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	CANbus, EBI/EMI, I²C, SCI, SPI
Peripherals	LCD, Motor control PWM, POR, PWM, WDT
Number of I/O	85
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12hz256cal

Chapter 18
Background Debug Module (BDMV4)

Chapter 19
Debug Module (DBGV1)

Appendix B
PCB Layout Guidelines

Appendix C
Package Information

Appendix D
Derivative Differences

Appendix E
ROM Description

Appendix F
Ordering Information

Appendix G
Detailed Register Map

Figure 1-3 shows the device memory map for the MC9(3)S12HZ128 out of reset.

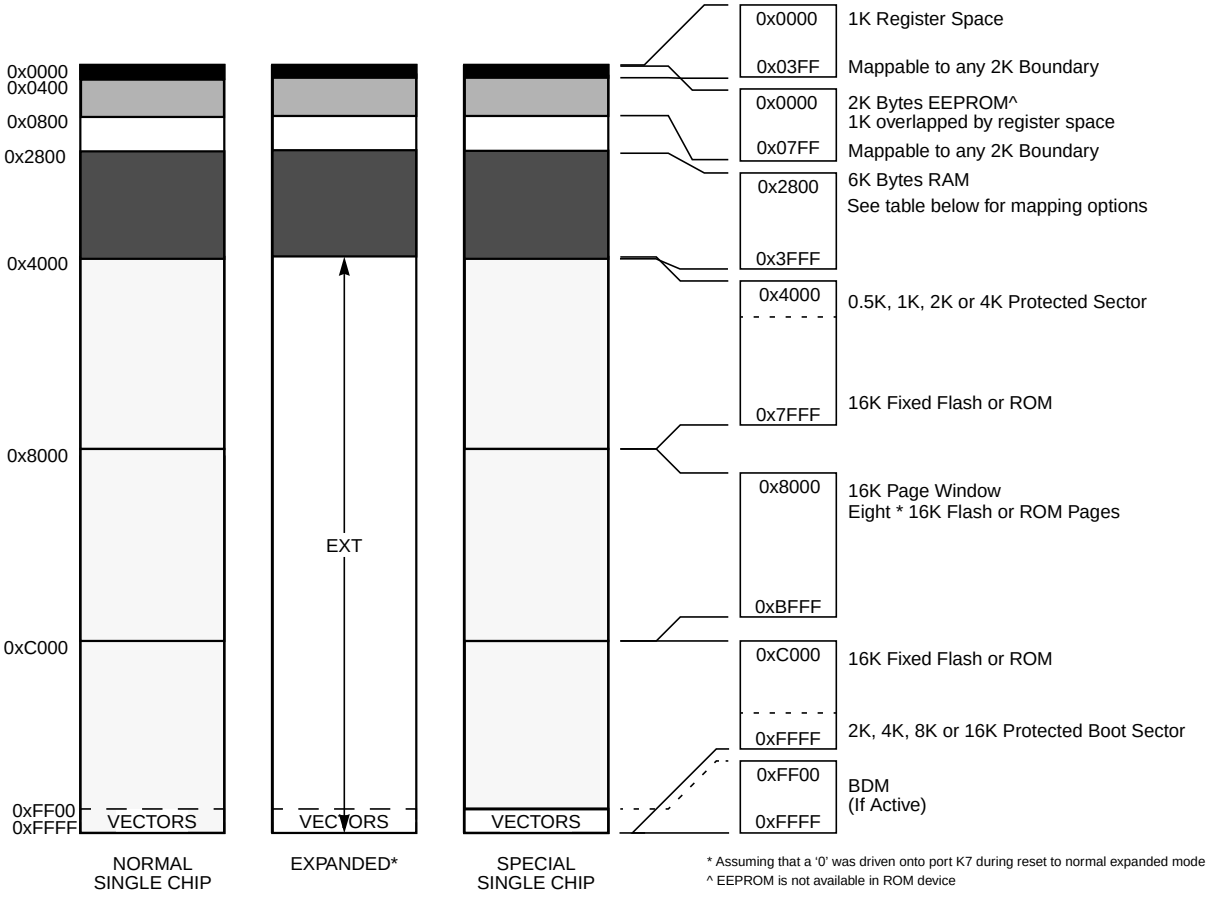

Figure 1-3. MC9(3)S12HZ128 Memory Map

Table 1-3. MC9(3)S12HZ128 RAM mapping options

INITRM[1]	RAM location	Reserved location (no Flash/ROM/EEPROM access)
0x00	0x0000 - 0x17FF	0x1800 - 0x2FFF
0x39	0x2800 - 0x3FFF	0x1000 - 0x27FF
0x40	0x4000 - 0x57FF	0x5800 - 0x6FFF
0x79	0x6800 - 0x7FFF	0x5000 - 0x67FF
0x80	0x8000 - 0x97FF	0x9800 - 0xAFFF
0xB9	0xA800 - 0xBFFF	0x9000 - 0xA7FF
0xC0	0xC000 - 0xD7FF	0xD800 - 0xEFFF
0xF9	0xE800 - 0xFFFF	0xD000 - 0xE7FF

1 User must initialize RAM13 bit to the same value as RAMHAL bit

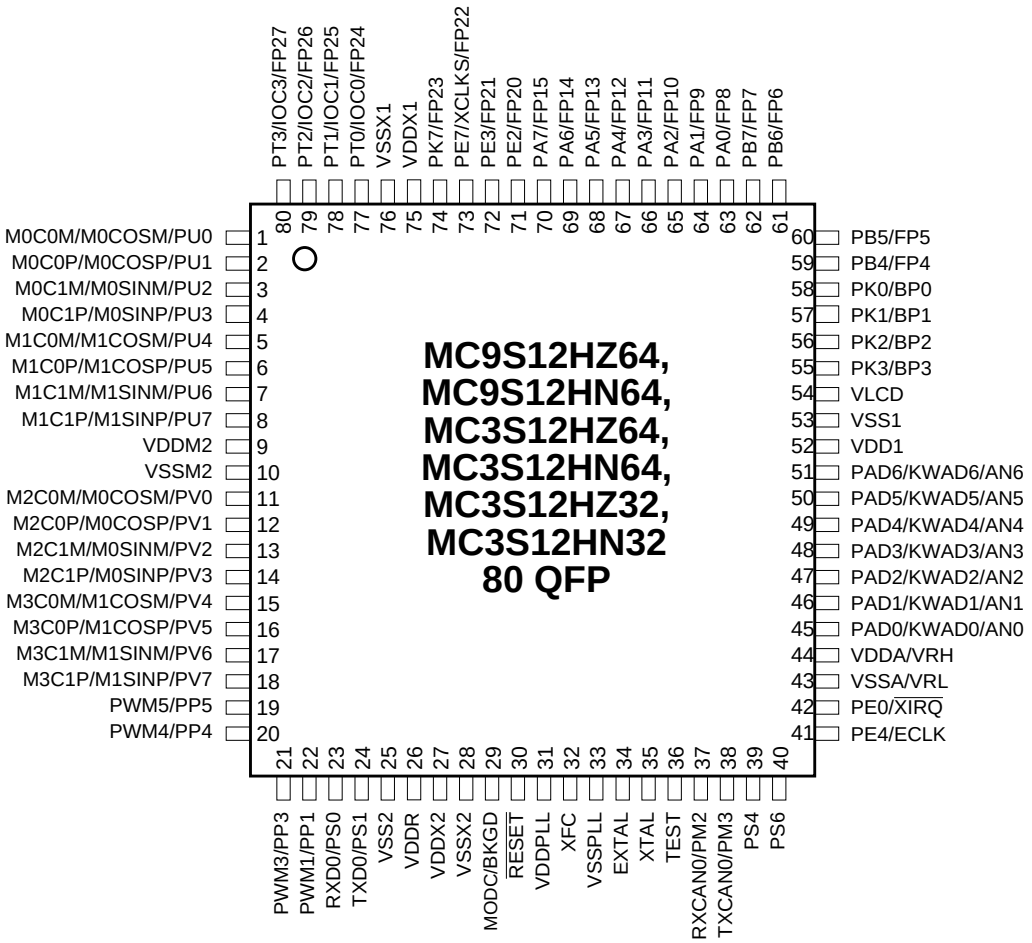

Figure 1-8. 80-Pin QFP for MC9S12HZ(N)64, MC3S12HZ(N)64 and MC3S12HZ(N)32

1.5.3 TEST — Test Pin

This pin is reserved for test.

NOTE

The TEST pin must be tied to V_{SS} in all applications.

1.5.4 XFC — PLL Loop Filter Pin

Dedicated pin used to create the PLL loop filter. Please ask your Freescale representative for the interactive application note to compute PLL loop filter elements. Any current leakage on this pin must be avoided.

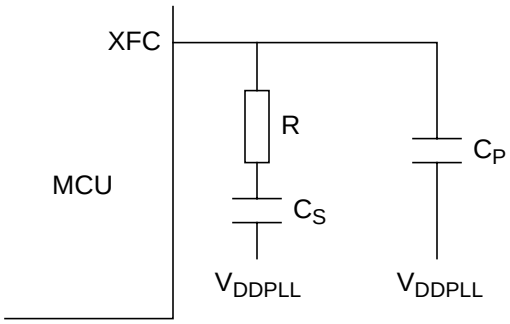

Figure 1-9. PLL Loop Filter Connections

1.5.5 BKGD / $\overline{\text{TAGHI}}$ / MODC — Background Debug, Tag High, and Mode Pin

The BKGD/$\overline{\text{TAGHI}}$/MODC pin is used as a pseudo-open-drain pin for the background debug communication. In MCU expanded modes of operation when instruction tagging is on, an input low on this pin during the falling edge of E-clock tags the high half of the instruction word being read into the instruction queue. It is used as a MCU operating mode select pin during reset. The state of this pin is latched to the MODC bit at the rising edge of $\overline{\text{RESET}}$.

1.5.6 Port Pins

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

PAD7–PAD0 are general-purpose input or output pins and analog inputs for the analog-to-digital converter. They can be configured to generate an interrupt causing the MCU to exit STOP or WAIT mode.

1.5.6.2 PA[7:0] / FP[15:8] / ADDR[15:8] / DATA[15:8] — Port A I/O Pins

PA7–PA0 are general-purpose input or output pins. They can be configured as frontplane segment driver outputs FP15–FP8 of the LCD. In MCU expanded modes of operation, these pins are used for the multiplexed external address and data bus.

1.5.6.36 PV[7:4] / M3C1(SIN)P, M3C1(SIN)M, M3C0(COS)P, M3C0(COS)M — Port V I/O Pins [7:4]

PV7–PV4 are general-purpose input or output pins. They can be configured as high current PWM output pins which can be used for motor drive or to measure the back EMF to calibrate the pointer reset position. These pins interface to the coils of motor 3.

1.5.6.37 PV[3:0] / M2C1(SIN)P, M2C1(SIN)M, M2C0(COS)P, M2C0(COS)M — Port V I/O Pins [3:0]

PV3–PV0 are general-purpose input or output pins. They can be configured as high current PWM output pins which can be used for motor drive or to measure the back EMF to calibrate the pointer reset position. These pins interface to the coils of motor 2.

1.5.7 Power Supply Pins

MC9S12HZ256 power and ground pins are described below.

NOTE

All V_{SS} pins must be connected together in the application (See Appendix B, “PCB Layout Guidelines”).

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 how heavily the MCU pins are loaded (Table B-1).

1.5.7.1 V_{DDR} — External Power Pin

V_{DDR} is the power supply pin for the internal voltage regulator.

1.5.7.2 V_{DDX1}, V_{DDX2}, V_{SSX1}, V_{SSX2} — External Power and Ground Pins

V_{DDX1}, V_{DDX2}, V_{SSX1} and V_{SSX2} are the power supply and ground pins for input/output drivers.V_{DDX1} and V_{DDX2} as well as V_{SSX1} and V_{SSX2} are not internally connected.

1.5.7.3 V_{DD1}, V_{SS1}, V_{SS2} — Internal Logic Power Pins

V_{DD1}, V_{SS1} and V_{SS2} are the internal logic power and ground pins and related to the voltage regulator output. These pins serve as connection points for filter capacitors. V_{SS1} and V_{SS2} are internally connected.

NOTE

No load allowed except for bypass capacitors.

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

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

4.3.3.3 Port M Data Direction Register (DDRM)

	7	6	5	4	3	2	1	0
R	0	0	DDRM5	DDRM4	DDRM3	DDRM2	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Reserved or Unimplemented

Figure 4-18. Port M Data Direction Register (DDRM)

Read: Anytime. Write: Anytime.

This register configures port pins PM[5:2] as either input or output.

When a CAN module is enabled, the corresponding transmitter (TXCANx) pin becomes an output, the corresponding receiver (RXCANx) pin becomes an input, and the associated Data Direction Register bits have no effect. If a CAN module is disabled, the corresponding Data Direction Register bit reverts to control the I/O direction of the associated pin.

Table 4-13. DDRM Field Descriptions

Field	Description
5:2 DDRM[5:2]	**Data Direction Port M** 0 Associated pin is configured as input. 1 Associated pin is configured as output.

4.3.3.4 Port M Reduced Drive Register (RDRM)

	7	6	5	4	3	2	1	0
R	0	0	RDRM5	RDRM4	RDRM3	RDRM2	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Reserved or Unimplemented

Figure 4-19. Port M Reduced Drive Register (RDRM)

Read: Anytime. Write: Anytime.

This register configures the drive strength of configured output pins as either full or reduced. If a pin is configured as input, the corresponding Reduced Drive Register bit has no effect.

Table 4-14. RDRM Field Descriptions

Field	Description
5:2 RDRM[5:2]	**Reduced Drive Port M** 0 Full drive strength at output 1 Associated pin drives at about 1/3 of the full drive strength.

4.3.4.4 Port P Reduced Drive Register (RDRP)

Figure 4-26. Port P Reduced Drive Register (RDRP)

Read:Anytime. Write:Anytime.

This register configures the drive strength of configured output pins as either full or reduced. If a pin is configured as input, the corresponding Reduced Drive Register bit has no effect.

Table 4-19. RDRP Field Descriptions

Field	Description
5:0 RDRP[5:0]	**Reduced Drive Port P** 0 Full drive strength at output. 1 Associated pin drives at about 1/3 of the full drive strength.

Table 5-7. RTI Frequency Divide Rates

RTR[3:0]	RTR[6:4] =							
	000 (OFF)	**001 (2^{10})**	**010 (2^{11})**	**011 (2^{12})**	**100 (2^{13})**	**101 (2^{14})**	**110 (2^{15})**	**111 (2^{16})**
0000 (÷1)	OFF*	2^{10}	2^{11}	2^{12}	2^{13}	2^{14}	2^{15}	2^{16}
0001 (÷2)	OFF*	$2x2^{10}$	$2x2^{11}$	$2x2^{12}$	$2x2^{13}$	$2x2^{14}$	$2x2^{15}$	$2x2^{16}$
0010 (÷3)	OFF*	$3x2^{10}$	$3x2^{11}$	$3x2^{12}$	$3x2^{13}$	$3x2^{14}$	$3x2^{15}$	$3x2^{16}$
0011 (÷4)	OFF*	$4x2^{10}$	$4x2^{11}$	$4x2^{12}$	$4x2^{13}$	$4x2^{14}$	$4x2^{15}$	$4x2^{16}$
0100 (÷5)	OFF*	$5x2^{10}$	$5x2^{11}$	$5x2^{12}$	$5x2^{13}$	$5x2^{14}$	$5x2^{15}$	$5x2^{16}$
0101 (÷6)	OFF*	$6x2^{10}$	$6x2^{11}$	$6x2^{12}$	$6x2^{13}$	$6x2^{14}$	$6x2^{15}$	$6x2^{16}$
0110 (÷7)	OFF*	$7x2^{10}$	$7x2^{11}$	$7x2^{12}$	$7x2^{13}$	$7x2^{14}$	$7x2^{15}$	$7x2^{16}$
0111 (÷8)	OFF*	$8x2^{10}$	$8x2^{11}$	$8x2^{12}$	$8x2^{13}$	$8x2^{14}$	$8x2^{15}$	$8x2^{16}$
1000 (÷9)	OFF*	$9x2^{10}$	$9x2^{11}$	$9x2^{12}$	$9x2^{13}$	$9x2^{14}$	$9x2^{15}$	$9x2^{16}$
1001 (÷10)	OFF*	$10x2^{10}$	$10x2^{11}$	$10x2^{12}$	$10x2^{13}$	$10x2^{14}$	$10x2^{15}$	$10x2^{16}$
1010 (÷11)	OFF*	$11x2^{10}$	$11x2^{11}$	$11x2^{12}$	$11x2^{13}$	$11x2^{14}$	$11x2^{15}$	$11x2^{16}$
1011 (÷12)	OFF*	$12x2^{10}$	$12x2^{11}$	$12x2^{12}$	$12x2^{13}$	$12x2^{14}$	$12x2^{15}$	$12x2^{16}$
1100 (÷ 13)	OFF*	$13x2^{10}$	$13x2^{11}$	$13x2^{12}$	$13x2^{13}$	$13x2^{14}$	$13x2^{15}$	$13x2^{16}$
1101 (÷14)	OFF*	$14x2^{10}$	$14x2^{11}$	$14x2^{12}$	$14x2^{13}$	$14x2^{14}$	$14x2^{15}$	$14x2^{16}$
1110 (÷15)	OFF*	$15x2^{10}$	$15x2^{11}$	$15x2^{12}$	$15x2^{13}$	$15x2^{14}$	$15x2^{15}$	$15x2^{16}$
1111 (÷ 16)	OFF*	$16x2^{10}$	$16x2^{11}$	$16x2^{12}$	$16x2^{13}$	$16x2^{14}$	$16x2^{15}$	$16x2^{16}$

* Denotes the default value out of reset.This value should be used to disable the RTI to ensure future backwards compatibility.

The PLL filter can be manually or automatically configured into one of two possible operating modes:

- Acquisition mode

  In acquisition mode, the filter can make large frequency corrections to the VCO. This mode is used at PLL start-up or when the PLL has suffered a severe noise hit and the VCO frequency is far off the desired frequency. When in acquisition mode, the TRACK status bit is cleared in the CRGFLG register.
- Tracking mode

  In tracking mode, the filter makes only small corrections to the frequency of the VCO. PLL jitter is much lower in tracking mode, but the response to noise is also slower. The PLL enters tracking mode when the VCO frequency is nearly correct and the TRACK bit is set in the CRGFLG register.

The PLL can change the bandwidth or operational mode of the loop filter manually or automatically.

In automatic bandwidth control mode (AUTO = 1), the lock detector automatically switches between acquisition and tracking modes. Automatic bandwidth control mode also is used to determine when the PLL clock (PLLCLK) is safe to use as the source for the system and core clocks. If PLL LOCK interrupt requests are enabled, the software can wait for an interrupt request and then check the LOCK bit. If CPU interrupts are disabled, software can poll the LOCK bit continuously (during PLL start-up, usually) or at periodic intervals. In either case, only when the LOCK bit is set, is the PLLCLK clock safe to use as the source for the system and core clocks. If the PLL is selected as the source for the system and core clocks and the LOCK bit is clear, the PLL has suffered a severe noise hit and the software must take appropriate action, depending on the application.

The following conditions apply when the PLL is in automatic bandwidth control mode (AUTO = 1):

- The TRACK bit is a read-only indicator of the mode of the filter.
- The TRACK bit is set when the VCO frequency is within a certain tolerance, Δ_{trk}, and is clear when the VCO frequency is out of a certain tolerance, Δ_{unt}.
- The LOCK bit is a read-only indicator of the locked state of the PLL.
- The LOCK bit is set when the VCO frequency is within a certain tolerance, Δ_{Lock}, and is cleared when the VCO frequency is out of a certain tolerance, Δ_{unl}.
- CPU interrupts can occur if enabled (LOCKIE = 1) when the lock condition changes, toggling the LOCK bit.

The PLL can also operate in manual mode (AUTO = 0). Manual mode is used by systems that do not require an indicator of the lock condition for proper operation. Such systems typically operate well below the maximum system frequency (f_{sys}) and require fast start-up. The following conditions apply when in manual mode:

- ACQ is a writable control bit that controls the mode of the filter. Before turning on the PLL in manual mode, the ACQ bit should be asserted to configure the filter in acquisition mode.
- After turning on the PLL by setting the PLLON bit software must wait a given time (t_{acq}) before entering tracking mode (ACQ = 0).
- After entering tracking mode software must wait a given time (t_{al}) before selecting the PLLCLK as the source for system and core clocks (PLLSEL = 1).

If the MCU gets an external reset during pseudo-stop mode active, the CRG asynchronously restores all configuration bits in the register space to its default settings and starts the reset generator. After completing the reset sequence processing begins by fetching the normal reset vector. Pseudo-stop mode is exited and the MCU is in run mode again.

If the clock monitor is enabled (CME = 1) the MCU is able to leave pseudo-stop mode when loss of oscillator/external clock is detected by a clock monitor fail. If the SCME bit is not asserted the CRG generates a clock monitor fail reset (CMRESET). The CRG's behavior for CMRESET is the same compared to external reset, but another reset vector is fetched after completion of the reset sequence. If the SCME bit is asserted the CRG generates a SCM interrupt if enabled (SCMIE=1). After generating the interrupt the CRG enters self-clock mode and starts the clock quality checker (see Section 5.4.4, "Clock Quality Checker"). Then the MCU continues with normal operation. If the SCM interrupt is blocked by SCMIE = 0, the SCMIF flag will be asserted but the CRG will not wake-up from pseudo-stop mode.

If any other interrupt source (e.g. RTI) triggers exit from pseudo-stop mode the MCU immediately continues with normal operation. Because the PLL has been powered-down during stop mode the PLLSEL bit is cleared and the MCU runs on OSCCLK after leaving stop mode. The software must set the PLLSEL bit again, in order to switch system and core clocks to the PLLCLK.

Table 5-12 summarizes the outcome of a clock loss while in pseudo-stop mode.

7.3.2.8 Reserved Register 0 (ATDTEST0)

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

= Unimplemented or Reserved u = Unaffected

Figure 7-10. Reserved Register 0 (ATDTEST0)

Read: Anytime, returns unpredictable values

Write: Anytime in special modes, unimplemented in normal modes

NOTE

Writing to this register when in special modes can alter functionality.

7.3.2.9 ATD Test Register 1 (ATDTEST1)

This register contains the SC bit used to enable special channel conversions.

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

= Unimplemented or Reserved u = Unaffected

Figure 7-11. Reserved Register 1 (ATDTEST1)

Read: Anytime, returns unpredictable values for bit 7 and bit 6

Write: Anytime

NOTE

Writing to this register when in special modes can alter functionality.

Table 7-19. ATDTEST1 Field Descriptions

Field	Description
0 SC	**Special Channel Conversion Bit** — If this bit is set, then special channel conversion can be selected using CC, CB, and CA of ATDCTL5. Table 7-20 lists the coding. 0 Special channel conversions disabled 1 Special channel conversions enabled

Table 8-8. LCD Duty and Bias

Duty	LCDCR0 Register		Backplanes				Bias (BIAS = 0)			Bias (BIAS = 1)		
	DUTY1	DUTY0	BP3	BP2	BP1	BP0	1/1	1/2	1/3	1/1	1/2	1/3
1/1	0	1	OFF	OFF	OFF	BP0	YES	NA	NA	YES	NA	NA
1/2	1	0	OFF	OFF	BP1	BP0	NA	YES	NA	NA	NA	YES
1/3	1	1	OFF	BP2	BP1	BP0	NA	NA	YES	NA	NA	YES
1/4	0	0	BP3	BP2	BP1	BP0	NA	NA	YES	NA	NA	YES

8.4.2 Operation in Wait Mode

The LCD32F4B driver system operation during wait mode is controlled by the LCD stop in wait (LCDSWAI) bit in the LCD control register 1 (LCDCR1). If LCDSWAI is reset, the LCD32F4B driver system continues to operate during wait mode. If LCDSWAI is set, the LCD32F4B driver system is turned off during wait mode. In this case, the LCD waveform generation clocks are stopped and the LCD32F4B drivers pull down to VSSX those frontplane and backplane pins that were enabled before entering wait mode. The contents of the LCD RAM and the LCD registers retain the values they had prior to entering wait mode.

8.4.3 Operation in Pseudo Stop Mode

The LCD32F4B driver system operation during pseudo stop mode is controlled by the LCD run in pseudo stop (LCDRPSTP) bit in the LCD control register 1 (LCDCR1). If LCDRPSTP is reset, the LCD32F4B driver system is turned off during pseudo stop mode. In this case, the LCD waveform generation clocks are stopped and the LCD32F4B drivers pull down to VSSX those frontplane and backplane pins that were enabled before entering pseudo stop mode. If LCDRPSTP is set, the LCD32F4B driver system continues to operate during pseudo stop mode. The contents of the LCD RAM and the LCD registers retain the values they had prior to entering pseudo stop mode.

8.4.4 Operation in Stop Mode

All LCD32F4B driver system clocks are stopped, the LCD32F4B driver system pulls down to VSSX those frontplane and backplane pins that were enabled before entering stop mode. Also, during stop mode, the contents of the LCD RAM and the LCD registers retain the values they had prior to entering stop mode. As a result, after exiting from stop mode, the LCD32F4B driver system clocks will run (if LCDEN = 1) and the frontplane and backplane pins retain the functionality they had prior to entering stop mode.

8.4.5 LCD Waveform Examples

Figure 8-9 through Figure 8-13 show the timing examples of the LCD output waveforms for the available modes of operation.

```
;-------------------------------------------------------------------------------
;global motor controller init
;-------------------------------------------------------------------------------
GLB_INIT: MOVB    #$0000,MCCTL0             ; fMC = fBUS, FAST=0, DITH=0
          MOVB    #$0000,MCCTL1             ; RECIRC=0, MCTOIE=0
          MOVW    #$D0D0,MCCC0              ; dual full h-bridge mode, left aligned,
                                            ; no channel delay
          MOVW    #$0000,MCPER_HI           ; disable motor controller
;-------------------------------------------------------------------------------
;motor controller startup
;-------------------------------------------------------------------------------
STARTUP:
          MOVW    #$0000,MCDC0_HI           ; define startup duty cycles
          MOVW    #$0000,MCDC1_HI
          MOVW    #MCPERIOD,MCPER_HI        ; define PWM period
          MOVB    #$80,TSCR1                ; enable timer
MAIN:     LDAA    PORTB                     ; if PB=0, activate shutdown
          ANDA    #$01
          BEQ     MN0
          JSR     TIM_SR
MN0:      TST     TFLG2                     ; poll for timer counter overflow flag
          BEQ     MAIN                      ; TOF set?
          JSR     TIM_SR                    ; yes, go to TIM_SR
          BRA     MAIN
TIM_SR:   LDX     TEMP_X                    ; restore index register X
          LDAA    PORTB                     ; if PB=0, enter shutdown routine
          ANDA    #$01
          BNE     SHUTDOWN
          LDX     TEMP_X                    ; restore index register X
          BEQ     NEW_SEQ                   ; all mc configurations done?
NEW_CFG:  LDD     DTYDAT,X                  ; load new config's
          STD     MCDC0_HI
          DEX
          DEX
          LDD     DTYDAT,X
          STD     MCDC1_HI
          BRA     END_SR                    ; leave sub-routine
SHUTDOWN: MOVB    #$00,TSCR1                ; disable timer
          MOVW    #$0000,MCDC0_HI           ; define startup duty cycle
          MOVW    #$0000,MCDC1_HI           ; define startup duty cycle
          LDAA    #$0000                    ; ensure that duty cycle registers are
                                            ; cleared for some time before disabling
                                            ; the motor controller
LOOP      DECA
          BNE     LOOP
          MOVW    #$0000,MCPER_HI           ; define pwm period
NEW_SEQ:  MOVW    TABLESIZE,TEMP_X          ; start new tx loop
          LDX     TEMP_X
END_SR:   STX     TEMP_X                    ; save byte counter
          MOVB    #$80,TFLG2                ; clear TOF
          RTS                               ; wait for new timer overflow
```

12.2.2 TXCAN — CAN Transmitter Output Pin

TXCAN is the MSCAN transmitter output pin. The TXCAN output pin represents the logic level on the CAN bus:

0 = Dominant state
1 = Recessive state

12.2.3 CAN System

A typical CAN system with MSCAN is shown in Figure 12-2. Each CAN station is connected physically to the CAN bus lines through a transceiver device. The transceiver is capable of driving the large current needed for the CAN bus and has current protection against defective CAN or defective stations.

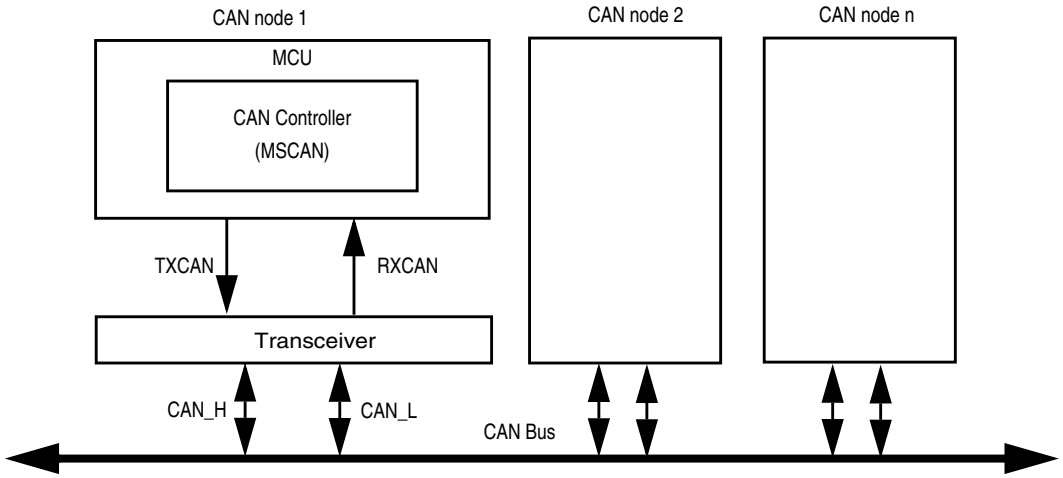

Figure 12-2. CAN System

12.3 Memory Map and Register Definition

This section provides a detailed description of all registers accessible in the MSCAN.

12.3.1 Module Memory Map

Table 12-1 gives an overview on all registers and their individual bits in the MSCAN memory map. The *register address* results from the addition of *base address* and *address offset*. The *base address* is determined at the MCU level and can be found in the Memory block description chapter. The *address offset* is defined at the module level.

The MSCAN occupies 64 bytes in the memory space. The base address of the MSCAN module is determined at the MCU level when the MCU is defined. The register decode map is fixed and begins at the first address of the module address offset.

Table 12-1 shows the individual registers associated with the MSCAN and their relative offset from the base address. The detailed register descriptions follow in the order they appear in the register map.

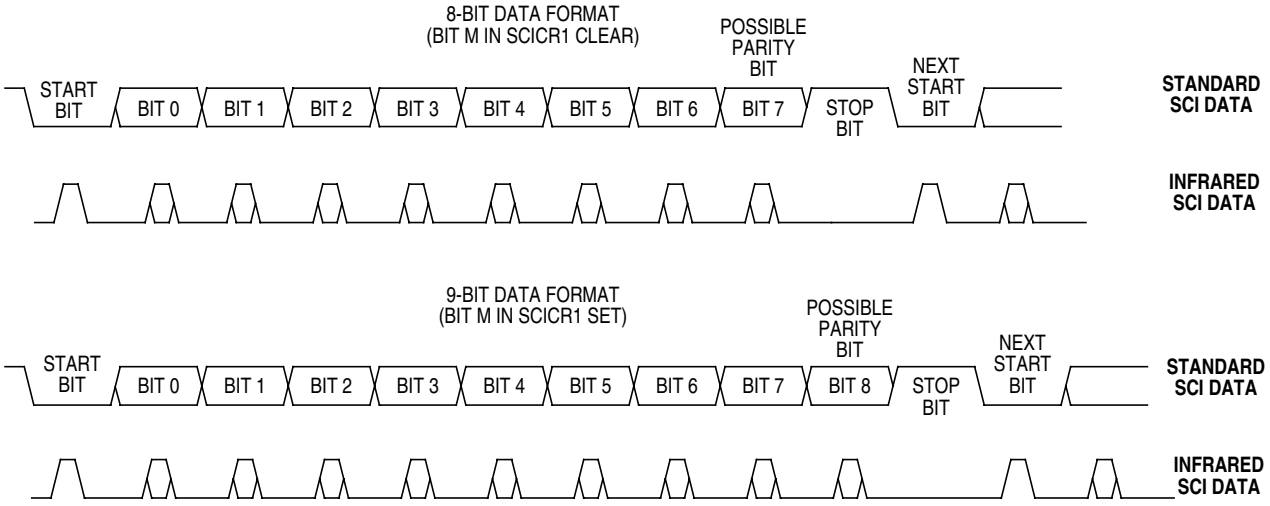

Figure 13-12. SCI Data Formats

Each data character is contained in a frame that includes a start bit, eight or nine data bits, and a stop bit. Clearing the M bit in SCI control register 1 configures the SCI for 8-bit data characters. A frame with eight data bits has a total of 10 bits. Setting the M bit configures the SCI for nine-bit data characters. A frame with nine data bits has a total of 11 bits

Table 13-11. Example of 8-bit Data Formats

Start Bit	Data Bits	Address Bits	Parity Bits	Stop Bit
1	8	0	0	1
1	7	0	1	1
1	7	1[1]	0	1

1 The address bit identifies the frame as an address character. See Section 13.4.5.6, “Receiver Wakeup”.

When the SCI is configured for 9-bit data characters, the ninth data bit is the T8 bit in SCI data register high (SCIDRH). It remains unchanged after transmission and can be used repeatedly without rewriting it. A frame with nine data bits has a total of 11 bits.

Table 13-12. Example of 9-Bit Data Formats

Start Bit	Data Bits	Address Bits	Parity Bits	Stop Bit
1	9	0	0	1
1	8	0	1	1
1	8	1[1]	0	1

1 The address bit identifies the frame as an address character. See Section 13.4.5.6, “Receiver Wakeup”.

Table 16-14. Timer Clock Selection

PR2	PR1	PR0	Timer Clock
0	0	0	Bus Clock / 1
0	0	1	Bus Clock / 2
0	1	0	Bus Clock / 4
0	1	1	Bus Clock / 8
1	0	0	Bus Clock / 16
1	0	1	Bus Clock / 32
1	1	0	Bus Clock / 64
1	1	1	Bus Clock / 128

NOTE

The newly selected prescale factor will not take effect until the next synchronized edge where all prescale counter stages equal zero.

16.3.2.12 Main Timer Interrupt Flag 1 (TFLG1)

	7	6	5	4	3	2	1	0
R W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
Reset	0	0	0	0	0	0	0	0

Figure 16-20. Main Timer Interrupt Flag 1 (TFLG1)

Read: Anytime

Write: Used in the clearing mechanism (set bits cause corresponding bits to be cleared). Writing a zero will not affect current status of the bit.

Table 16-15. TRLG1 Field Descriptions

Field	Description
7:0 C[7:0]F	**Input Capture/Output Compare Channel “x” Flag** — These flags are set when an input capture or output compare event occurs. Clear a channel flag by writing one to it. When TFFCA bit in TSCR register is set, a read from an input capture or a write into an output compare channel (0x0010–0x001F) will cause the corresponding channel flag CxF to be cleared.

18.3.2.2 BDM CCR Holding Register (BDMCCR)

	7	6	5	4	3	2	1	0
R W	CCR7	CCR6	CCR5	CCR4	CCR3	CCR2	CCR1	CCR0
Reset	0	0	0	0	0	0	0	0

Figure 18-4. BDM CCR Holding Register (BDMCCR)

Read: All modes

Write: All modes

NOTE

When BDM is made active, the CPU stores the value of the CCR register in the BDMCCR register. However, out of special single-chip reset, the BDMCCR is set to 0xD8 and not 0xD0 which is the reset value of the CCR register.

When entering background debug mode, the BDM CCR holding register is used to save the contents of the condition code register of the user's program. It is also used for temporary storage in the standard BDM firmware mode. The BDM CCR holding register can be written to modify the CCR value.

18.3.2.3 BDM Internal Register Position Register (BDMINR)

	7	6	5	4	3	2	1	0
R	0	REG14	REG13	REG12	REG11	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 18-5. BDM Internal Register Position (BDMINR)

Read: All modes

Write: Never

Table 18-4. BDMINR Field Descriptions

Field	Description
6:3 REG[14:11]	**Internal Register Map Position** — These four bits show the state of the upper five bits of the base address for the system's relocatable register block. BDMINR is a shadow of the INITRG register which maps the register block to any 2K byte space within the first 32K bytes of the 64K byte address space.

This register is not in the on-chip memory map in expanded and special peripheral modes. Therefore, these accesses will be echoed externally.

NOTE

These bits have no effect when the associated pin(s) are outputs. (The pull resistors are inactive.)

Table 21-9. PUCR Field Descriptions

Field	Description
7 PUPKE	**Pull resistors Port K Enable** 0 Port K pull resistors are disabled. 1 Enable pull resistors for port K input pins.
4 PUPEE	**Pull resistors Port E Enable** 0 Port E pull resistors on bits 7, 4:0 are disabled. 1 Enable pull resistors for port E input pins bits 7, 4:0. **Note:** Pins 5 and 6 of port E have pull resistors which are only enabled during reset. This bit has no effect on these pins.
1 PUPBE	**Pull resistors Port B Enable** 0 Port B pull resistors are disabled. 1 Enable pull resistors for all port B input pins.
0 PUPAE	**Pull resistors Port A Enable** 0 Port A pull resistors are disabled. 1 Enable pull resistors for all port A input pins.

21.3.2.11 Reduced Drive Register (RDRIV)

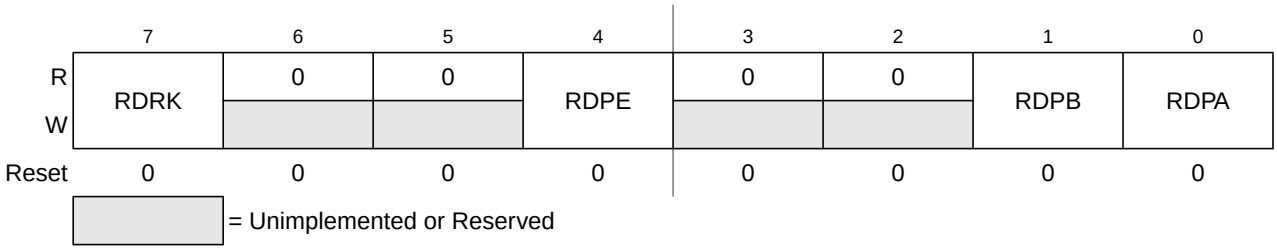

Figure 21-15. Reduced Drive Register (RDRIV)

Read: Anytime (provided this register is in the map)

Write: Anytime (provided this register is in the map)

This register is used to select reduced drive for the pins associated with the core ports. This gives reduced power consumption and reduced RFI with a slight increase in transition time (depending on loading). This feature would be used on ports which have a light loading. The reduced drive function is independent of which function is being used on a particular port.

This register is not in the on-chip memory map in expanded and special peripheral modes. Therefore, these accesses will be echoed externally.