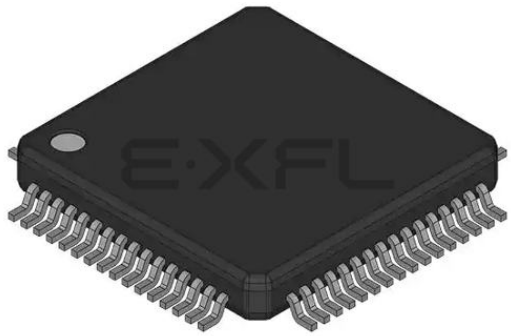


Welcome to E-XFL.COM

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	HC08
Core Size	8-Bit
Speed	8.4MHz
Connectivity	CANbus, SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	52
Program Memory Size	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 15x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-QFP
Supplier Device Package	64-QFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908as60acfu

Chapter 11
Configuration Register (CONFIG-1)

Chapter 12
Configuration Register (CONFIG-2)

Chapter 13
Break Module (BRK)

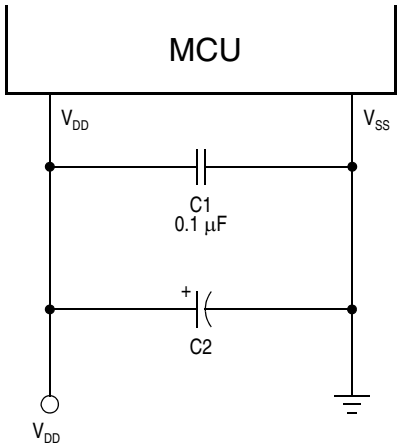
NOTE

The following pin descriptions are just a quick reference. For a more detailed representation, see Chapter 22 Input/Output Ports.

1.4.1 Power Supply Pins (V_{DD} and V_{SS})

V_{DD} and V_{SS} are the power supply and ground pins. The MCU operates from a single power supply.

Fast signal transitions on MCU pins place high, short-duration current demands on the power supply. To prevent noise problems, take special care to provide power supply bypassing at the MCU as shown in Figure 1-6. Place the C1 bypass capacitor as close to the MCU as possible. Use a high-frequency response ceramic capacitor for C1. C2 is an optional bulk current bypass capacitor for use in applications that require the port pins to source high current levels.



NOTE: Component values shown represent typical applications.

Figure 1-6. Power Supply Bypassing

V_{SS} is also the ground for the port output buffers and the ground return for the serial clock in the Serial Peripheral Interface module (SPI). See Chapter 19 Serial Peripheral Interface (SPI).

NOTE

V_{SS} must be grounded for proper MCU operation.

1.4.2 Oscillator Pins (OSC1 and OSC2)

The OSC1 and OSC2 pins are the connections for the on-chip oscillator circuit. See Chapter 10 Clock Generator Module (CGM).

1.4.3 External Reset Pin ($\overline{RST}$)

A 0 on the $\overline{RST}$ pin forces the MCU to a known startup state. $\overline{RST}$ is bidirectional, allowing a reset of the entire system. It is driven low when any internal reset source is asserted. See Chapter 9 System Integration Module (SIM) for more information.

1.4.4 External Interrupt Pin ($\overline{IRQ}$)

$\overline{IRQ}$ is an asynchronous external interrupt pin. See Chapter 17 External Interrupt Module (IRQ).

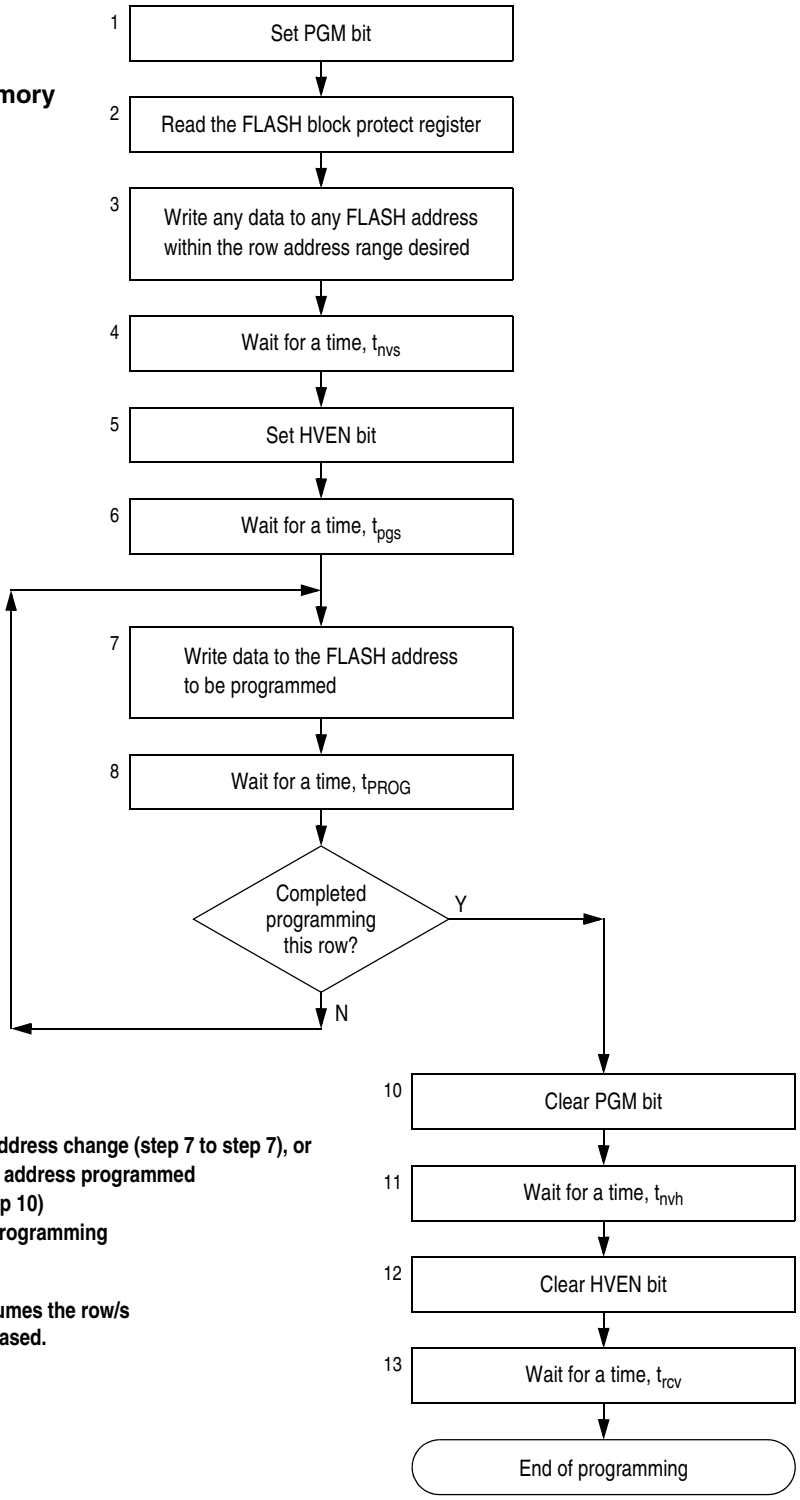
Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$0043	Timer B Modulo Register High (TBMODH)	Read: Write:	Bit 15	14	13	12	11	10	9	Bit 8
$0044	Timer B Modulo Register Low (TBMODL)	Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0
$0045	Timer B CH0 Status and Control Register (TBSC0)	Read: Write:	CH0F 0	CH0IE	MS0B	MS0A	ELS0B	ELS0A	TOV0	CH0MAX
$0046	Timer B CH0 Register High (TBCH0H)	Read: Write:	Bit 15	14	13	12	11	10	9	Bit 8
$0047	Timer B CH0 Register Low (TBCH0L)	Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0
$0048	Timer B CH1 Status and Control Register (TBSC1)	Read: Write:	CH1F 0	CH1IE	0 R	MS1A	ELS1B	ELS1A	TOV1	CH1MAX
$0049	Timer B CH1 Register High (TBCH1H)	Read: Write:	Bit 15	14	13	12	11	10	9	Bit 8
$004A	Timer B CH1 Register Low (TBCH1L)	Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0
$004B	PIT Status and Control Register (PSC)	Read: Write:	POF 0	POIE	PSTOP	0 PRST	0	PPS2	PPS1	PPS0
$004C	PIT Counter Register High (PCNTH)	Read: Write:	Bit 15	14	13	12	11	10	9	Bit 8
$004D	PIT Counter Register Low (PCNTL)	Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0
$004E	PIT Modulo Register High (PMODH)	Read: Write:	Bit 15	14	13	12	11	10	9	Bit 8
$004F	PIT Modulo Register Low (PMODL)	Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0

= Unimplemented R = Reserved

Figure 2-2. I/O Data, Status and Control Registers (Sheet 5 of 5)

All registers are shown for both MC68HC908AS60A and MC68HC908AZ60A. Refer to individual module chapters to determine if the module is available and the register active or not.

Algorithm for programming a row (64 bytes) of FLASH memory



NOTE:

The time between each FLASH address change (step 7 to step 7), or the time between the last FLASH address programmed to clearing PGM bit (step 7 to step 10) must not exceed the maximum programming time, t_{PROG} max.

This row program algorithm assumes the row/s to be programmed are initially erased.

Figure 5-4. FLASH Programming Algorithm Flowchart

6.4.5.2 EEPROM-1 Programming

The unprogrammed or erase state of an EEPROM bit is a logic 1. Programming changes the state to a logic 0. Only EEPROM bytes in the non-protected blocks and the EE1NVR register can be programmed.

Use the following procedure to program a byte of EEPROM:

1. Clear EERAS1 and EERAS0 and set EELAT in the EE1CR.(A)

NOTE

If using the AUTO mode, also set the AUTO bit during Step 1.

2. Write the desired data to the desired EEPROM address.(B)
3. Set the EEPGM bit.(C) Go to Step 7 if AUTO is set.
4. Wait for time, t_{EEPGM}, to program the byte.
5. Clear EEPGM bit.
6. Wait for time, t_{EEFPV}, for the programming voltage to fall. Go to Step 8.
7. Poll the EEPGM bit until it is cleared by the internal timer.(D)
8. Clear EELAT bits.(E)

NOTE

A. *EERAS1 and EERAS0 must be cleared for programming. Setting the EELAT bit configures the address and data buses to latch data for programming the array. Only data with a valid EEPROM-1 address will be latched. If EELAT is set, other writes to the EE1CR will be allowed after a valid EEPROM-1 write.*

B. *If more than one valid EEPROM write occurs, the last address and data will be latched overriding the previous address and data. Once data is written to the desired address, do not read EEPROM-1 locations other than the written location. (Reading an EEPROM location returns the latched data and causes the read address to be latched).*

C. *The EEPGM bit cannot be set if the EELAT bit is cleared or a non-valid EEPROM address is latched. This is to ensure proper programming sequence. Once EEPGM is set, do not read any EEPROM-1 locations; otherwise, the current program cycle will be unsuccessful. When EEPGM is set, the on-board programming sequence will be activated.*

D. *The delay time for the EEPGM bit to be cleared in AUTO mode is less than t_{EEPGM}. However, on other MCUs, this delay time may be different. For forward compatibility, software should not make any dependency on this delay time.*

E. *Any attempt to clear both EEPGM and EELAT bits with a single instruction will only clear EEPGM. This is to allow time for removal of high voltage from the EEPROM-1 array.*

9.6.2 Stop Mode

In stop mode, the SIM counter is reset and the system clocks are disabled. An interrupt request from a module can cause an exit from stop mode. Stacking for interrupts begins after the selected stop recovery time has elapsed. Reset also causes an exit from stop mode.

The SIM disables the clock generator module outputs (CGMOUT and CGMXCLK) in stop mode, stopping the CPU and peripherals. Stop recovery time is selectable using the SSREC bit in the configuration register (CONFIG-1). If SSREC is set, stop recovery is reduced from the normal delay of 4096 CGMXCLK cycles down to 32. This is ideal for applications using canned oscillators that do not require long startup times from stop mode.

NOTE

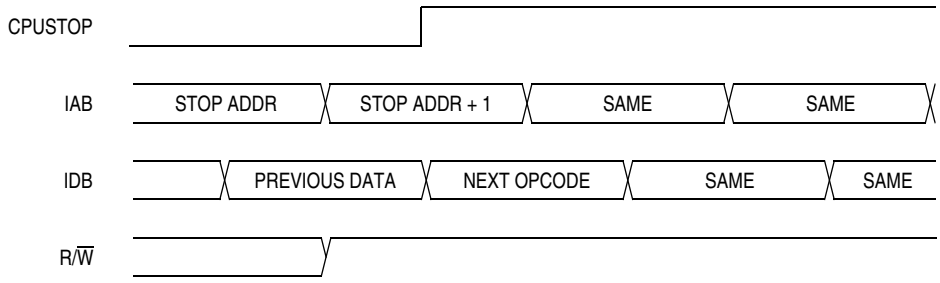
External crystal applications should use the full stop recovery time by clearing the SSREC bit.

The break module is inactive in Stop mode. The STOP instruction does not affect break module register states.

The SIM counter is held in reset from the execution of the STOP instruction until the beginning of stop recovery. It is then used to time the recovery period. Figure 9-15 shows stop mode entry timing.

NOTE

To minimize stop current, all pins configured as inputs should be driven to a logic 1 or logic 0.



NOTE: Previous data can be operand data or the STOP opcode, depending on the last instruction.

Figure 9-15. Stop Mode Entry Timing

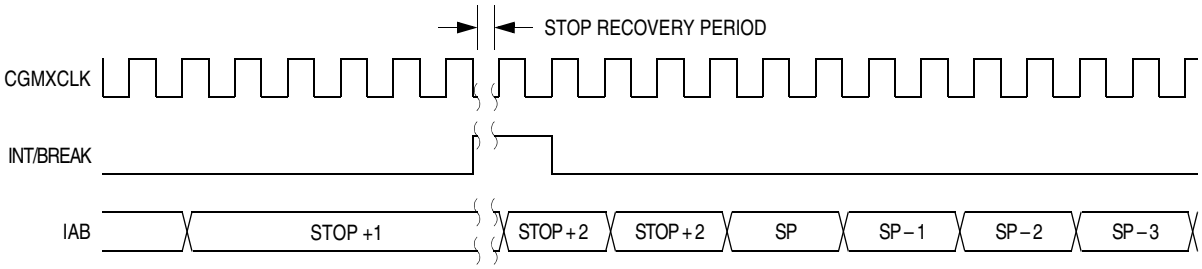


Figure 9-16. Stop Mode Recovery from Interrupt or Break

Chapter 14
Monitor ROM (MON)

14.1 Introduction

This chapter describes the monitor ROM (MON). The monitor ROM allows complete testing of the MCU through a single-wire interface with a host computer.

14.2 Features

Features of the monitor ROM include:

- Normal User-Mode Pin Functionality
- One Pin Dedicated to Serial Communication between Monitor ROM and Host Computer
- Standard Mark/Space Non-Return-to-Zero (NRZ) Communication with Host Computer
- Up to 28.8 kBaud Communication with Host Computer
- Execution of Code in RAM or FLASH
- FLASH Security
- FLASH Programming

14.3 Functional Description

Monitor ROM receives and executes commands from a host computer. Figure 14-1 shows a sample circuit used to enter monitor mode and communicate with a host computer via a standard RS-232 interface.

While simple monitor commands can access any memory address, the MC68HC908AS60A and MC68HC908AZ60A have a FLASH security feature to prevent external viewing of the contents of FLASH. Proper procedures must be followed to verify FLASH content. Access to the FLASH is denied to unauthorized users of customer specified software (see 14.3.8 Security).

In monitor mode, the MCU can execute host-computer code in RAM while all MCU pins except PTA0 retain normal operating mode functions. All communication between the host computer and the MCU is through the PTA0 pin. A level-shifting and multiplexing interface is required between PTA0 and the host computer. PTA0 is used in a wired-OR configuration and requires a pullup resistor.

Table 14-5. IREAD (Indexed Read) Command

Description	Read next 2 bytes in memory from last address accessed
Operand	Specifies 2-byte address in high byte:low byte order
Data Returned	Returns contents of next two addresses
Opcode	$1A
Command Sequence	

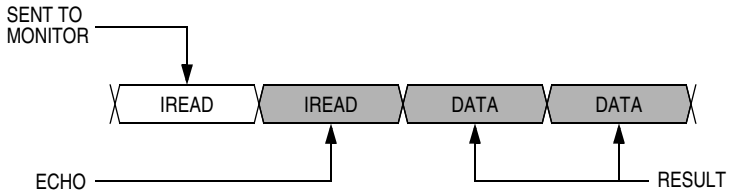


Table 14-6. IWRITE (Indexed Write) Command

Description	Write to last address accessed + 1
Operand	Specifies single data byte
Data Returned	None
Opcode	$19
Command Sequence	

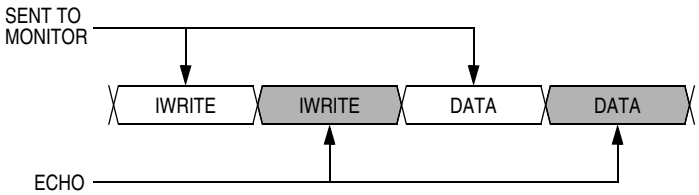
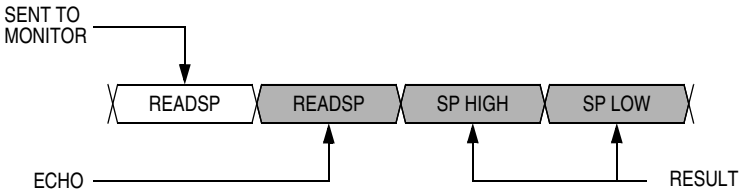


Table 14-7. READSP (Read Stack Pointer) Command

Description	Reads stack pointer
Operand	None
Data Returned	Returns stack pointer in high byte:low byte order
Opcode	$0C
Command Sequence	



18.7 I/O Signals

Port E shares two of its pins with the SCI module. The two SCI I/O pins are:

- PTE0/SCTxD — Transmit data
- PTE1/SCRxD — Receive data

18.7.1 PTE0/SCTxD (Transmit Data)

The PTE0/SCTxD pin is the serial data output from the SCI transmitter. The SCI shares the PTE0/SCTxD pin with port E. When the SCI is enabled, the PTE0/SCTxD pin is an output regardless of the state of the DDRE2 bit in data direction register E (DDRE).

18.7.2 PTE1/SCRxD (Receive Data)

The PTE1/SCRxD pin is the serial data input to the SCI receiver. The SCI shares the PTE1/SCRxD pin with port E. When the SCI is enabled, the PTE1/SCRxD pin is an input regardless of the state of the DDRE1 bit in data direction register E (DDRE).

18.8 I/O Registers

The following I/O registers control and monitor SCI operation:

- SCI control register 1 (SCC1)
- SCI control register 2 (SCC2)
- SCI control register 3 (SCC3)
- SCI status register 1 (SCS1)
- SCI status register 2 (SCS2)
- SCI data register (SCDR)
- SCI baud rate register (SCBR)

18.8.1 SCI Control Register 1

SCI control register 1:

- Enables loop mode operation
- Enables the SCI
- Controls output polarity
- Controls character length
- Controls SCI wakeup method
- Controls idle character detection
- Enables parity function
- Controls parity type

22.4 Port C

Port C is an 6-bit general-purpose bidirectional I/O port. Note that PTC5 is only available on 64-pin package options.

22.4.1 Port C Data Register

The port C data register contains a data latch for each of the six port C pins.

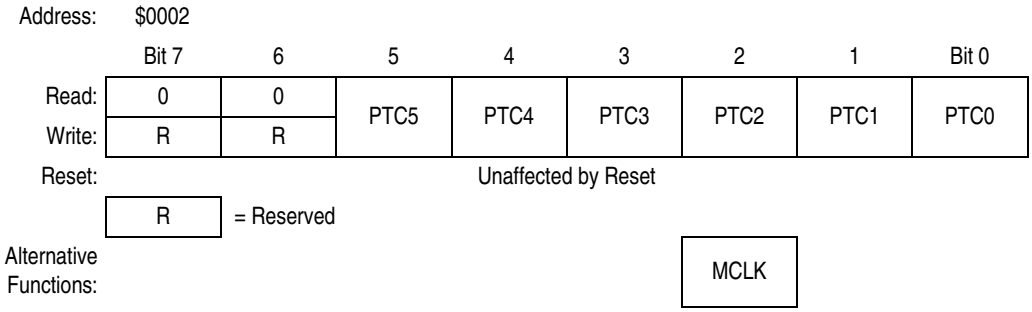


Figure 22-8. Port C Data Register (PTC)

PTC[5:0] — Port C Data Bits

These read/write bits are software-programmable. Data direction of each port C pin is under the control of the corresponding bit in data direction register C. Reset has no effect on port C data (5:0).

MCLK — System Clock Bit

The system clock is driven out of PTC2 when enabled by MCLKEN bit in PTCDDR7.

22.4.2 Data Direction Register C

Data direction register C determines whether each port C pin is an input or an output. Writing a logic 1 to a DDRC bit enables the output buffer for the corresponding port C pin; a logic 0 disables the output buffer.

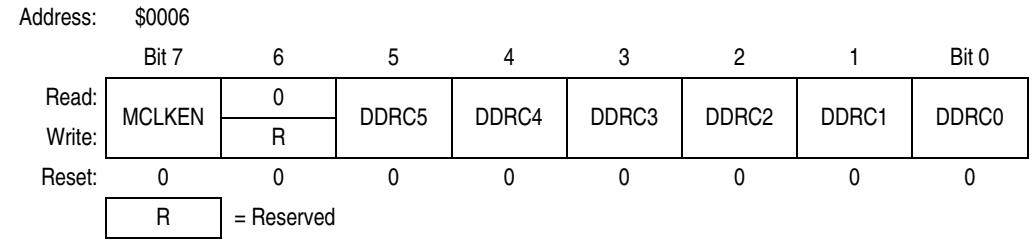


Figure 22-9. Data Direction Register C (DDRC)

MCLKEN — MCLK Enable Bit

This read/write bit enables MCLK to be an output signal on PTC2. If MCLK is enabled, DDRC2 has no effect. Reset clears this bit.

1 = MCLK output enabled
0 = MCLK output disabled

DDRC[5:0] — Data Direction Register C Bits

These read/write bits control port C data direction. Reset clears DDRC[7:0], configuring all port C pins as inputs.

1 = Corresponding port C pin configured as output
0 = Corresponding port C pin configured as input

NOTE

Avoid glitches on port C pins by writing to the port C data register before changing data direction register C bits from 0 to 1.

Figure 22-10 shows the port C I/O logic.

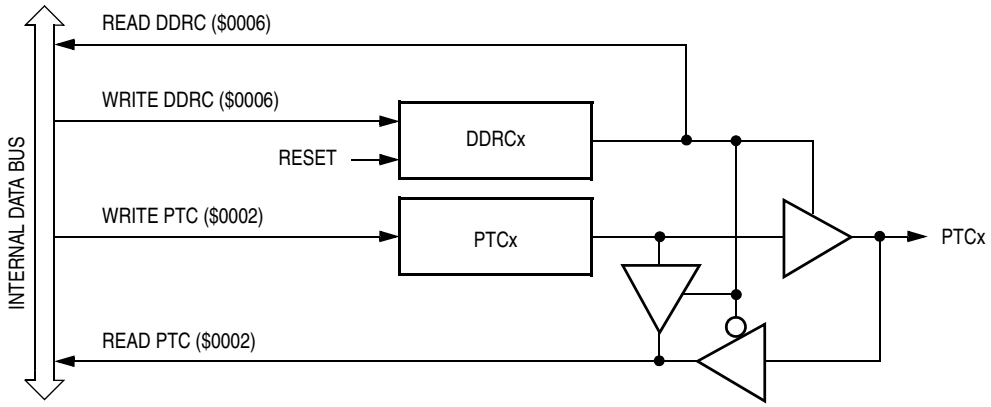


Figure 22-10. Port C I/O Circuit

When bit DDRCx is a logic 1, reading address $0002 reads the PTCx data latch. When bit DDRCx is a logic 0, reading address $0002 reads the voltage level on the pin. The data latch can always be written, regardless of the state of its data direction bit. Table 22-3 summarizes the operation of the port C pins.

Table 22-3. Port C Pin Functions

Bit Value	PTC Bit	I/O Pin Mode	Accesses to DDRC	Accesses to PTC	
			Read/Write	Read	Write
0	2	Input, Hi-Z	DDRC[2]	Pin	PTC2
1	2	Output	DDRC[2]	0	—
0	X	Input, Hi-Z	DDRC[5:0]	Pin	PTC[5:0](1)
1	X	Output	DDRC[5:0]	PTC[5:0]	PTC[5:0]

X = don't care
Hi-Z = high impedance
1. Writing affects data register, but does not affect input.

23.13.1 MSCAN08 Module Control Register 0

Address: $0500

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	SYNCH	TLNKEN	SLPAK	SLPRQ	SFTRES
Write:								
Reset:	0	0	0	0	0	0	0	1

= Unimplemented

Figure 23-15. Module Control Register 0 (CMCR0)

SYNCH — Synchronized Status

This bit indicates whether the MSCAN08 is synchronized to the CAN bus and as such can participate in the communication process.

1 = MSCAN08 synchronized to the CAN bus
0 = MSCAN08 not synchronized to the CAN bus

TLNKEN — Timer Enable

This flag is used to establish a link between the MSCAN08 and the on-chip timer (see 23.9 Timer Link).

1 = The MSCAN08 timer signal output is connected to the timer input.
0 = The port is connected to the timer input.

SLPAK — Sleep Mode Acknowledge

This flag indicates whether the MSCAN08 is in module internal sleep mode. It shall be used as a handshake for the sleep mode request (see 23.8.1 MSCAN08 Sleep Mode). If the MSCAN08 detects bus activity while in Sleep mode, it clears the flag.

1 = Sleep – MSCAN08 in internal sleep mode
0 = Wakeup – MSCAN08 is not in Sleep mode

SLPRQ — Sleep Request, Go to Internal Sleep Mode

This flag requests the MSCAN08 to go into an internal power-saving mode (see 23.8.1 MSCAN08 Sleep Mode).

1 = Sleep — The MSCAN08 will go into internal sleep mode.
0 = Wakeup — The MSCAN08 will function normally.

SFTRES — Soft Reset

When this bit is set by the CPU, the MSCAN08 immediately enters the soft reset state. Any ongoing transmission or reception is aborted and synchronization to the bus is lost.

The following registers enter and stay in their hard reset state: CMCR0, CRFLG, CRIER, CTFLG, and CTCR.

The registers CMCR1, CBTR0, CBTR1, CIDAC, CIDAR0–3, and CIDMR0–3 can only be written by the CPU when the MSCAN08 is in soft reset state. The values of the error counters are not affected by soft reset.

When this bit is cleared by the CPU, the MSCAN08 tries to synchronize to the CAN bus. If the MSCAN08 is not in bus-off state, it will be synchronized after 11 recessive bits on the bus; if the MSCAN08 is in bus-off state, it continues to wait for 128 occurrences of 11 recessive bits.

Clearing SFTRES and writing to other bits in CMCR0 must be in separate instructions.

1 = MSCAN08 in soft reset state
0 = Normal operation

23.13.3 MSCAN08 Bus Timing Register 0

Address: $0502

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
Reset:	0	0	0	0	0	0	0	0

Figure 23-17. Bus Timing Register 0 (CBTR0)

SJW1 and SJW0 — Synchronization Jump Width

The synchronization jump width (SJW) defines the maximum number of time quanta (T_q) clock cycles by which a bit may be shortened, or lengthened, to achieve resynchronization on data transitions on the bus (see Table 23-6).

Table 23-6. Synchronization Jump Width

SJW1	SJW0	Synchronization Jump Width
0	0	1 T_q cycle
0	1	2 T_q cycle
1	0	3 T_q cycle
1	1	4 T_q cycle

BRP5–BRP0 — Baud Rate Prescaler

These bits determine the time quanta (T_q) clock, which is used to build up the individual bit timing, according to Table 23-7.

Table 23-7. Baud Rate Prescaler

BRP5	BRP4	BRP3	BRP2	BRP1	BRP0	Prescaler Value (P)
0	0	0	0	0	0	1
0	0	0	0	0	1	2
0	0	0	0	1	0	3
0	0	0	0	1	1	4
:	:	:	:	:	:	:
:	:	:	:	:	:	:
1	1	1	1	1	1	64

NOTE

The CBTR0 register can be written only if the SFTRES bit in the MSCAN08 module control register is set.

23.13.9 MSCAN08 Identifier Acceptance Control Register

Address: $0508

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	IDAM1	IDAM0	0	0	IDHIT1	IDHIT0
Write:								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 23-23. Identifier Acceptance Control Register (CIDAC)

IDAM1–IDAM0— Identifier Acceptance Mode

The CPU sets these flags to define the identifier acceptance filter organization (see 23.5 Identifier Acceptance Filter). Table 23-9 summarizes the different settings. In “filter closed” mode no messages will be accepted so that the foreground buffer will never be reloaded.

Table 23-9. Identifier Acceptance Mode Settings

IDAM1	IDAM0	Identifier Acceptance Mode
0	0	Single 32-Bit Acceptance Filter
0	1	Two 16-Bit Acceptance Filter
1	0	Four 8-Bit Acceptance Filters
1	1	Filter Closed

IDHIT1–IDHIT0— Identifier Acceptance Hit Indicator

The MSCAN08 sets these flags to indicate an identifier acceptance hit (see 23.5 Identifier Acceptance Filter). Table 23-9 summarizes the different settings.

Table 23-10. Identifier Acceptance Hit Indication

IDHIT1	IDHIT0	Identifier Acceptance Hit
0	0	Filter 0 Hit
0	1	Filter 1 Hit
1	0	Filter 2 Hit
1	1	Filter 3 Hit

The IDHIT indicators are always related to the message in the foreground buffer. When a message gets copied from the background to the foreground buffer, the indicators are updated as well.

NOTE

The CIDAC register can be written only if the SFTRES bit in the CMCR0 is set.

CRC Error

A cyclical redundancy check (CRC) error is detected when the data bytes and CRC byte of a received message are processed and the CRC calculation result is not equal to $C4. The CRC code will detect any single and 2-bit errors, as well as all 8-bit burst errors and almost all other types of errors. The CRC error flag ($18 in BSVR) is set when a CRC error is detected. (See 27.6.4 BDLC State Vector Register.)

Symbol Error

A symbol error is detected when an abnormal (invalid) symbol is detected in a message being received from the J1850 bus. However, if the BDLC is transmitting when this happens, it will be treated as a loss of arbitration ($14 in BSVR) rather than a transmitter error. The ($1C) symbol invalid or the out-of-range flag is set when a symbol error is detected. Therefore, ($1C) symbol invalid flag is stacked behind the ($14) LOA flag during a transmission error process. (See 27.6.4 BDLC State Vector Register.)

Framing Error

A framing error is detected if an EOD or EOF symbol is detected on a non-byte boundary from the J1850 bus. A framing error also is detected if the BDLC is transmitting the EOD and instead receives an active symbol. The ($1C) symbol invalid or the out-of-range flag is set when a framing error is detected. (See 27.6.4 BDLC State Vector Register.)

Bus Fault

If a bus fault occurs, the response of the BDLC will depend upon the type of bus fault.
If the bus is shorted to battery, the BDLC will wait for the bus to fall to a passive state before it will attempt to transmit a message. As long as the short remains, the BDLC will never attempt to transmit a message onto the J1850 bus.

If the bus is shorted to ground, the BDLC will see an idle bus, begin to transmit the message, and then detect a transmission error ($1C in BSVR), since the short to ground would not allow the bus to be driven to the active (dominant) SOF state. The BDLC will abort that transmission and wait for the next CPU command to transmit.

In any case, if the bus fault is temporary, as soon as the fault is cleared, the BDLC will resume normal operation. If the bus fault is permanent, it may result in permanent loss of communication on the J1850 bus. (See BDLC State Vector Register.)

BREAK — Break

If a BREAK symbol is received while the BDLC is transmitting or receiving, an invalid symbol ($1C in BSVR) interrupt will be generated. Reading the BSVR register (see 27.6.4 BDLC State Vector Register.) will clear this interrupt condition. The BDLC will wait for the bus to idle, then wait for a start-of-frame (SOF) symbol.

The BDLC cannot transmit a BREAK symbol. It can only receive a BREAK symbol from the J1850 bus.

Chapter 28
Electrical Specifications

28.1 Electrical Specifications

28.1.1 Maximum Ratings

Maximum ratings are the extreme limits to which the microcontroller unit (MCU) can be exposed without permanently damaging it.

NOTE

This device is not guaranteed to operate properly at the maximum ratings. Refer to 28.1.4 5.0 Volt DC Electrical Characteristics for guaranteed operating conditions.

Rating[1]	Symbol	Value	Unit
Supply Voltage	V_{DD}	–0.3 to +6.0	V
Input Voltage	V_{IN}	V_{SS} –0.3 to V_{DD} +0.3	V
Maximum Current Per Pin Excluding V_{DD} and V_{SS}	I	± 25	mA
Storage Temperature	T_{STG}	–55 to +150	°C
Maximum Current out of V_{SS}	I_{MVSS}	100	mA
Maximum Current into V_{DD}	I_{MVDD}	100	mA
Reset and $\overline{IRQ}$ Input Voltage	V_{HI}	V_{DD} + 4.5	V

1. Voltages are referenced to V_{SS}.

NOTE

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. For proper operation, it is recommended that V_{IN} and V_{OUT} be constrained to the range $V_{SS} \le (V_{IN}$ or $V_{OUT}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (for example, either V_{SS} or V_{DD}).

$FE20 ↓ $FF1F	MONITOR ROM 256BYTES
$FF20 ↓ $FF6F	UNIMPLEMENTED 80 BYTES
$FF70	EEPROM-2 EEDIVH NONVOLATILE REGISTER (EE2DIVHNVR)
$FF71	EEPROM-2 EEDIVL NONVOLATILE REGISTER (EE2DIVLNVR)
$FF72	RESERVED
$FF73	RESERVED
$FF74	RESERVED
$FF75	RESERVED
$FF76	RESERVED
$FF77	RESERVED
$FF78	RESERVED
$FF79	RESERVED
$FF7A	EEPROM-2 EE DIVIDER HIGH REGISTER (EE2DIVH)
$FF7B	EEPROM-2 EE DIVIDER LOW REGISTER (EE2DIVL)
$FF7C	EEPROM-2 EEPROM NONVOLATILE REGISTER (EE2NVR)
$FF7D	EEPROM-2 EEPROM CONTROL REGISTER (EE2CR)
$FF7E	RESERVED
$FF7F	EEPROM-2 EEPROM ARRAY CONFIGURATION REGISTER (EE2ACR)
$FF80	FLASH-1 BLOCK PROTECT REGISTER (FL1BPR)
$FF81	FLASH-2 BLOCK PROTECT REGISTER (FL2BPR)
$FF82 ↓ $FF87	RESERVED 6 BYTES
$FF88	FLASH-1 CONTROL REGISTER (FL1CR)
$FF89	RESERVED
$FF8A	RESERVED
$FF8B ↓ $FFCB	RESERVED 64 BYTES
$FFCC ↓ $FFFF	VECTORS 52 BYTES See Table B-2

Figure B-1. MC68HC908AZ60E Memory Map (Sheet 3 of 3)

bus clock — The bus clock is derived from the CGMOUT output from the CGM. The bus clock frequency, f_{op}, is equal to the frequency of the oscillator output, CGMXCLK, divided by four.

byte — A set of eight bits.

C — The carry/borrow bit in the condition code register. The CPU08 sets the carry/borrow bit when an addition operation produces a carry out of bit 7 of the accumulator or when a subtraction operation requires a borrow. Some logical operations and data manipulation instructions also clear or set the carry/borrow bit (as in bit test and branch instructions and shifts and rotates).

CCR — See “condition code register.”

central processor unit (CPU) — The primary functioning unit of any computer system. The CPU controls the execution of instructions.

CGM — See “clock generator module (CGM).”

clear — To change a bit from logic 1 to logic 0; the opposite of set.

clock — A square wave signal used to synchronize events in a computer.

clock generator module (CGM) — A module in the M68HC08 Family. The CGM generates a base clock signal from which the system clocks are derived. The CGM may include a crystal oscillator circuit and or phase-locked loop (PLL) circuit.

comparator — A device that compares the magnitude of two inputs. A digital comparator defines the equality or relative differences between two binary numbers.

computer operating properly module (COP) — A counter module in the M68HC08 Family that resets the MCU if allowed to overflow.

condition code register (CCR) — An 8-bit register in the CPU08 that contains the interrupt mask bit and five bits that indicate the results of the instruction just executed.

control bit — One bit of a register manipulated by software to control the operation of the module.

control unit — One of two major units of the CPU. The control unit contains logic functions that synchronize the machine and direct various operations. The control unit decodes instructions and generates the internal control signals that perform the requested operations. The outputs of the control unit drive the execution unit, which contains the arithmetic logic unit (ALU), CPU registers, and bus interface.

COP — See "computer operating properly module (COP)."

counter clock — The input clock to the TIM counter. This clock is the output of the TIM prescaler.

CPU — See “central processor unit (CPU).”

CPU08 — The central processor unit of the M68HC08 Family.

CPU clock — The CPU clock is derived from the CGMOUT output from the CGM. The CPU clock frequency is equal to the frequency of the oscillator output, CGMXCLK, divided by four.

CPU cycles — A CPU cycle is one period of the internal bus clock, normally derived by dividing a crystal oscillator source by two or more so the high and low times will be equal. The length of time required to execute an instruction is measured in CPU clock cycles.