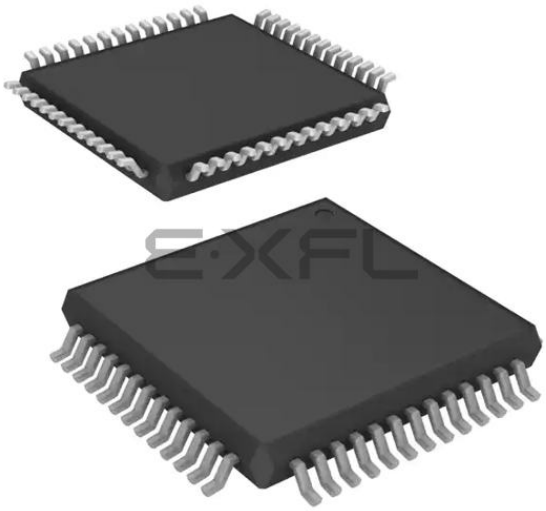

Welcome to E-XFL.COM

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	Obsolete
Core Processor	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	35
Program Memory Size	96KB (96K 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	52-LQFP
Supplier Device Package	52-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s12gc96cpber

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:								

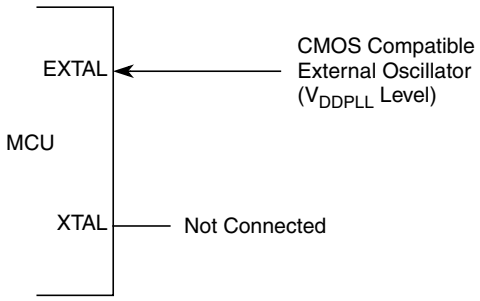

Figure 1-13. External Clock Connections (PE7 = 0)

1.3.4.9 PE6 / MODB / IPIPE1 — Port E I/O Pin 6

PE6 is a general purpose input or output pin. It is used as a MCU operating mode select pin during reset. The state of this pin is latched to the MODB bit at the rising edge of $\overline{\text{RESET}}$. This pin is shared with the instruction queue tracking signal IPIPE1. This pin is an input with a pull-down device which is only active when $\overline{\text{RESET}}$ is low. PE[6] is not available in the 48- / 52-pin package versions.

1.3.4.10 PE5 / MODA / IPIPE0 — Port E I/O Pin 5

PE5 is a general purpose input or output pin. It is used as a MCU operating mode select pin during reset. The state of this pin is latched to the MODA bit at the rising edge of $\overline{\text{RESET}}$. This pin is shared with the instruction queue tracking signal IPIPE0. This pin is an input with a pull-down device which is only active when $\overline{\text{RESET}}$ is low. This pin is not available in the 48- / 52-pin package versions.

1.3.4.11 PE4 / ECLK— Port E I/O Pin [4] / E-Clock Output

ECLK is the output connection for the internal bus clock. It is used to demultiplex the address and data in expanded modes and is used as a timing reference. ECLK frequency is equal to 1/2 the crystal frequency out of reset. The ECLK pin is initially configured as ECLK output with stretch in all expanded modes. The E clock output function depends upon the settings of the NECLK bit in the PEAR register, the IVIS bit in the MODE register and the ESTR bit in the EBICTL register. All clocks, including the E clock, are halted when the MCU is in stop mode. It is possible to configure the MCU to interface to slow external memory. ECLK can be stretched for such accesses. Reference the MISC register (EXSTR[1:0] bits) for more information. In normal expanded narrow mode, the E clock is available for use in external select decode logic or as a constant speed clock for use in the external application system. Alternatively PE4 can be used as a general purpose input or output pin.

1.3.4.12 PE3 / $\overline{\text{LSTRB}}$ — Port E I/O Pin [3] / Low-Byte Strobe ($\overline{\text{LSTRB}}$)

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 strobe function is required, it should be enabled by setting the LSTRE bit in the PEAR register. This signal is used in write operations. Therefore external low byte writes will not be possible until this function is enabled. This pin is also used as $\overline{\text{TAGLO}}$ in special expanded modes and is multiplexed with the $\overline{\text{LSTRB}}$ function. This pin is not available in the 48- / 52-pin package versions.

Chapter 2
Port Integration Module (PIM9C32) Block Description

2.1 Introduction

The Port Integration Module establishes the interface between the peripheral modules and the I/O pins for all ports.

This chapter covers:

- Port A, B, and E related to the core logic and the multiplexed bus interface
- Port T connected to the TIM module (PWM module can be routed to port T as well)
- Port S connected to the SCI module
- Port M associated to the MSCAN and SPI module
- Port P connected to the PWM module, external interrupt sources available
- Port J pins can be used as external interrupt sources and standard I/O's

The following I/O pin configurations can be selected:

- Available on all I/O pins:
  - Input/output selection
  - Drive strength reduction
  - Enable and select of pull resistors
- Available on all Port P and Port J pins:
  - Interrupt enable and status flags

The implementation of the Port Integration Module is device dependent.

2.1.1 Features

A standard port has the following minimum features:

- Input/output selection
- 5-V output drive with two selectable drive strength
- 5-V digital and analog input
- Input with selectable pull-up or pull-down device

Optional features:

- Open drain for wired-OR connections
- Interrupt inputs with glitch filtering

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.

2.3.2.5.3 Port J Data Direction Register (DDRJ)

Module Base + 0x002A

	7	6	5	4	3	2	1	0
R	DDRJ7	DDRJ6	0	0	0	0	0	0
W								
Reset	0	0	—	—	—	—	—	—

= Unimplemented or Reserved

Figure 2-34. Port J Data Direction Register (DDRJ)

Read: Anytime.

Write: Anytime.

Table 2-28. DDRJ Field Descriptions

Field	Description
7–6 DDRJ[7:6]	**Data Direction Port J** — This register configures port pins J[7:6] as either input or output. DDRJ[7:6] — Data Direction Port J 0 Associated pin is configured as input. 1 Associated pin is configured as output. **Note:** Due to internal synchronization circuits, it can take up to 2 bus cycles until the correct value is read on PTJ or PTIJ registers, when changing the DDRJ register.

2.3.2.5.4 Port J Reduced Drive Register (RDRJ)

Module Base + 0x002B

	7	6	5	4	3	2	1	0
R	RDRJ7	RDRJ6	0	0	0	0	0	0
W								
Reset	0	0	—	—	—	—	—	—

= Unimplemented or Reserved

Figure 2-35. Port J Reduced Drive Register (RDRJ)

Read: Anytime.

Write: Anytime.

Table 2-29. RDRJ Field Descriptions

Field	Description
7–6 RDRJ[7:6]	**Reduced Drive Port J** — This register configures the drive strength of each port J output pin as either full or reduced. If the port is used as input this bit is ignored. 0 Full drive strength at output. 1 Associated pin drives at about 1/3 of the full drive strength.

2.3.2.6 Port AD Registers

2.3.2.6.1 Port AD I/O Register (PTAD)

Module Base + 0x0030

	7	6	5	4	3	2	1	0
R W	PTAD7	PTAD6	PTAD5	PTAD4	PTAD3	PTAD2	PTAD1	PTAD0
Reset	0	0	0	0	0	0	0	0

Figure 2-40. Port AD I/O Register (PTAD)

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.

2.3.2.6.2 Port AD Input Register (PTIAD)

Module Base + 0x0031

	7	6	5	4	3	2	1	0
R	PTIAD7	PTIAD6	PTIAD5	PTIAD4	PTIAD3	PTIAD2	PTIAD1	PTIAD0
W								
Reset	—	—	—	—	—	—	—	—

= Unimplemented or Reserved

Figure 2-41. Port AD Input Register (PTIAD)

Read: Anytime.

Write: Never, writes to this register have no effect.

This register always reads back the status of the associated pins. This can be used to detect overload or short circuit conditions on output pins.

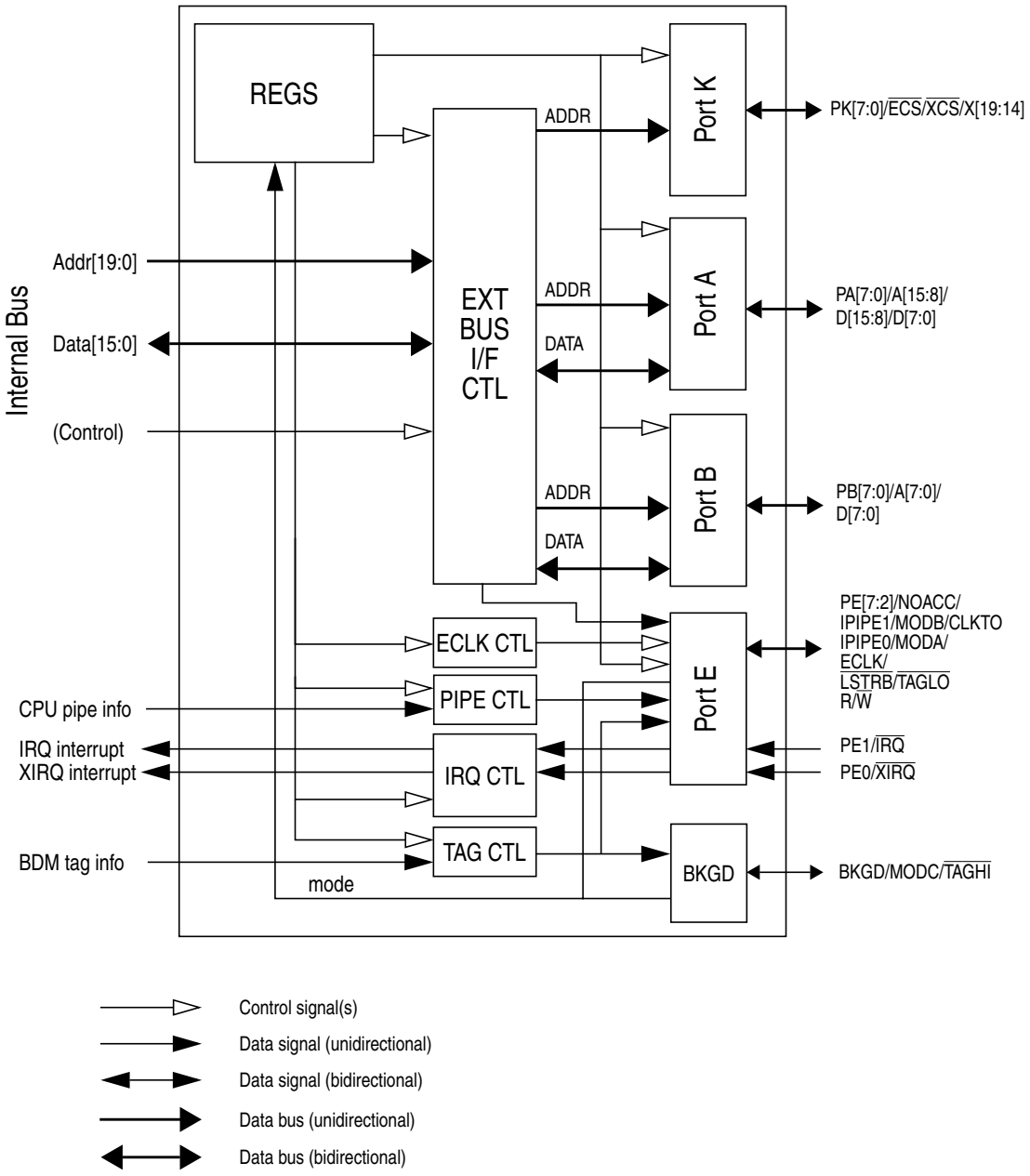

Figure 4-1. MEBI Block Diagram

4.4.3.1.2 Normal Expanded Wide Mode

In expanded wide modes, Ports A and B are configured as a 16-bit multiplexed address and data bus and Port E bit 4 is configured as the E clock output signal. These signals allow external memory and peripheral devices to be interfaced to the MCU.

Port E pins other than PE4/ECLK are configured as general purpose I/O pins (initially high-impedance inputs with internal pull resistors enabled). Control bits PIPOE, NECLK, LSTRE, and RDWE in the PEAR register can be used to configure Port E pins to act as bus control outputs instead of general purpose I/O pins.

It is possible to enable the pipe status signals on Port E bits 6 and 5 by setting the PIPOE bit in PEAR, but it would be unusual to do so in this mode. Development systems where pipe status signals are monitored would typically use the special variation of this mode.

The Port E bit 2 pin can be reconfigured as the R/$\overline{W}$ bus control signal by writing “1” to the RDWE bit in PEAR. If the expanded system includes external devices that can be written, such as RAM, the RDWE bit would need to be set before any attempt to write to an external location. If there are no writable resources in the external system, PE2 can be left as a general purpose I/O pin.

The Port E bit 3 pin can be reconfigured as the $\overline{LSTRB}$ bus control signal by writing “1” to the LSTRE bit in PEAR. The default condition of this pin is a general purpose input because the $\overline{LSTRB}$ function is not needed in all expanded wide applications.

The Port E bit 4 pin is initially configured as ECLK output with stretch. The E clock output function depends upon the settings of the NECLK bit in the PEAR register, the IVIS bit in the MODE register and the ESTR bit in the EBICTL register. The E clock is available for use in external select decode logic or as a constant speed clock for use in the external application system.

4.4.3.1.3 Normal Expanded Narrow Mode

This mode is used for lower cost production systems that use 8-bit wide external EPROMs or RAMs. Such systems take extra bus cycles to access 16-bit locations but this may be preferred over the extra cost of additional external memory devices.

Ports A and B are configured as a 16-bit address bus and Port A is multiplexed with data. Internal visibility is not available in this mode because the internal cycles would need to be split into two 8-bit cycles.

Since the PEAR register can only be written one time in this mode, use care to set all bits to the desired states during the single allowed write.

The PE3/$\overline{LSTRB}$ pin is always a general purpose I/O pin in normal expanded narrow mode. Although it is possible to write the LSTRE bit in PEAR to “1” in this mode, the state of LSTRE is overridden and Port E bit 3 cannot be reconfigured as the $\overline{LSTRB}$ output.

It is possible to enable the pipe status signals on Port E bits 6 and 5 by setting the PIPOE bit in PEAR, but it would be unusual to do so in this mode. LSTRB would also be needed to fully understand system activity. Development systems where pipe status signals are monitored would typically use special expanded wide mode or occasionally special expanded narrow mode.

12.3.2.11 Reserved Registers (PWMSCNTx)

The registers PWMSCNTA and PWMSCNTB are reserved for factory testing of the PWM module and are not available in normal modes.

Module Base + 0x000A

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-13. Reserved Register (PWMSCNTA)

Module Base + 0x000B

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-14. Reserved Register (PWMSCNTB)

Read: always read 0x0000 in normal modes

Write: unimplemented in normal modes

NOTE

Writing to these registers when in special modes can alter the PWM functionality.

12.3.2.12 PWM Channel Counter Registers (PWMCNTx)

Each channel has a dedicated 8-bit up/down counter which runs at the rate of the selected clock source. The counter can be read at any time without affecting the count or the operation of the PWM channel. In left aligned output mode, the counter counts from 0 to the value in the period register – 1. In center aligned output mode, the counter counts from 0 up to the value in the period register and then back down to 0.

Any value written to the counter causes the counter to reset to 0x0000, the counter direction to be set to up, the immediate load of both duty and period registers with values from the buffers, and the output to change according to the polarity bit. The counter is also cleared at the end of the effective period (see Section 12.4.2.5, “Left Aligned Outputs,” and Section 12.4.2.6, “Center Aligned Outputs,” for more details). When the channel is disabled (PWMEx = 0), the PWMCNTx register does not count. When a channel becomes enabled (PWMEx = 1), the associated PWM counter starts at the count in the PWMCNTx register. For more detailed information on the operation of the counters, reference Section 12.4.2.4, “PWM Timer Counters.”

In concatenated mode, writes to the 16-bit counter by using a 16-bit access or writes to either the low- or high-order byte of the counter will reset the 16-bit counter. Reads of the 16-bit counter must be made by 16-bit access to maintain data coherency.

NOTE

Writing to the counter while the channel is enabled can cause an irregular PWM cycle to occur.

Module Base + 0x000C

	7	6	5	4	3	2	1	0
R	Bit 7	6	5	4	3	2	1	Bit 0
W	0	0	0	0	0	0	0	0
Reset	0	0	0	0	0	0	0	0

Figure 12-15. PWM Channel Counter Registers (PWMCNT0)

Module Base + 0x000D

	7	6	5	4	3	2	1	0
R	Bit 7	6	5	4	3	2	1	Bit 0
W	0	0	0	0	0	0	0	0
Reset	0	0	0	0	0	0	0	0

Figure 12-16. PWM Channel Counter Registers (PWMCNT1)

12.4.1.3 Clock Select

Each PWM channel has the capability of selecting one of two clocks. For channels 0, 1, 4, and 5 the clock choices are clock A or clock SA. For channels 2 and 3 the choices are clock B or clock SB. The clock selection is done with the PCLKx control bits in the PWMCLK register.

NOTE

Changing clock control bits while channels are operating can cause irregularities in the PWM outputs.

12.4.2 PWM Channel Timers

The main part of the PWM module are the actual timers. Each of the timer channels has a counter, a period register and a duty register (each are 8 bit). The waveform output period is controlled by a match between the period register and the value in the counter. The duty is controlled by a match between the duty register and the counter value and causes the state of the output to change during the period. The starting polarity of the output is also selectable on a per channel basis. Figure 12-35 shows a block diagram for PWM timer.

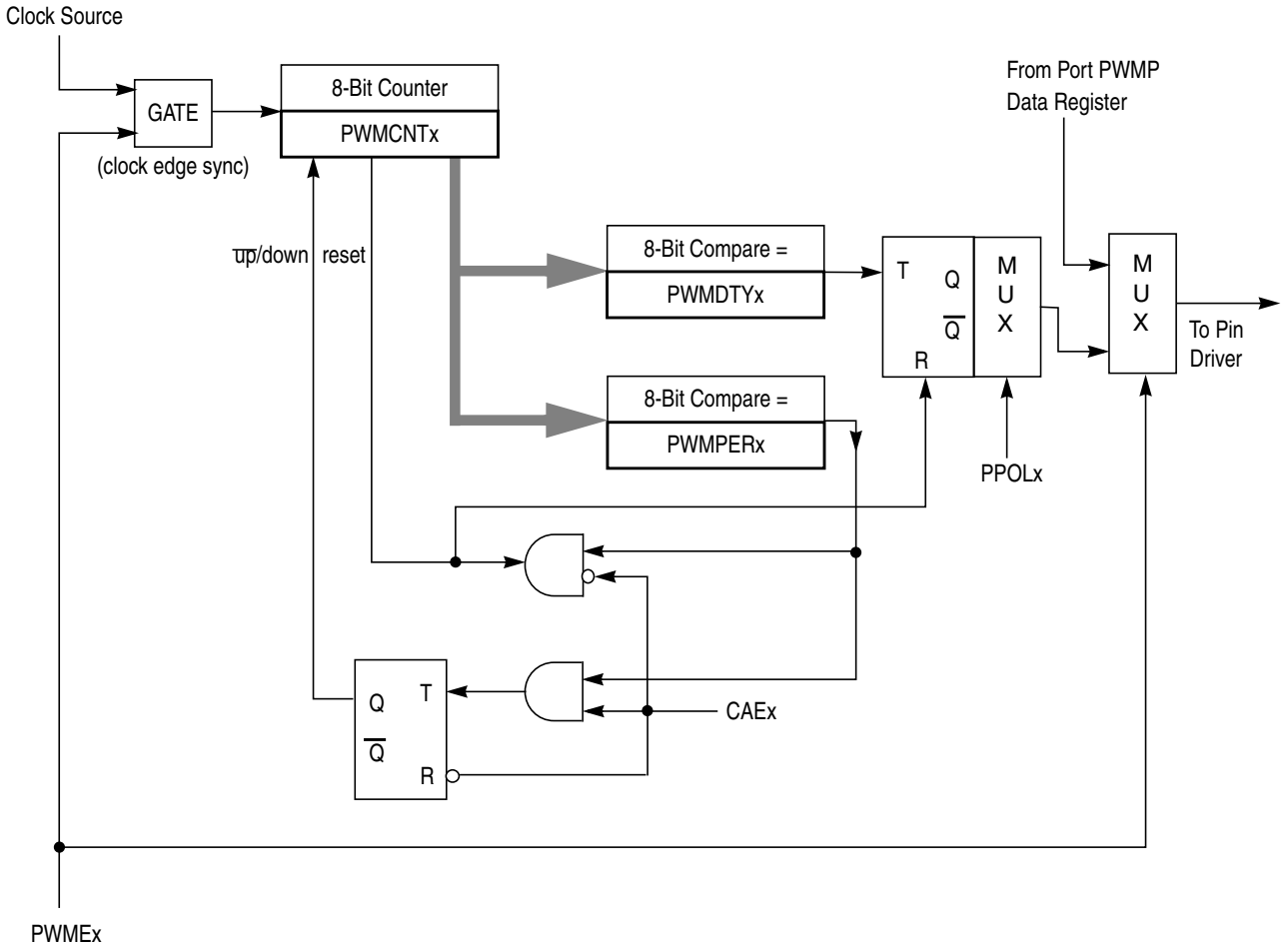

Figure 12-35. PWM Timer Channel Block Diagram

Table 13-4. SCICR2 Field Descriptions (continued)

Field	Description
1 RWU	**Receiver Wakeup Bit** — Standby state 0 Normal operation. 1 RWU enables the wakeup function and inhibits further receiver interrupt requests. Normally, hardware wakes the receiver by automatically clearing RWU.
0 SBK	**Send Break Bit** — Toggling SBK sends one break character (10 or 11 logic 0s, respectively 13 or 14 logics 0s if BRK13 is set). Toggling implies clearing the SBK bit before the break character has finished transmitting. As long as SBK is set, the transmitter continues to send complete break characters (10 or 11 bits, respectively 13 or 14 bits). 0 No break characters 1 Transmit break characters

13.3.2.4 SCI Status Register 1 (SCISR1)

The SCISR1 and SCISR2 registers provides inputs to the MCU for generation of SCI interrupts. Also, these registers can be polled by the MCU to check the status of these bits. The flag-clearing procedures require that the status register be read followed by a read or write to the SCI Data Register.It is permissible to execute other instructions between the two steps as long as it does not compromise the handling of I/O, but the order of operations is important for flag clearing.

Module Base + 0x_0004

	7	6	5	4	3	2	1	0
R	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 13-6. SCI Status Register 1 (SCISR1)

Read: Anytime

Write: Has no meaning or effect

Table 13-5. SCISR1 Field Descriptions

Field	Description
7 TDRE	**Transmit Data Register Empty Flag** — TDRE is set when the transmit shift register receives a byte from the SCI data register. When TDRE is 1, the transmit data register (SCIDRH/L) is empty and can receive a new value to transmit.Clear TDRE by reading SCI status register 1 (SCISR1), with TDRE set and then writing to SCI data register low (SCIDRL). 0 No byte transferred to transmit shift register 1 Byte transferred to transmit shift register; transmit data register empty
6 TC	**Transmit Complete Flag** — TC is set low when there is a transmission in progress or when a preamble or break character is loaded. TC is set high when the TDRE flag is set and no data, preamble, or break character is being transmitted.When TC is set, the TXD out signal becomes idle (logic 1). Clear TC by reading SCI status register 1 (SCISR1) with TC set and then writing to SCI data register low (SCIDRL). TC is cleared automatically when data, preamble, or break is queued and ready to be sent. TC is cleared in the event of a simultaneous set and clear of the TC flag (transmission not complete). 0 Transmission in progress 1 No transmission in progress

14.4.2 Slave Mode

The SPI operates in slave mode when the MSTR bit in SPI Control Register1 is clear.

- SCK Clock

  In slave mode, SCK is the SPI clock input from the master.
- MISO and MOSI Pins

  In slave mode, the function of the serial data output pin (MISO) and serial data input pin (MOSI) is determined by the SPC0 bit and BIDIROE bit in SPI Control Register 2.
- $\overline{SS}$ Pin

  The $\overline{SS}$ pin is the slave select input. Before a data transmission occurs, the $\overline{SS}$ pin of the slave SPI must be low. $\overline{SS}$ must remain low until the transmission is complete. If $\overline{SS}$ goes high, the SPI is forced into idle state.

  The $\overline{SS}$ input also controls the serial data output pin, if $\overline{SS}$ is high (not selected), the serial data output pin is high impedance, and, if $\overline{SS}$ is low the first bit in the SPI Data Register is driven out of the serial data output pin. Also, if the slave is not selected ($\overline{SS}$ is high), then the SCK input is ignored and no internal shifting of the SPI shift register takes place.

Although the SPI is capable of duplex operation, some SPI peripherals are capable of only receiving SPI data in a slave mode. For these simpler devices, there is no serial data out pin.

NOTE

When peripherals with duplex capability are used, take care not to simultaneously enable two receivers whose serial outputs drive the same system slave's serial data output line.

As long as no more than one slave device drives the system slave's serial data output line, it is possible for several slaves to receive the same transmission from a master, although the master would not receive return information from all of the receiving slaves.

If the CPHA bit in SPI Control Register 1 is clear, odd numbered edges on the SCK input cause the data at the serial data input pin to be latched. Even numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

If the CPHA bit is set, even numbered edges on the SCK input cause the data at the serial data input pin to be latched. Odd numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

When CPHA is set, the first edge is used to get the first data bit onto the serial data output pin. When CPHA is clear and the $\overline{SS}$ input is low (slave selected), the first bit of the SPI data is driven out of the serial data output pin. After the eighth shift, the transfer is considered complete and the received data is transferred into the SPI Data Register. To indicate transfer is complete, the SPIF flag in the SPI Status Register is set.

NOTE

A change of the bits CPOL, CPHA, SSOE, LSBFE, MODFEN, SPC0 and BIDIROE with SPC0 set in slave mode will corrupt a transmission in progress and has to be avoided.

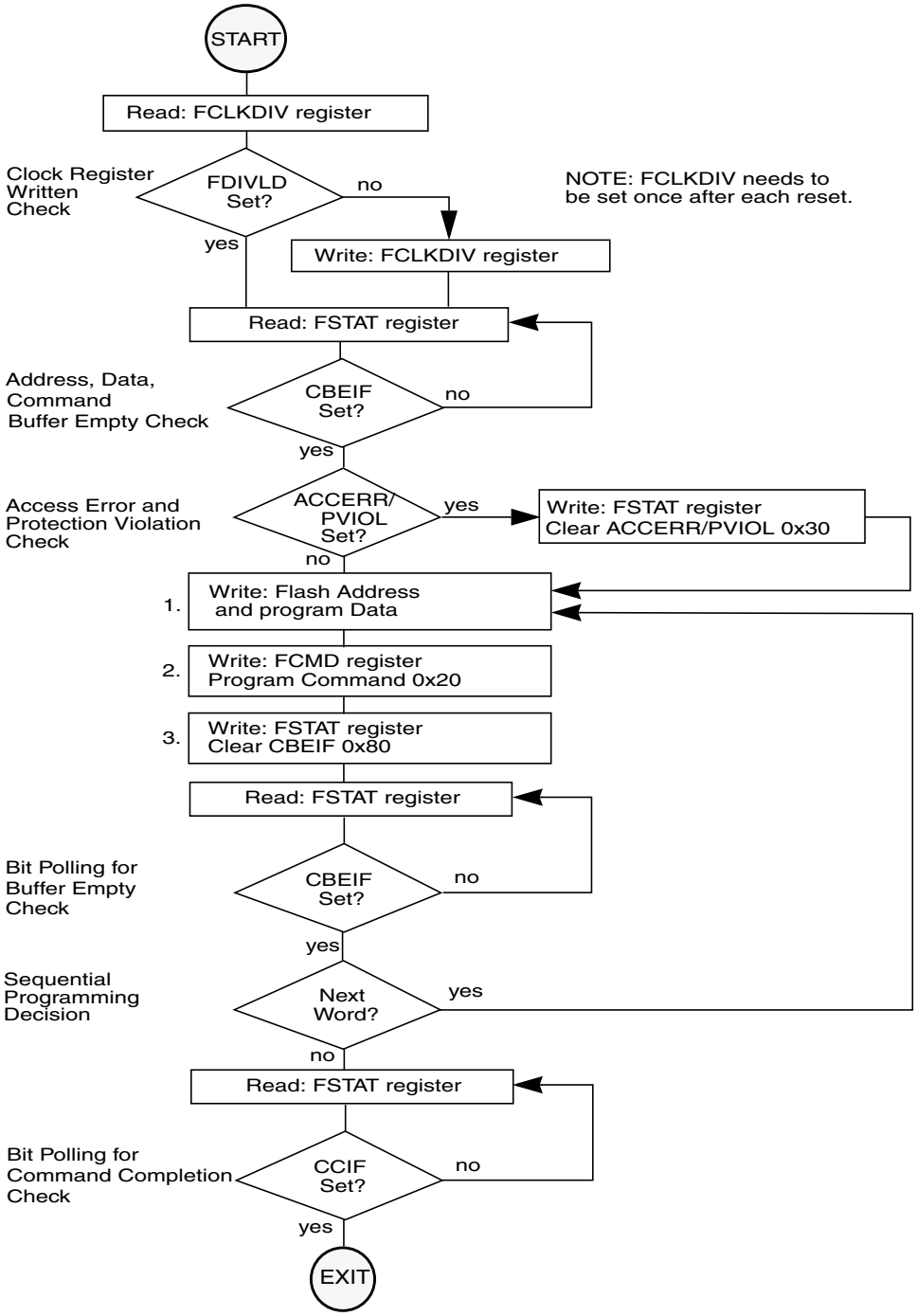

Figure 17-23. Example Program Command Flow

17.4.1.4 Illegal Flash Operations

17.4.1.4.1 Access Error

The ACCERR flag in the FSTAT register will be set during the command write sequence if any of the following illegal Flash operations are performed causing the command write sequence to immediately abort:

1. Writing to the Flash address space before initializing the FCLKDIV register
2. Writing a misaligned word or a byte to the valid Flash address space
3. Writing to the Flash address space while CBEIF is not set
4. Writing a second word to the Flash address space before executing a program or erase command on the previously written word
5. Writing to any Flash register other than FCMD after writing a word to the Flash address space
6. Writing a second command to the FCMD register before executing the previously written command
7. Writing an invalid command to the FCMD register
8. Writing to any Flash register other than FSTAT (to clear CBEIF) after writing to the FCMD register
9. The part enters stop mode and a program or erase command is in progress. The command is aborted and any pending command is killed
10. When security is enabled, a command other than mass erase originating from a non-secure memory or from the background debug mode is written to the FCMD register
11. A 0 is written to the CBEIF bit in the FSTAT register to abort a command write sequence.

The ACCERR flag will not be set if any Flash register is read during the command write sequence. If the Flash array is read during execution of an algorithm (CCIF=0), the Flash module will return invalid data and the ACCERR flag will not be set. If an ACCERR flag is set in the FSTAT register, the Flash command controller is locked. It is not possible to launch another command until the ACCERR flag is cleared.

17.4.1.4.2 Protection Violation

The PVIOL flag in the FSTAT register will be set during the command write sequence after the word write to the Flash address space if any of the following illegal Flash operations are performed, causing the command write sequence to immediately abort:

1. Writing a Flash address to program in a protected area of the Flash array (see Section 17.3.2.5).
2. Writing a Flash address to erase in a protected area of the Flash array.
3. Writing the mass erase command to the FCMD register while any protection is enabled.

If the PVIOL flag is set, the Flash command controller is locked. It is not possible to launch another command until the PVIOL flag is cleared.

19.4.1.2 Command Write Sequence

The Flash command controller is used to supervise the command write sequence to execute program, erase, and erase verify algorithms.

Before starting a command write sequence, the ACCERR and PVIOL flags in the FSTAT register must be clear and the CBEIF flag should be tested to determine the state of the address, data, and command buffers. If the CBEIF flag is set, indicating the buffers are empty, a new command write sequence can be started. If the CBEIF flag is clear, indicating the buffers are not available, a new command write sequence will overwrite the contents of the address, data, and command buffers.

A command write sequence consists of three steps which must be strictly adhered to with writes to the Flash module not permitted between the steps. However, Flash register and array reads are allowed during a command write sequence. The basic command write sequence is as follows:

1. Write to a valid address in the Flash array memory.
2. Write a valid command to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the command.

The address written in step 1 will be stored in the FADDR registers and the data will be stored in the FDATA registers. When the CBEIF flag is cleared in step 3, the CCIF flag is cleared by the Flash command controller indicating that the command was successfully launched. For all command write sequences, the CBEIF flag will set after the CCIF flag is cleared indicating that the address, data, and command buffers are ready for a new command write sequence to begin. A buffered command will wait for the active operation to be completed before being launched. Once a command is launched, the completion of the command operation is indicated by the setting of the CCIF flag in the FSTAT register. The CCIF flag will set upon completion of all active and buffered commands.

address 0x8000 to 0xBFFF to any physical 16K byte page in the Flash array memory.[1] The FPROT register (see Section 20.3.2.5) can be set to globally protect the entire Flash array. Three separate areas, one starting from the Flash array starting address (called lower) towards higher addresses, one growing downward from the Flash array end address (called higher), and the remaining addresses, can be activated for protection. The Flash array addresses covered by these protectable regions are shown in Figure 20-3Figure 20-4. The higher address area is mainly targeted to hold the boot loader code since it covers the vector space. The lower address area can be used for EEPROM emulation in an MCU without an EEPROM module since it can be left unprotected while the remaining addresses are protected from program or erase. Default protection settings as well as security information that allows the MCU to restrict access to the Flash module are stored in the Flash configuration field described in Table 20-1.

Table 20-1. Flash Configuration Field

Flash Address	Size (bytes)	Description
0xFF00–0xFF07	8	Backdoor Key to unlock security
0xFF08–0xFF0C	5	Reserved
0xFF0D	1	Flash Protection byte Refer to Section 20.3.2.5, “Flash Protection Register (FPROT)”
0xFF0E	1	Reserved
0xFF0F	1	Flash Security/Options byte Refer to Section 20.3.2.2, “Flash Security Register (FSEC)”

1. By placing 0x3E/0x3F in the HCS12 Core PPAGE register, the bottom/top fixed 16 Kbyte pages can be seen twice in the MCU memory map.

Table 21-4. FSEC Field Descriptions

Field	Description
7–6 KEYEN[1:0]	**Backdoor Key Security Enable Bits** — The KEYEN[1:0] bits define the enabling of the backdoor key access to the Flash module as shown in Table 21-5.
5–2 NV[5:2]	**Nonvolatile Flag Bits** — The NV[5:2] bits are available to the user as nonvolatile flags.
1–0 SEC[1:0]	**Flash Security Bits** — The SEC[1:0] bits define the security state of the MCU as shown in Table 21-6. If the Flash module is unsecured using backdoor key access, the SEC[1:0] bits are forced to 1:0.

Table 21-5. Flash KEYEN States

KEYEN[1:0]	Status of Backdoor Key Access
00	DISABLED
01(1)	DISABLED
10	ENABLED
11	DISABLED

1. Preferred KEYEN state to disable Backdoor Key Access.

Table 21-6. Flash Security States

SEC[1:0]	Status of Security
00	Secured
01(1)	Secured
10	Unsecured
11	Secured

1. Preferred SEC state to set MCU to secured state.

The security function in the Flash module is described in Section 21.4.3, “Flash Module Security”.

21.3.2.3 RESERVED1

This register is reserved for factory testing and is not accessible to the user.

Module Base + 0x0002

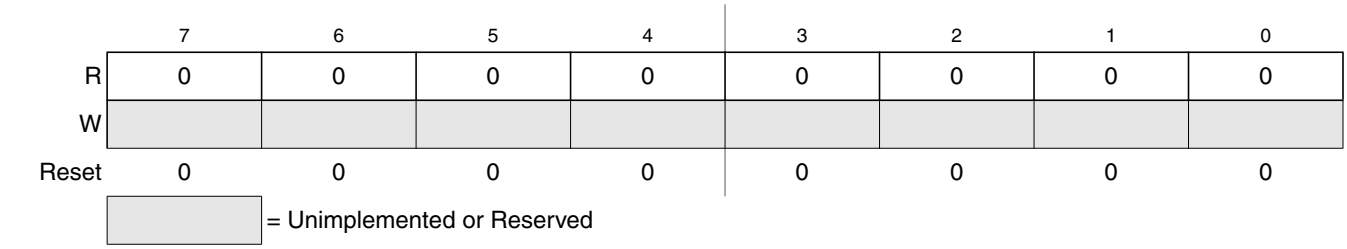

Figure 21-6. RESERVED1

All bits read 0 and are not writable.

In Table A-21 the timing characteristics for master mode are listed.

Table A-21. SPI Master Mode Timing Characteristics

Num	C	Characteristic	Symbol	Min	Typ	Max	Unit
1	P	SCK Frequency	f_{sck}	1/2048	—	1/2	f_{bus}
1	P	SCK Period	t_{sck}	2	—	2048	t_{bus}
2	D	Enable Lead Time	t_{lead}	—	1/2	—	t_{sck}
3	D	Enable Lag Time	t_{lag}	—	1/2	—	t_{sck}
4	D	Clock (SCK) High or Low Time	t_{wsck}	—	1/2	—	t_{sck}
5	D	Data Setup Time (Inputs)	t_{su}	8	—	—	ns
6	D	Data Hold Time (Inputs)	t_{hi}	8	—	—	ns
9	D	Data Valid after SCK Edge	t_{vsck}	—	—	30	ns
10	D	Data Valid after $\overline{SS}$ fall (CPHA=0)	t_{vss}	—	—	15	ns
11	D	Data Hold Time (Outputs)	t_{ho}	20	—	—	ns
12	D	Rise and Fall Time Inputs	t_{rfi}	—	—	8	ns
13	D	Rise and Fall Time Outputs	t_{rfo}	—	—	8	ns

A.6.2 Slave Mode

In Figure A-8 the timing diagram for slave mode with transmission format CPHA=0 is depicted.

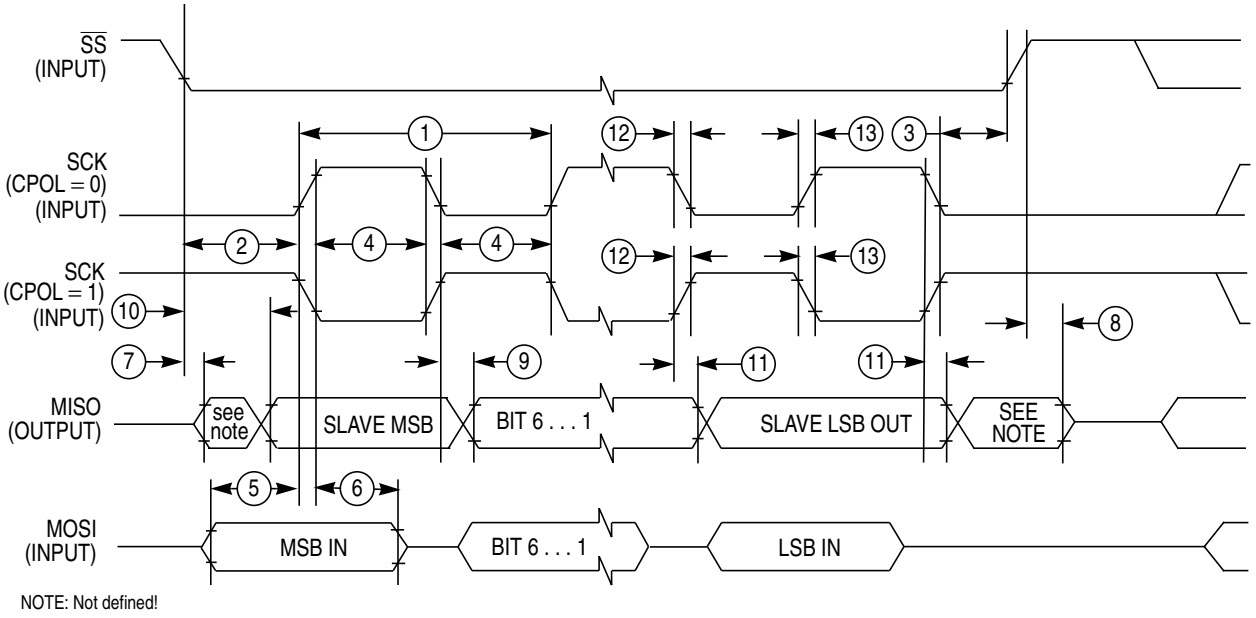

Figure A-8. SPI Slave Timing (CPHA=0)

Part Number	Mask[1] set	Temp.	Package	Speed	Die Type	Flash	RAM	I/O[2], [3]
MC9S12C64CFU	XL09S/0M66G	-40˚C, 85˚C	80QFP	25MHz	C128 die	64K	4K	60
MC9S12C64VFA	XL09S/0M66G	-40˚C,105˚C	48LQFP	25MHz	C128 die	64K	4K	31
MC9S12C64VPB	XL09S/0M66G	-40˚C,105˚C	52LQFP	25MHz	C128 die	64K	4K	35
MC9S12C64VFU	XL09S/0M66G	-40˚C, 105˚C	80QFP	25MHz	C128 die	64K	4K	60
MC9S12C64MFA	XL09S/0M66G	-40˚C,125˚C	48LQFP	25MHz	C128 die	64K	4K	31
MC9S12C64MPB	XL09S/0M66G	-40˚C,125˚C	52LQFP	25MHz	C128 die	64K	4K	35
MC9S12C64MFU	XL09S/0M66G	-40˚C, 125˚C	80QFP	25MHz	C128 die	64K	4K	60
MC9S12C32CFA16	xL45J / xM34C	-40˚C, 85˚C	48LQFP	16MHz	C32 die	32K	2K	31
MC9S12C32CPB16	xL45J / xM34C	-40˚C, 85˚C	52LQFP	16MHz	C32 die	32K	2K	35
MC9S12C32CFU16	xL45J / xM34C	-40˚C, 85˚C	80QFP	16MHz	C32 die	32K	2K	60
MC9S12C32VFA16	xL45J / xM34C	-40˚C,105˚C	48LQFP	16MHz	C32 die	32K	2K	31
MC9S12C32VPB16	xL45J / xM34C	-40˚C,105˚C	52LQFP	16MHz	C32 die	32K	2K	35
MC9S12C32VFU16	xL45J / xM34C	-40˚C, 105˚C	80QFP	16MHz	C32 die	32K	2K	60
MC9S12C32MFA16	xL45J / xM34C	-40˚C,125˚C	48LQFP	16MHz	C32 die	32K	2K	31
MC9S12C32MPB16	xL45J / xM34C	-40˚C,125˚C	52LQFP	16MHz	C32 die	32K	2K	35
MC9S12C32MFU16	xL45J / xM34C	-40˚C, 125˚C	80QFP	16MHz	C32 die	32K	2K	60
MC9S12C32CFA25	xL45J / xM34C	-40˚C, 85˚C	48LQFP	25MHz	C32 die	32K	2K	31
MC9S12C32CPB25	xL45J / xM34C	-40˚C, 85˚C	52LQFP	25MHz	C32 die	32K	2K	35
MC9S12C32CFU25	xL45J / xM34C	-40˚C, 85˚C	80QFP	25MHz	C32 die	32K	2K	60
MC9S12C32VFA25	xL45J / xM34C	-40˚C,105˚C	48LQFP	25MHz	C32 die	32K	2K	31
MC9S12C32VPB25	xL45J / xM34C	-40˚C,105˚C	52LQFP	25MHz	C32 die	32K	2K	35
MC9S12C32VFU25	xL45J / xM34C	-40˚C, 105˚C	80QFP	25MHz	C32 die	32K	2K	60
MC9S12C32MFA25	xL45J / xM34C	-40˚C,125˚C	48LQFP	25MHz	C32 die	32K	2K	31
MC9S12C32MPB25	xL45J / xM34C	-40˚C,125˚C	52LQFP	25MHz	C32 die	32K	2K	35
MC9S12C32MFU25	xL45J / xM34C	-40˚C, 125˚C	80QFP	25MHz	C32 die	32K	2K	60

1. XL09S denotes all minor revisions of L09S maskset
   XL45J denotes all minor revisions of L45J maskset
   Maskset dependent errata can be accessed at
   http://e-www.motorola.com/wbapp/sps/site/prod_summary.jsp
2. All C-Family derivatives feature 1 CAN, 1 SCI, 1 SPI, an 8-channel A/D, a 6-channel PWM and an 8 channel timer. The GC-Family members do not have the CAN module
3. I/O is the sum of ports able to act as digital input or output.

Table E-2. MC9S12GC-Family Part Number Coding

Part Number	Mask[1] set	Temp.	Package	Speed	Die Type	Flash	RAM	I/O[2], [3]
MC9S12GC32CFA	xL45J / xM34C	-40˚C, 85˚C	48LQFP	25MHz	C32 die	32K	2K	31
MC9S12GC32CPB	xL45J / xM34C	-40˚C, 85˚C	52LQFP	25MHz	C32 die	32K	2K	35
MC9S12GC32CFU	xL45J / xM34C	-40˚C, 85˚C	80QFP	25MHz	C32 die	32K	2K	60
MC9S12GC32VFA	xL45J / xM34C	-40˚C,105˚C	48LQFP	25MHz	C32 die	32K	2K	31
MC9S12GC32VPB	xL45J / xM34C	-40˚C,105˚C	52LQFP	25MHz	C32 die	32K	2K	35
MC9S12GC32VFU	xL45J / xM34C	-40˚C, 105˚C	80QFP	25MHz	C32 die	32K	2K	60
MC9S12GC32MFA	xL45J / xM34C	-40˚C,125˚C	48LQFP	25MHz	C32 die	32K	2K	31
MC9S12GC32MPB	xL45J / xM34C	-40˚C,125˚C	52LQFP	25MHz	C32 die	32K	2K	35