read:	TXERR7	TXERR6	TXERR5	TXERR4	TXERR3	TXERR2	TXERR1	TXERR0
		Write:								
0x0150–0x0153	CANIDAR0 - CANIDAR3	Read: Write:	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
0x0154–0x0157	CANIDMR0 - CANIDMR3	Read: Write:	AM7	AM6	AM5	AM4	AM3	AM2	AM1	AM0
0x0158–0x015B	CANIDAR4 - CANIDAR7	Read: Write:	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
0x015C–0x015F	CANIDMR4 - CANIDMR7	Read: Write:	AM7	AM6	AM5	AM4	AM3	AM2	AM1	AM0
0x0160–0x016F	CANRXFG	Read:	FOREGROUND RECEIVE BUFFER see Table 1-2							
		Write:								
0x0170–0x017F	CANTXFG	Read: Write:	FOREGROUND TRANSMIT BUFFER see Table 1-2							

1. Not available on the MC9S12GC Family members. Those memory locations should not be accessed.

Table 1-2. Detailed MSCAN Foreground Receive and Transmit Buffer Layout

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Extended ID	Read:	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21
0xXXX0	Standard ID	Read:	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3
	CANxRIDR0	Write:								
	Extended ID	Read:	ID20	ID19	ID18	SRR=1	IDE=1	ID17	ID16	ID15
0xXXX1	Standard ID	Read:	ID2	ID1	ID0	RTR	IDE=0			
	CANxRIDR1	Write:								

1.3.4.13 PE2 / R/$\overline{W}$ — Port E I/O Pin [2] / Read/Write

In all modes this pin can be used as a general-purpose I/O and is an input with an active pull-up out of reset. If the read/write function is required it should be enabled by setting the RDWE bit in the PEAR register. External writes will not be possible until enabled. This pin is not available in the 48- / 52-pin package versions.

1.3.4.14 PE1 / $\overline{IRQ}$ — Port E Input Pin [1] / Maskable Interrupt Pin

The $\overline{IRQ}$ input provides a means of applying asynchronous interrupt requests to the MCU. Either falling edge-sensitive triggering or level-sensitive triggering is program selectable (INTCR register). $\overline{IRQ}$ is always enabled and configured to level-sensitive triggering out of reset. It can be disabled by clearing IRQEN bit (INTCR register). When the MCU is reset the $\overline{IRQ}$ function is masked in the condition code register. This pin is always an input and can always be read. There is an active pull-up on this pin while in reset and immediately out of reset. The pull-up can be turned off by clearing PUPEE in the PUCR register.

1.3.4.15 PE0 / $\overline{XIRQ}$ — Port E input Pin [0] / Non Maskable Interrupt Pin

The $\overline{XIRQ}$ input provides a means of requesting a non-maskable interrupt after reset initialization. During reset, the X bit in the condition code register (CCR) is set and any interrupt is masked until MCU software enables it. Because the $\overline{XIRQ}$ input is level sensitive, it can be connected to a multiple-source wired-OR network. This pin is always an input and can always be read. There is an active pull-up on this pin while in reset and immediately out of reset. The pull-up can be turned off by clearing PUPEE in the PUCR register.

1.3.4.16 PAD[7:0] / AN[7:0] — Port AD I/O Pins [7:0]

PAD7–PAD0 are general purpose I/O pins and also analog inputs for the analog to digital converter. In order to use a PAD pin as a standard input, the corresponding ATDDIEN register bit must be set. These bits are cleared out of reset to configure the PAD pins for A/D operation.

When the A/D converter is active in multi-channel mode, port inputs are scanned and converted irrespective of Port AD configuration. Thus Port AD pins that are configured as digital inputs or digital outputs are also converted in the A/D conversion sequence.

1.3.4.17 PP[7] / KWP[7] — Port P I/O Pin [7]

PP7 is a general purpose input or output pin, shared with the keypad interrupt function. When configured as an input, it can generate interrupts causing the MCU to exit stop or wait mode. This pin is not available in the 48- / 52-pin package versions.

1.3.4.18 PP[6] / KWP[6]/ROMCTL — Port P I/O Pin [6]

PP6 is a general purpose input or output pin, shared with the keypad interrupt function. When configured as an input, it can generate interrupts causing the MCU to exit stop or wait mode. This pin is not available in the 48- / 52-pin package versions. During MCU expanded modes of operation, this pin is used to enable

1.3.5.4 V_{DDA}, V_{SSA} — Power Supply Pins for ATD and VREG

V_{DDA}, V_{SSA} are the power supply and ground input pins for the voltage regulator reference and the analog to digital converter.

1.3.5.5 V_{RH}, V_{RL} — ATD Reference Voltage Input Pins

V_{RH} and V_{RL} are the reference voltage input pins for the analog to digital converter.

1.3.5.6 V_{DDPLL}, V_{SSPLL} — Power Supply Pins for PLL

Provides operating voltage and ground for the oscillator and the phased-locked loop. This allows the supply voltage to the oscillator and PLL to be bypassed independently. This 2.5V voltage is generated by the internal voltage regulator.

Table 1-6. Power and Ground Connection Summary

Mnemonic	Nominal Voltage (V)	Description
$V_{DD1, VDD2}$	2.5	Internal power and ground generated by internal regulator. These also allow an external source to supply the core V_{DD}/V_{SS} voltages and bypass the internal voltage regulator. In the 48 and 52 LQFP packages V_{DD2} and V_{SS2} are not available.
$V_{SS1, VSS2}$	0	
V_{DDR}	5.0	External power and ground, supply to internal voltage regulator.
V_{SSR}	0	
V_{DDX}	5.0	External power and ground, supply to pin drivers.
V_{SSX}	0	
V_{DDA}	5.0	Operating voltage and ground for the analog-to-digital converters and the reference for the internal voltage regulator, allows the supply voltage to the A/D to be bypassed independently.
V_{SSA}	0	
V_{RH}	5.0	Reference voltage low for the ATD converter. In the 48 and 52 LQFP packages V_{RL} is bonded to V_{SSA}.
V_{RL}	0	
V_{DDPLL}	2.5	Provides operating voltage and ground for the phased-locked loop. This allows the supply voltage to the PLL to be bypassed independently. Internal power and ground generated by internal regulator.
V_{SSPLL}	0	

NOTE

All V_{SS} pins must be connected together in the application. Because fast signal transitions place high, short-duration current demands on the power supply, use bypass capacitors with high-frequency characteristics and place them as close to the MCU as possible. Bypass requirements depend on MCU pin load.

Table 1-11. Device Specific Flash PAGE Mapping

Device	PAGE	PAGE Visible with PPAGE Contents
MC9S12GC16	3F	$01,$03,$05,$07,$09.....$35,$37,$39,$3B,$3D,$3F
MC9S12C32 MC9S12GC32	3E	$00,$02,$04,$06,$08,$0A,$0C,$0E,$10,$12.....$2C,$2E,$30,$32,$34,$36,$38,$3A,$3C,$3E
	3F	$01,$03,$05,$07,$09,$0B,$0D,$0F,$11,$13.....$2D,$2F,$31,$33,$35,$37,$39,$3B,$3D,$3F
MC9S12C64 MC9S12GC64	3C	$04,$0C,$14,$1C,$24,$2C,$34,$3C
	3D	$05,$0D,$15,$1D,$25,$2D,$35,$3D
	3E	$06,$0E,$16,$1E,$26,$2E,$36,$3E
	3F	$07,$0F,$17,$1F,$27,$2F,$37,$3F
MC9S12C96 MC9S12GC96	3A	$02,$0A,$12,$1A,$22,$2A,$32,$3A
	3B	$03,$0B,$13,$1B,$23,$2B,$33,$3B
	3C	$04,$0C,$14,$1C,$24,$2C,$34,$3C
	3D	$05,$0D,$15,$1D,$25,$2D,$35,$3D
	3E	$06,$0E,$16,$1E,$26,$2E,$36,$3E
	3F	$07,$0F,$17,$1F,$27,$2F,$37,$3F
MC9S12C128 MC9S12GC128	38	$00,$08,$10,$18,$20,$28,$30,$38
	39	$01,$09,$11,$19,$21,$29,$31,$39
	3A	$02,$0A,$12,$1A,$22,$2A,$32,$3A
	3B	$03,$0B,$13,$1B,$23,$2B,$33,$3B
	3C	$04,$0C,$14,$1C,$24,$2C,$34,$3C
	3D	$05,$0D,$15,$1D,$25,$2D,$35,$3D
	3E	$06,$0E,$16,$1E,$26,$2E,$36,$3E
	3F	$07,$0F,$17,$1F,$27,$2F,$37,$3F

1.7.2 BDM Alternate Clock

The BDM section reference to alternate clock is equivalent to the oscillator clock.

1.7.3 Extended Address Range Emulation Implications

In order to emulate the MC9S12GC or MC9S12C-Family / MC9S12GC-Family devices, external addressing of a 128K memory map is required. This is provided in a 112 LQFP package version which includes the 3 necessary extra external address bus signals via PortK[2:0]. This package version is for emulation only and not provided as a general production package.

The reset state of DDRK is 0x0000, configuring the pins as inputs.

The reset state of PUPKE in the PUCR register is “1” enabling the internal Port K pullups.

In this reset state the pull-ups provide a defined state and prevent a floating input, thereby preventing unnecessary current flow at the input stage.

To prevent unnecessary current flow in production package options, the states of DDRK and PUPKE should not be changed by software.

2.3.2.1 Port T Registers

2.3.2.1.1 Port T I/O Register (PTT)

Module Base + 0x0000

	7	6	5	4	3	2	1	0
R W	PTT7	PTT6	PTT5	PTT4	PTT3	PTT2	PTT1	PTT0
TIM	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
PWM				PWM4	PWM3	PWM2	PWM1	PWM0
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-3. Port T I/O Register (PTT)

Read: Anytime.

Write: Anytime.

If the data direction bits of the associated I/O pins are set to 1, a read returns the value of the port register, otherwise the value at the pins is read.

If a TIM-channel is defined as output, the related port T is assigned to IOC function.

In addition to the possible timer functionality of port T pins PWM channels can be routed to port T. For this the Module Routing Register (MODRR) needs to be configured.

Table 2-3. Port T[4:0] Pin Functionality Configurations(1)

MODRR[x]	PWME[x]	TIMEN[x] (2)	Port T[x] Output
0	0	0	General Purpose I/O
0	0	1	Timer
0	1	0	General Purpose I/O
0	1	1	Timer
1	0	0	General Purpose I/O
1	0	1	Timer
1	1	0	PWM
1	1	1	PWM

1. All fields in the that are not shaded are standard use cases.
2. TIMEN[x] means that the timer is enabled (TSCR1[7]), the related channel is configured for output compare function (TIOS[x] or special output on a timer overflow event — configurable in TTOV[x]) and the timer output is routed to the port pin (TCTL1/TCTL2).

2.3.2.3.6 Port M Polarity Select Register (PPSM)

Module Base + 0x0015

	7	6	5	4	3	2	1	0
R	0	0	PPSM5	PPSM4	PPSM3	PPSM2	PPSM1	PPSM0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-22. Port M Polarity Select Register (PPSM)

Read: Anytime.

Write: Anytime.

Table 2-20. PPSM Field Descriptions

Field	Description
5–0 PPSM[5:0]	**Polarity Select Port M** — This register selects whether a pull-down or a pull-up device is connected to the pin. 0 A pull-up device is connected to the associated port M pin, if enabled by the associated bit in register PERM and if the port is used as input or as wired-or output. 1 A pull-down device is connected to the associated port M pin, if enabled by the associated bit in register PERM and if the port is used as input.

2.3.2.3.7 Port M Wired-OR Mode Register (WOMM)

Module Base + 0x0016

	7	6	5	4	3	2	1	0
R	0	0	WOMM5	WOMM4	WOMM3	WOMM2	WOMM1	WOMM0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-23. Port M Wired-OR Mode Register (WOMM)

Read: Anytime.

Write: Anytime.

Table 2-21. WOMM Field Descriptions

Field	Description
5–0 WOMM[5:0]	**Wired-OR Mode Port M** — This register configures the output pins as wired-or. If enabled the output is driven active low only (open-drain). A logic level of “1” is not driven. This bit has no influence on pins used as inputs. 0 Output buffers operate as push-pull outputs. 1 Output buffers operate as open-drain outputs.

Chapter 3
Module Mapping Control (MMCV4) Block Description

3.1 Introduction

This section describes the functionality of the module mapping control (MMC) sub-block of the S12 core platform.

The block diagram of the MMC is shown in Figure 3-1.

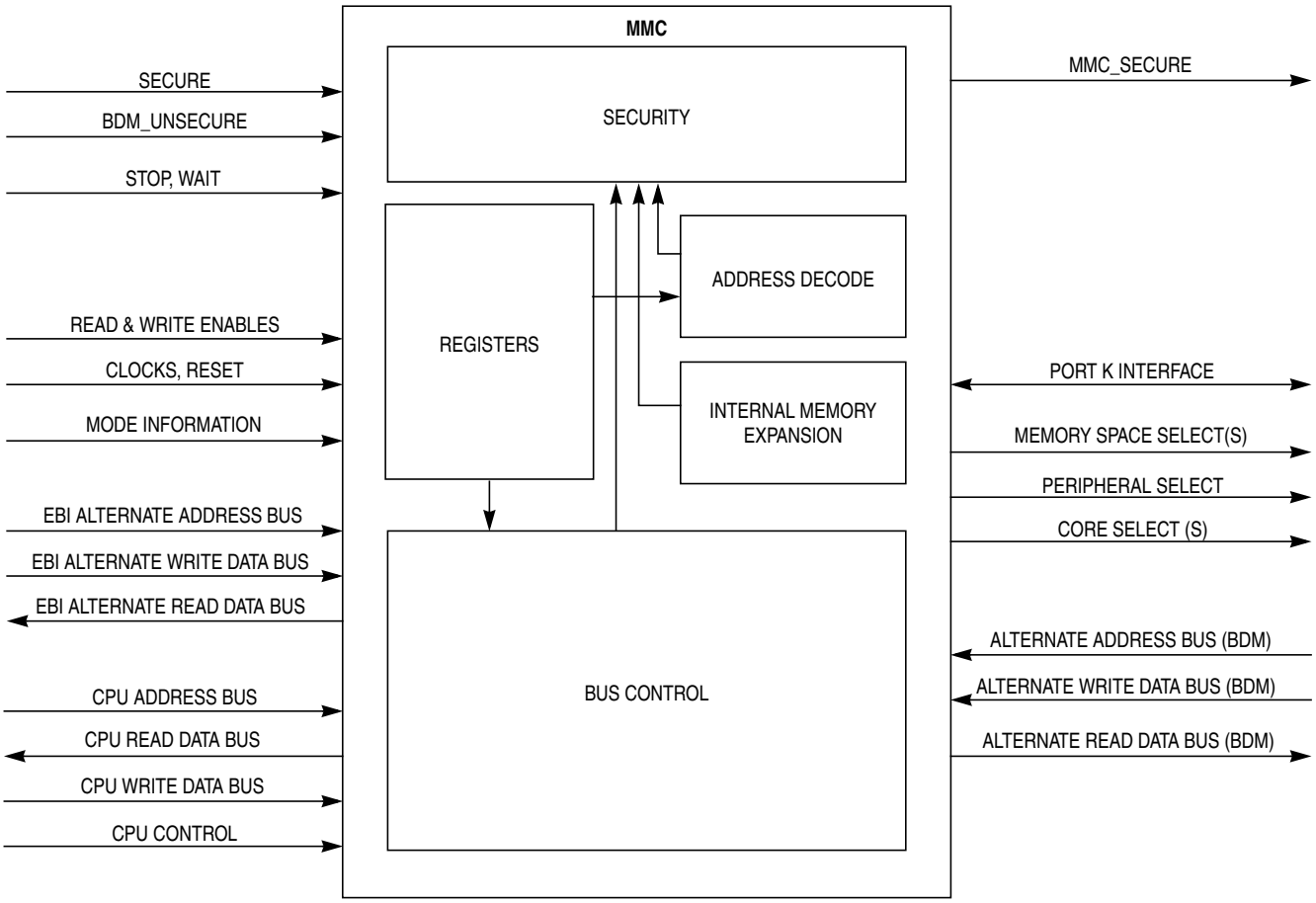

Figure 3-1. MMC Block Diagram

The MMC is the sub-module which controls memory map assignment and selection of internal resources and external space. Internal buses between the core and memories and between the core and peripherals is controlled in this module. The memory expansion is generated in this module.

8.3.2.3 ATD Control Register 2 (ATDCTL2)

This register controls power down, interrupt, and external trigger. Writes to this register will abort current conversion sequence but will not start a new sequence.

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R	ADPU	AFFC	AWAI	ETRIGLE	ETRIGP	ETRIGE	ASCIE	ASCIF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 8-5. ATD Control Register 2 (ATDCTL2)

Read: Anytime

Write: Anytime

Table 8-1. ATDCTL2 Field Descriptions

Field	Description
7 ADPU	**ATD Power Down** — This bit provides on/off control over the ATD10B8C block allowing reduced MCU power consumption. Because analog electronic is turned off when powered down, the ATD requires a recovery time period after ADPU bit is enabled. 0 Power down ATD 1 Normal ATD functionality
6 AFFC	**ATD Fast Flag Clear All** 0 ATD flag clearing operates normally (read the status register ATDSTAT1 before reading the result register to clear the associate CCF flag). 1 Changes all ATD conversion complete flags to a fast clear sequence. Any access to a result register will cause the associate CCF flag to clear automatically.
5 AWAI	**ATD Power Down in Wait Mode** — When entering Wait Mode this bit provides on/off control over the ATD10B8C block allowing reduced MCU power. Because analog electronic is turned off when powered down, the ATD requires a recovery time period after exit from Wait mode. 0 ATD continues to run in Wait mode 1 Halt conversion and power down ATD during Wait mode After exiting Wait mode with an interrupt conversion will resume. But due to the recovery time the result of this conversion should be ignored.
4 ETRIGLE	**External Trigger Level/Edge Control** — This bit controls the sensitivity of the external trigger signal. See Table 8-2 for details.
3 ETRIGP	**External Trigger Polarity** — This bit controls the polarity of the external trigger signal. See Table 8-2 for details.
2 ETRIGE	**External Trigger Mode Enable** — This bit enables the external trigger on ATD channel 7. The external trigger allows to synchronize sample and ATD conversions processes with external events. 0 Disable external trigger 1 Enable external trigger **Note:** The conversion results for the external trigger ATD channel 7 have no meaning while external trigger mode is enabled.

9.1.2 Modes of Operation

This subsection lists and briefly describes all operating modes supported by the CRG.

- Run mode

  All functional parts of the CRG are running during normal run mode. If RTI or COP functionality is required the individual bits of the associated rate select registers (COPCTL, RTICTL) have to be set to a nonzero value.

- Wait mode

  This mode allows to disable the system and core clocks depending on the configuration of the individual bits in the CLKSEL register.

- Stop mode

  Depending on the setting of the PSTP bit, stop mode can be differentiated between full stop mode (PSTP = 0) and pseudo-stop mode (PSTP = 1).

  — Full stop mode

  The oscillator is disabled and thus all system and core clocks are stopped. The COP and the RTI remain frozen.

  — Pseudo-stop mode

  The oscillator continues to run and most of the system and core clocks are stopped. If the respective enable bits are set the COP and RTI will continue to run, else they remain frozen.

- Self-clock mode

  Self-clock mode will be entered if the clock monitor enable bit (CME) and the self-clock mode enable bit (SCME) are both asserted and the clock monitor in the oscillator block detects a loss of clock. As soon as self-clock mode is entered the CRGV4 starts to perform a clock quality check. Self-clock mode remains active until the clock quality check indicates that the required quality of the incoming clock signal is met (frequency and amplitude). Self-clock mode should be used for safety purposes only. It provides reduced functionality to the MCU in case a loss of clock is causing severe system conditions.

9.1.3 Block Diagram

Figure 9-1 shows a block diagram of the CRGV4.

The MSCAN is able to leave sleep mode (wake up) only when:

- CAN bus activity occurs and WUPE = 1

  or
- the CPU clears the SLPRQ bit

NOTE

The CPU cannot clear the SLPRQ bit before sleep mode (SLPRQ = 1 and SLPAK = 1) is active.

After wake-up, the MSCAN waits for 11 consecutive recessive bits to synchronize to the CAN bus. As a consequence, if the MSCAN is woken-up by a CAN frame, this frame is not received.

The receive message buffers (RxFG and RxBG) contain messages if they were received before sleep mode was entered. All pending actions will be executed upon wake-up; copying of RxBG into RxFG, message aborts and message transmissions. If the MSCAN remains in bus-off state after sleep mode was exited, it continues counting the 128 occurrences of 11 consecutive recessive bits.

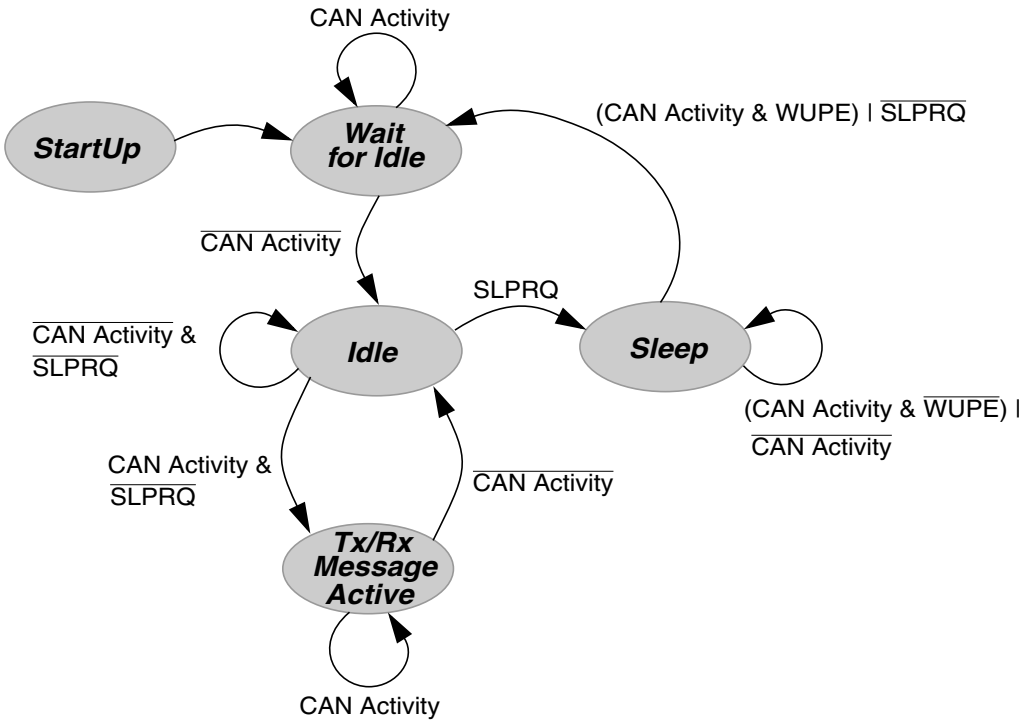

Figure 10-45. Simplified State Transitions for Entering/Leaving Sleep Mode

Table 12-3. PWMPOL Field Descriptions (continued)

Field	Description
3 PPOL3	**Pulse Width Channel 3 Polarity** 0 PWM channel 3 output is low at the beginning of the period, then goes high when the duty count is reached. 1 PWM channel 3 output is high at the beginning of the period, then goes low when the duty count is reached.
2 PPOL2	**Pulse Width Channel 2 Polarity** 0 PWM channel 2 output is low at the beginning of the period, then goes high when the duty count is reached. 1 PWM channel 2 output is high at the beginning of the period, then goes low when the duty count is reached.
1 PPOL1	**Pulse Width Channel 1 Polarity** 0 PWM channel 1 output is low at the beginning of the period, then goes high when the duty count is reached. 1 PWM channel 1 output is high at the beginning of the period, then goes low when the duty count is reached.
0 PPOL0	**Pulse Width Channel 0 Polarity** 0 PWM channel 0 output is low at the beginning of the period, then goes high when the duty count is reached 1 PWM channel 0 output is high at the beginning of the period, then goes low when the duty count is reached.

12.3.2.3 PWM Clock Select Register (PWMCLK)

Each PWM channel has a choice of two clocks to use as the clock source for that channel as described below.

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R	0	0	PCLK5	PCLK4	PCLK3	PCLK2	PCLK1	PCLK0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-5. PWM Clock Select Register (PWMCLK)

Read: anytime

Write: anytime

NOTE

Register bits PCLK0 to PCLK5 can be written anytime. If a clock select is changed while a PWM signal is being generated, a truncated or stretched pulse can occur during the transition.

14.3.2 Register Descriptions

This section consists of register descriptions in address order. Each description includes a standard register diagram with an associated figure number. Details of register bit and field function follow the register diagrams, in bit order.

Name		7	6	5	4	3	2	1	0
0x0000 SPICR1	R	SPIE	SPE	SPTIE	MSTR	CPOL	CPHA	SSOE	LSBFE
	W								
0x0001 SPICR2	R	0	0	0	MODFEN	BIDIROE	0	SPISWAI	SPC0
	W								
0x0002 SPIBR	R	0	SPPR2	SPPR1	SPPR0	0	SPR2	SPR1	SPR0
	W								
0x0003 SPISR	R	SPIF	0	SPTEF	MODF	0	0	0	0
	W								
0x0004 Reserved	R								
	W								
0x0005 SPIDR	R	Bit 7	6	5	4	3	2	2	Bit 0
	W								
0x0006 Reserved	R								
	W								
0x0007 Reserved	R								
	W								

= Unimplemented or Reserved

Figure 14-2. SPI Register Summary

14.3.2.1 SPI Control Register 1 (SPICR1)

Module Base 0x0000

	7	6	5	4	3	2	1	0
R	SPIE	SPE	SPTIE	MSTR	CPOL	CPHA	SSOE	LSBFE
W								
Reset	0	0	0	0	0	1	0	0

Figure 14-3. SPI Control Register 1 (SPICR1)

Read: anytime

Write: anytime

14.4.3 Transmission Formats

During an SPI transmission, data is transmitted (shifted out serially) and received (shifted in serially) simultaneously. The serial clock (SCK) synchronizes shifting and sampling of the information on the two serial data lines. A slave select line allows selection of an individual slave SPI device, slave devices that are not selected do not interfere with SPI bus activities. Optionally, on a master SPI device, the slave select line can be used to indicate multiple-master bus contention.

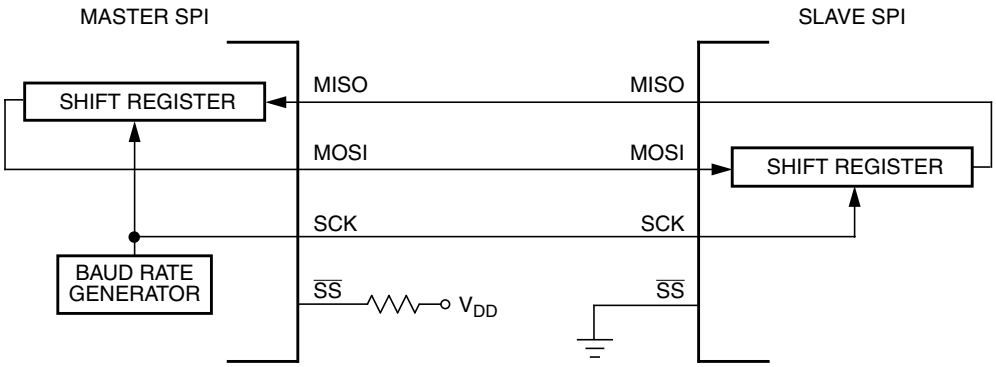

Figure 14-8. Master/Slave Transfer Block Diagram

14.4.3.1 Clock Phase and Polarity Controls

Using two bits in the SPI Control Register1, software selects one of four combinations of serial clock phase and polarity.

The CPOL clock polarity control bit specifies an active high or low clock and has no significant effect on the transmission format.

The CPHA clock phase control bit selects one of two fundamentally different transmission formats.

Clock phase and polarity should be identical for the master SPI device and the communicating slave device. In some cases, the phase and polarity are changed between transmissions to allow a master device to communicate with peripheral slaves having different requirements.

14.4.3.2 CPHA = 0 Transfer Format

The first edge on the SCK line is used to clock the first data bit of the slave into the master and the first data bit of the master into the slave. In some peripherals, the first bit of the slave's data is available at the slave's data out pin as soon as the slave is selected. In this format, the first SCK edge is issued a half cycle after $\overline{SS}$ has become low.

A half SCK cycle later, the second edge appears on the SCK line. When this second edge occurs, the value previously latched from the serial data input pin is shifted into the LSB or MSB of the shift register, depending on LSBFE bit.

After this second edge, the next bit of the SPI master data is transmitted out of the serial data output pin of the master to the serial input pin on the slave. This process continues for a total of 16 edges on the SCK line, with data being latched on odd numbered edges and shifted on even numbered edges.

In master mode, with slave select output enabled the $\overline{SS}$ line is always deasserted and reasserted between successive transfers for at least minimum idle time.

14.4.3.3 CPHA = 1 Transfer Format

Some peripherals require the first SCK edge before the first data bit becomes available at the data out pin, the second edge clocks data into the system. In this format, the first SCK edge is issued by setting the CPHA bit at the beginning of the 8-cycle transfer operation.

The first edge of SCK occurs immediately after the half SCK clock cycle synchronization delay. This first edge commands the slave to transfer its first data bit to the serial data input pin of the master.

A half SCK cycle later, the second edge appears on the SCK pin. This is the latching edge for both the master and slave.

When the third edge occurs, the value previously latched from the serial data input pin is shifted into the LSB or MSB of the SPI shift register, depending on LSBFE bit. After this edge, the next bit of the master data is coupled out of the serial data output pin of the master to the serial input pin on the slave.

This process continues for a total of 16 edges on the SCK line with data being latched on even numbered edges and shifting taking place on odd numbered edges.

Data reception is double buffered, data is serially shifted into the SPI shift register during the transfer and is transferred to the parallel SPI Data Register after the last bit is shifted in.

After the 16th SCK edge:

- Data that was previously in the SPI Data Register of the master is now in the data register of the slave, and data that was in the data register of the slave is in the master.
- The SPIF flag bit in SPISR is set indicating that the transfer is complete.

Figure 14-10 shows two clocking variations for CPHA = 1. The diagram may be interpreted as a master or slave timing diagram because the SCK, MISO, and MOSI pins are connected directly between the master and the slave. The MISO signal is the output from the slave, and the MOSI signal is the output from the master. The $\overline{SS}$ line is the slave select input to the slave. The $\overline{SS}$ pin of the master must be either high or reconfigured as a general-purpose output not affecting the SPI.

The $\overline{SS}$ line can remain active low between successive transfers (can be tied low at all times). This format is sometimes preferred in systems having a single fixed master and a single slave that drive the MISO data line.

- Back-to-back transfers in master mode

In master mode, if a transmission has completed and a new data byte is available in the SPI Data Register, this byte is send out immediately without a trailing and minimum idle time.

The SPI interrupt request flag (SPIF) is common to both the master and slave modes. SPIF gets set one half SCK cycle after the last SCK edge.

15.2.6 IOC2 — Input Capture and Output Compare Channel 2 Pin

This pin serves as input capture or output compare for channel 2.

15.2.7 IOC1 — Input Capture and Output Compare Channel 1 Pin

This pin serves as input capture or output compare for channel 1.

15.2.8 IOC0 — Input Capture and Output Compare Channel 0 Pin

This pin serves as input capture or output compare for channel 0.

NOTE

For the description of interrupts see Section 15.6, “Interrupts”.

15.3 Memory Map and Register Definition

This section provides a detailed description of all memory and registers.

15.3.1 Module Memory Map

The memory map for the TIM16B8CV1 module is given below in Table 15-2. The address listed for each register is the address offset. The total address for each register is the sum of the base address for the TIM16B8CV1 module and the address offset for each register.

Write: Anytime

Table 15-3. TIOS Field Descriptions

Field	Description
7:0 IOS[7:0]	**Input Capture or Output Compare Channel Configuration** 0 The corresponding channel acts as an input capture. 1 The corresponding channel acts as an output compare.

15.3.2.2 Timer Compare Force Register (CFORC)

Module Base + 0x0001

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W	FOC7	FOC6	FOC5	FOC4	FOC3	FOC2	FOC1	FOC0
Reset	0	0	0	0	0	0	0	0

Figure 15-7. Timer Compare Force Register (CFORC)

Read: Anytime but will always return 0x0000 (1 state is transient)

Write: Anytime

Table 15-4. CFORC Field Descriptions

Field	Description
7:0 FOC[7:0]	**Force Output Compare Action for Channel 7:0** — A write to this register with the corresponding data bit(s) set causes the action which is programmed for output compare “x” to occur immediately. The action taken is the same as if a successful comparison had just taken place with the TCx register except the interrupt flag does not get set. **Note:** A successful channel 7 output compare overrides any channel 6:0 compares. If forced output compare on any channel occurs at the same time as the successful output compare then forced output compare action will take precedence and interrupt flag won’t get set.

15.3.2.3 Output Compare 7 Mask Register (OC7M)

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R W	OC7M7	OC7M6	OC7M5	OC7M4	OC7M3	OC7M2	OC7M1	OC7M0
Reset	0	0	0	0	0	0	0	0

Figure 15-8. Output Compare 7 Mask Register (OC7M)

Read: Anytime

Write: Anytime

18.4.1.3.3 Sector Erase Command

The sector erase operation will erase all addresses in a 512 byte sector of the Flash array using an embedded algorithm.

An example flow to execute the sector erase operation is shown in Figure 18-24. The sector erase command write sequence is as follows:

1. Write to a Flash array address to start the command write sequence for the sector erase command. The Flash address written determines the sector to be erased while MCU address bits [8:0] and the data written are ignored.
2. Write the sector erase command, 0x40, to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the sector erase command.

If a Flash sector to be erased is in a protected area of the Flash array, the PVIOL flag in the FSTAT register will set and the sector erase command will not launch. Once the sector erase command has successfully launched, the CCIF flag in the FSTAT register will set after the sector erase operation has completed unless a new command write sequence has been buffered.

18.4.4 Flash Reset Sequence

On each reset, the Flash module executes a reset sequence to hold CPU activity while loading the following registers from the Flash array memory according to Table 18-1:

- FPROT — Flash Protection Register (see Section 18.3.2.5)
- FSEC — Flash Security Register (see Section 18.3.2.2)

18.4.4.1 Reset While Flash Command Active

If a reset occurs while any Flash command is in progress, that command will be immediately aborted. The state of the word being programmed or the sector/array being erased is not guaranteed.

18.4.5 Interrupts

The Flash module can generate an interrupt when all Flash commands have completed execution or the Flash address, data, and command buffers are empty.

Table 18-17. Flash Interrupt Sources

Interrupt Source	Interrupt Flag	Local Enable	Global (CCR) Mask
Flash Address, Data, and Command Buffers are empty	CBEIF (FSTAT register)	CBEIE	I Bit
All Flash commands have completed execution	CCIF (FSTAT register)	CCIE	I Bit

NOTE

Vector addresses and their relative interrupt priority are determined at the MCU level.

18.4.5.1 Description of Interrupt Operation

Figure 18-26 shows the logic used for generating interrupts.

The Flash module uses the CBEIF and CCIF flags in combination with the enable bits CBIE and CCIE to discriminate for the generation of interrupts.

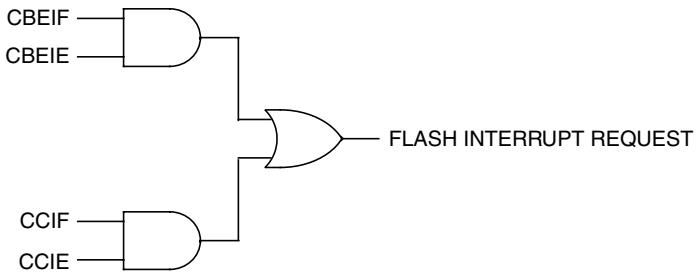

Figure 18-26. Flash Interrupt Implementation

For a detailed description of these register bits, refer to Section 18.3.2.4, “Flash Configuration Register (FCNFG)” and Section 18.3.2.6, “Flash Status Register (FSTAT)”.

Table 21-13. FSTAT Field Descriptions

Field	Description
5 PVIOL	**Protection Violation** — The PVIOL flag indicates an attempt was made to program or erase an address in a protected Flash array memory area. The PVIOL flag is cleared by writing a 1 to PVIOL. Writing a 0 to the PVIOL flag has no effect on PVIOL. While PVIOL is set, it is not possible to launch another command. 0 No protection violation detected 1 Protection violation has occurred
4 ACCERR	**Access Error** — The ACCERR flag indicates an illegal access to the Flash array caused by either a violation of the command write sequence, issuing an illegal command (illegal combination of the CMDBx bits in the FCMD register) or the execution of a CPU STOP instruction while a command is executing (CCIF=0). The ACCERR flag is cleared by writing a 1 to ACCERR. Writing a 0 to the ACCERR flag has no effect on ACCERR. While ACCERR is set, it is not possible to launch another command. 0 No access error detected 1 Access error has occurred
2 BLANK	**Flash Array Has Been Verified as Erased** — The BLANK flag indicates that an erase verify command has checked the Flash array and found it to be erased. The BLANK flag is cleared by hardware when CBEIF is cleared as part of a new valid command write sequence. Writing to the BLANK flag has no effect on BLANK. 0 If an erase verify command has been requested, and the CCIF flag is set, then a 0 in BLANK indicates the array is not erased 1 Flash array verifies as erased
1 FAIL	**Flag Indicating a Failed Flash Operation** — In special modes, the FAIL flag will set if the erase verify operation fails (Flash array verified as not erased). Writing a 0 to the FAIL flag has no effect on FAIL. The FAIL flag is cleared by writing a 1 to FAIL. While FAIL is set, it is not possible to launch another command. 0 Flash operation completed without error 1 Flash operation failed
0 DONE	**Flag Indicating a Failed Operation is not Active** — In special modes, the DONE flag will clear if a program, erase, or erase verify operation is active. 0 Flash operation is active 1 Flash operation is not active

21.3.2.7 Flash Command Register (FCMD)

The FCMD register defines the Flash commands.

Module Base + 0x0006

	7	6	5	4	3	2	1	0
R	0	CMDB6	CMDB5	0	0	CMDB2	0	CMDB0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 21-11. Flash Command Register (FCMD)

Bits CMDB6, CMDB5, CMDB2, and CMDB0 are readable and writable during a command write sequence while bits 7, 4, 3, and 1 read 0 and are not writable.