

Welcome to E-XFL.COM

#### What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

#### Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

E·XFl

Product Status	Active
Core Processor	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	CANbus, SCI, SPI
Peripherals	LVD, POR, PWM
Number of I/O	40
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 15x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°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=mc908az32avfue

Email: info@E-XFL.COM

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



General Description

# 1.4 Pin Assignments

Figure 1-2 shows the MC68HC908AZ32A pin assignments.





**NOTE** The following pin descriptions are just a quick reference. For a more detailed representation, see Chapter 23 I/O Ports.



# 7.3.2 Active Resets from Internal Sources

All internal reset sources actively pull the RST pin low for 32 CGMXCLK cycles to allow resetting of external peripherals. The internal reset signal IRST continues to be asserted for an additional 32 cycles (see Figure 7-5). An internal reset can be caused by an illegal address, illegal opcode, COP timeout, LVI, or POR (see Figure 7-6). Note that for LVI or POR resets, the SIM cycles through 4096 CGMXCLK cycles during which the SIM forces the RST pin low. The internal reset signal then follows the sequence from the falling edge of RST shown in Figure 7-5.

The COP reset is asynchronous to the bus clock.

The active reset feature allows the part to issue a reset to peripherals and other chips within a system built around the MCU.



Figure 7-6. Sources of Internal Reset

### 7.3.2.1 Power-On Reset

When power is first applied to the MCU, the power-on reset module (POR) generates a pulse to indicate that power-on has occurred. The external reset pin ( $\overline{\text{RST}}$ ) is held low while the SIM counter counts out 4096 CGMXCLK cycles. Another sixty-four CGMXCLK cycles later, the CPU and memories are released from reset to allow the reset vector sequence to occur. See Figure 7-7.

At power-on, the following events occur:

- A POR pulse is generated.
- The internal reset signal is asserted.
- The SIM enables CGMOUT.
- Internal clocks to the CPU and modules are held inactive for 4096 CGMXCLK cycles to allow stabilization of the oscillator.
- The  $\overline{RST}$  pin is driven low during the oscillator stabilization time.
- The POR bit of the SIM reset status register (SRSR) is set and all other bits in the register are cleared.



#### System Integration Module (SIM)

### 7.5.1 Interrupts

At the beginning of an interrupt, the CPU saves the CPU register contents on the stack and sets the interrupt mask (I bit) to prevent additional interrupts. At the end of an interrupt, the RTI instruction recovers the CPU register contents from the stack so that normal processing can resume. Figure 7-8 shows interrupt entry timing. Figure 7-9 shows interrupt recovery timing.

Interrupts are latched, and arbitration is performed in the SIM at the start of interrupt processing. The arbitration result is a constant that the CPU uses to determine which vector to fetch. Once an interrupt is latched by the SIM, no other interrupt can take precedence, regardless of priority, until the latched interrupt is serviced (or the I bit is cleared), see Figure 7-10.

	JLE IPT
I BIT	
IAB	X DUMMY SP - 1 X SP - 2 X SP - 4 X VECT H X X X
IDB	
R/W	
	Figure 7-8. Interrupt Entry
MOD	ULE JUPT
I BIT	
IAB	X X SP-4 X SP-3 X SP-2 X SP-1 X SP X PC X PC+1 X X X
IDB	X X X PC-1 [7:0] X PC-1[15:8] X OPCODE X OPERAND X X
R/W	Y

Figure 7-9. Interrupt Recovery



System Integration Module (SIM)



### 8.9.2 Parametric Influences on Reaction Time

Acquisition and lock times are designed to be as short as possible while still providing the highest possible stability. These reaction times are not constant, however. Many factors directly and indirectly affect the acquisition time.

The most critical parameter which affects the reaction times of the PLL is the reference frequency,  $f_{CGMRDV}$  (please reference Figure 8-1). This frequency is the input to the phase detector and controls how often the PLL makes corrections. For stability, the corrections must be small compared to the desired frequency, so several corrections are required to reduce the frequency error. Therefore, the slower the reference the longer it takes to make these corrections. This parameter is also under user control via the choice of crystal frequency  $f_{CGMXCLK}$ .

Another critical parameter is the external filter capacitor. The PLL modifies the voltage on the VCO by adding or subtracting charge from this capacitor. Therefore, the rate at which the voltage changes for a given frequency error (thus a change in charge) is proportional to the capacitor size. The size of the capacitor also is related to the stability of the PLL. If the capacitor is too small, the PLL cannot make small enough adjustments to the voltage and the system cannot lock. If the capacitor is too large, the PLL may not be able to adjust the voltage in a reasonable time. See 8.9.3 Choosing a Filter Capacitor.

Also important is the operating voltage potential applied to  $V_{DDA}$ . The power supply potential alters the characteristics of the PLL. A fixed value is best. Variable supplies, such as batteries, are acceptable if they vary within a known range at very slow speeds. Noise on the power supply is not acceptable, because it causes small frequency errors which continually change the acquisition time of the PLL.

Temperature and processing also can affect acquisition time because the electrical characteristics of the PLL change. The part operates as specified as long as these influences stay within the specified limits. External factors, however, can cause drastic changes in the operation of the PLL. These factors include noise injected into the PLL through the filter capacitor, filter capacitor leakage, stray impedances on the circuit board, and even humidity or circuit board contamination.

### 8.9.3 Choosing a Filter Capacitor

As described in 8.9.2 Parametric Influences on Reaction Time, the external filter capacitor,  $C_F$ , is critical to the stability and reaction time of the PLL. The PLL is also dependent on reference frequency and supply voltage. The value of the capacitor must, therefore, be chosen with supply potential and reference frequency in mind. For proper operation, the external filter capacitor must be chosen according to this equation:

$$C_{F} = C_{fact} \left( \frac{V_{DDA}}{f_{CGMRDV}} \right)$$

For acceptable values of  $C_{fact}$ , (see Chapter 25 Electrical Specifications). For the value of  $V_{DDA}$ , choose the voltage potential at which the MCU is operating. If the power supply is variable, choose a value near the middle of the range of possible supply values.

This equation does not always yield a commonly available capacitor size, so round to the nearest available size. If the value is between two different sizes, choose the higher value for better stability. Choosing the lower size may seem attractive for acquisition time improvement, but the PLL may become unstable. Also, always choose a capacitor with a tight tolerance ( $\pm 20\%$  or better) and low dissipation.



### Acquisition/Lock Time Specifications

bit is set. The resultant waveform can be captured on an oscilloscope and used to determine the typical lock time for the microcontroller and the associated external application circuit.

e.g.



NOTE

The filter capacitor should be fully discharged prior to making any measurements.



# Chapter 10 Configuration Register (CONFIG-2)

# **10.1 Introduction**

This section describes the configuration register (CONFIG-2). This register contains bits that configure these options:

- EEPROM Reference Clock Source
- EEPROM Read Protection

# **10.2 Functional Description**

The configuration register is a write-once register. Out of reset, the configuration register will read the default. Once the register is written, further writes will have no effect until a reset occurs.



### EEDIVCLK — EEPROM Timebase Divider Clock select bit

This bit selects the reference clock source for the EEPROM-1 and EEPROM-2 timebase divider modules.

1 = EExDIV clock input is driven by internal bus clock

0 = EExDIV clock input is driven by CGMXCLK

### EEMONSEC — EEPROM Read Protection in Monitor Mode Bit

When EEMONSEC is set the entire EEPROM array cannot be accessed in monitor mode unless a valid security code is entered.

1 = EEPROM read protection in monitor mode enabled

0 = EEPROM read protection in monitor mode disabled

### AZ32A — Device indicator

This read-only bit is used to indicate that the MC68HC908AZ32A is an 'A' suffix version HC08 rather than a non-'A'.

1 = 'A' version

0 = Non-'A' version



Monitor ROM (MON)

## 12.3.2 Data Format

Communication with the monitor ROM is in standard non-return-to-zero (NRZ) mark/space data format. (See Figure 12-2 and Figure 12-3.)

The data transmit and receive rate can be anywhere up to 28.8 kBaud. Transmit and receive baud rates must be identical.



Figure 12-3. Sample Monitor Waveforms

## 12.3.3 Echoing

As shown in Figure 12-4, the monitor ROM immediately echoes each received byte back to the PTA0 pin for error checking.

Any result of a command appears after the echo of the last byte of the command.



Figure 12-4. Read Transaction

### 12.3.4 Break Signal

A start bit followed by nine low bits is a break signal. (See Figure 12-5). When the monitor receives a break signal, it drives the PTA0 pin high for the duration of two bits before echoing the break signal.



Figure 12-5. Break Transaction



\_\_\_\_\_

### **Functional Description**

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
SCI Control Register 1 (SCC1)	Read: Write:	LOOPS	ENSCI	TXINV	М	WAKE	ILTY	PEN	PTY
	Reset:	0	0	0	0	0	0	0	0
SCI Control Register 2 (SCC2)	Read: Write:	SCTIE	TCIE	SCRIE	ILIE	TE	RE	RWU	SBK
	Reset:	0	0	0	0	0	0	0	0
	Read:	R8	тя	В	в	ORIE	NEIE	FEIE	DEIE
SCI Control Register 3 (SCC3)	Write:		10	n		Onic			
	Reset:	U	U	0	0	0	0	0	0
	Read:	SCTE	TC	SCRF	IDLE	OR	NF	FE	PE
SCI Status Register 1 (SCS1)	Write:								
	Reset:	1	1	0	0	0	0	0	0
	Read:	R7	R6	R5	R4	R3	R2	R1	R0
SCI Data Register (SCDR)	Write:	T7	T6	T5	T4	T3	T2	T1	T0
	Reset:				Unaffected	d by Reset			
	Read:	0	0	SCP1	SCPO	в	SCB2	SCB1	SCB0
SCI Baud Rate Register (SCBR)	Write:			0011	3010		00112	00111	00110
	Reset:	0	0	0	0	0	0	0	0
			= Unimplemented U = Unaffected R				R = Reserve	d	

Table 16-3.	SCI TI	ransmitter	I/O	Address	Summary
-------------	--------	------------	-----	---------	---------

Register	SCC1	SCC2	SCC3	SCS1	SCDR	SCBR
Address	\$0013	\$0014	\$0015	\$0016	\$0018	\$0019



#### Serial Communications Interface (SCI)

### 16.4.2.1 Character Length

The transmitter can accommodate either 8-bit or 9-bit data. The state of the M bit in SCI control register 1 (SCC1) determines character length. When transmitting 9-bit data, bit T8 in SCI control register 3 (SCC3) is the ninth bit (bit 8).

### 16.4.2.2 Character Transmission

During an SCI transmission, the transmit shift register shifts a character out to the TxD pin. The SCI data register (SCDR) is the write-only buffer between the internal data bus and the transmit shift register. To initiate an SCI transmission:

- 1. Enable the SCI by writing a logic 1 to the enable SCI bit (ENSCI) in SCI control register 1 (SCC1).
- 2. Enable the transmitter by writing a logic 1 to the transmitter enable bit (TE) in SCI control register 2 (SCC2).
- 3. Clear the SCI transmitter empty bit (SCTE) by first reading SCI status register 1 (SCS1) and then writing to the SCDR.
- 4. Repeat step 3 for each subsequent transmission.

At the start of a transmission, transmitter control logic automatically loads the transmit shift register with a preamble of logic 1s. After the preamble shifts out, control logic transfers the SCDR data into the transmit shift register. A logic 0 start bit automatically goes into the least significant bit position of the transmit shift register. A logic 1 stop bit goes into the most significant bit position.

The SCI transmitter empty bit, SCTE, in SCS1 becomes set when the SCDR transfers a byte to the transmit shift register. The SCTE bit indicates that the SCDR can accept new data from the internal data bus. If the SCI transmit interrupt enable bit, SCTIE, in SCC2 is also set, the SCTE bit generates a transmitter CPU interrupt request.

When the transmit shift register is not transmitting a character, the TxD pin goes to the idle condition, logic 1. If at any time software clears the ENSCI bit in SCI control register 1 (SCC1), the transmitter and receiver relinquish control of the port E pins.

### 16.4.2.3 Break Characters

Writing a logic 1 to the send break bit, SBK, in SCC2 loads the transmit shift register with a break character. A break character contains all logic 0s and has no start, stop, or parity bit. Break character length depends on the M bit in SCC1. As long as SBK is at logic 1, transmitter logic continuously loads break characters into the transmit shift register. After software clears the SBK bit, the shift register finishes transmitting the last break character and then transmits at least one logic 1. The automatic logic 1 at the end of a break character guarantees the recognition of the start bit of the next character.

The SCI recognizes a break character when a start bit is followed by eight or nine logic 0 data bits and a logic 0 where the stop bit should be. Receiving a break character has the following effects on SCI registers:

- Sets the framing error bit (FE) in SCS1
- Sets the SCI receiver full bit (SCRF) in SCS1
- Clears the SCI data register (SCDR)
- Clears the R8 bit in SCC3
- Sets the break flag bit (BKF) in SCS2
- May set the overrun (OR), noise flag (NF), parity error (PE), or reception in progress flag (RPF) bits







### 17.5.4 Transmission Initiation Latency

When the SPI is configured as a master (SPMSTR = 1), transmissions are started by a software write to the SPDR (\$0012). CPHA has no effect on the delay to the start of the transmission, but it does affect the initial state of the SCK signal. When CPHA = 0, the SCK signal remains inactive for the first half of the first SCK cycle. When CPHA = 1, the first SCK cycle begins with an edge on the SCK line from its inactive to its active level. The SPI clock rate (selected by SPR1–SPR0) affects the delay from the write to SPDR and the start of the SPI transmission. (See Figure 17-6). The internal SPI clock in the master is a free-running derivative of the internal MCU clock. It is only enabled when both the SPE and SPMSTR bits (SPCR) are set to conserve power. SCK edges occur half way through the low time of the internal MCU clock. Since the SPI clock is free-running, it is uncertain where the write to the SPDR will occur relative to the slower SCK. This uncertainty causes the variation in the initiation delay shown in Figure 17-6. This delay will be no longer than a single SPI bit time. That is, the maximum delay between the write to SPDR and the start of the SPI transmission is two MCU bus cycles for DIV2, eight MCU bus cycles for DIV8, 32 MCU bus cycles for DIV32, and 128 MCU bus cycles for DIV128.



Serial Peirpheral Interface (SPI)

# 17.7 Interrupts

Four SPI status flags can be enabled to generate CPU interrupt requests:

Table 17-3. SPI Interrupts

Flag	Request
SPTE (Transmitter Empty)	SPI Transmitter CPU Interrupt Request (SPTIE = 1)
SPRF (Receiver Full)	SPI Receiver CPU Interrupt Request (SPRIE = 1)
OVRF (Overflow)	SPI Receiver/Error Interrupt Request (SPRIE = 1, ERRIE = 1)
MODF (Mode Fault)	SPI Receiver/Error Interrupt Request (SPRIE = 1, ERRIE = 1, MODFEN = 1)

The SPI transmitter interrupt enable bit (SPTIE) enables the SPTE flag to generate transmitter CPU interrupt requests.

The SPI receiver interrupt enable bit (SPRIE) enables the SPRF bit to generate receiver CPU interrupt, provided that the SPI is enabled (SPE = 1).

The error interrupt enable bit (ERRIE) enables both the MODF and OVRF flags to generate a receiver/error CPU interrupt request.

The mode fault enable bit (MODFEN) can prevent the MODF flag from being set so that only the OVRF flag is enabled to generate receiver/error CPU interrupt requests.



Figure 17-9. SPI Interrupt Request Generation

Two sources in the SPI status and control register can generate CPU interrupt requests:

- 1. SPI receiver full bit (SPRF) The SPRF bit becomes set every time a byte transfers from the shift register to the receive data register. If the SPI receiver interrupt enable bit, SPRIE, is also set, SPRF can generate an SPI receiver/error CPU interrupt request.
- 2. SPI transmitter empty (SPTE) The SPTE bit becomes set every time a byte transfers from the transmit data register to the shift register. If the SPI transmit interrupt enable bit, SPTIE, is also set, SPTE can generate an SPTE CPU interrupt request.



# Chapter 21 Analog-To-Digital Converter (ADC-15)

# 21.1 Introduction

This section describes the analog-to-digital converter (ADC-15). The ADC is an 8-bit analog-to-digital converter.

# 21.2 Features

Features of the ADC module include:

- 15 Channels with Multiplexed Input
- Linear Successive Approximation
- 8-Bit Resolution
- Single or Continuous Conversion
- Conversion Complete Flag or Conversion Complete Interrupt
- Selectable ADC Clock

# **21.3 Functional Description**

Fifteen ADC channels are available for sampling external sources at pins

PTD6/ATD14/TCLK–PTD0/ATD8 and PTB7–PTB0. An analog multiplexer allows the single ADC converter to select one of 15 ADC channels as ADC voltage in (ADCVIN). ADCVIN is converted by the successive approximation register-based counters. When the conversion is completed, ADC places the result in the ADC data register and sets a flag or generates an interrupt. See Figure 21-1.

# 21.3.1 ADC Port I/O Pins

PTD6/ATD14/TCLK–PTD0/ATD8 and PTB7–PTB0 are general-purpose I/O pins that share with the ADC channels.

The channel select bits define which ADC channel/port pin will be used as the input signal. The ADC overrides the port I/O logic by forcing that pin as input to the ADC. The remaining ADC channels/port pins are controlled by the port I/O logic and can be used as general-purpose I/O. Writes to the port register or DDR will not have any affect on the port pin that is selected by the ADC. Read of a port pin which is in use by the ADC will return a logic 0 if the corresponding DDR bit is at logic 0. If the DDR bit is at logic 1, the value in the port data latch is read.

### NOTE

Do not use ADC channels ATD14 or ATD12 when using the PTD6/ATD14/TCLK or PTD4/ATD12 pins as the clock inputs for the 16-bit Timers.





### COCO — Conversions Complete Bit

When the AIEN bit is a logic 0, the COCO is a read-only bit which is set each time a conversion is completed. This bit is cleared whenever the ADC status and control register is written or whenever the ADC data register is read.

If the AIEN bit is a logic 1, the COCO is a read/write bit which selects the CPU to service the ADC interrupt request. Reset clears this bit.

1 = conversion completed (AIEN = 0)

0 = conversion not completed (AIEN = 0)

or

CPU interrupt enabled (AIEN = 1)

### AIEN — ADC Interrupt Enable Bit

When this bit is set, an interrupt is generated at the end of an ADC conversion. The interrupt signal is cleared when the data register is read or the status/control register is written. Reset clears the AIEN bit.

1 = ADC interrupt enabled

0 = ADC interrupt disabled

### ADCO — ADC Continuous Conversion Bit

When set, the ADC will convert samples continuously and update the ADR register at the end of each conversion. Only one conversion is allowed when this bit is cleared. Reset clears the ADCO bit.

1 = Continuous ADC conversion

0 = One ADC conversion

### ADCH[4:0] — ADC Channel Select Bits

ADCH4, ADCH3, ADCH2, ADCH1, and ADCH0 form a 5-bit field which is used to select one of 15 ADC channels. Channel selection is detailed in the following table. Care should be taken when using a port pin as both an analog and a digital input simultaneously to prevent switching noise from corrupting the analog signal. See Table 21-1.

The ADC subsystem is turned off when the channel select bits are all set to one. This feature allows for reduced power consumption for the MCU when the ADC is not used. Reset sets these bits.

### NOTE

Recovery from the disabled state requires one conversion cycle to stabilize.

ADCH4	ADCH3	ADCH2	ADCH1	ADCH0	Input Select
0	0	0	0	0	PTB0
0	0	0	0	1	PTB1
0	0	0	1	0	PTB2
0	0	0	1	1	PTB3
0	0	1	0	0	PTB4
0	0	1	0	1	PTB5
0	0	1	1	0	PTB6
0	0	1	1	1	PTB7
0	1	0	0	0	PTD0/ATD8/ATD8
0	1	0	0	1	PTD1/ATD9/ATD9
0	1	0	1	0	PTD2/ATD10/ATD10

Table 21-1. Mux Channel Select



### TACH[1:0] — Timer Channel I/O Bits

The PTE3/TCH1–PTE2/TCH0 pins are the TIM input capture/output compare pins. The edge/level select bits, ELSxB:ELSxA, determine whether the PTE3/TCH1–PTE2/TCH0 pins are timer channel I/O pins or general-purpose I/O pins. (See 18.8.4 TIMA Channel Status and Control Registers).

#### NOTE

Data direction register E (DDRE) does not affect the data direction of port E pins that are being used by the TIM. However, the DDRE bits always determine whether reading port E returns the states of the latches or the states of the pins. (See Table 23-5).

### RxD — SCI Receive Data Input Bit

The PTE1/RxD pin is the receive data input for the SCI module. When the enable SCI bit, ENSCI, is clear, the SCI module is disabled, and the PTE1/RxD pin is available for general-purpose I/O. (See 16.8.1 SCI Control Register 1).

### TxD — SCI Transmit Data Output

The PTE0/TxD pin is the transmit data output for the SCI module. When the enable SCI bit, ENSCI, is clear, the SCI module is disabled, and the PTE0/TxD pin is available for general-purpose I/O. (See 16.8.1 SCI Control Register 1).

#### NOTE

Data direction register E (DDRE) does not affect the data direction of port E pins that are being used by the SCI module. However, the DDRE bits always determine whether reading port E returns the states of the latches or the states of the pins. (See Table 23-5).

### 23.6.2 Data Direction Register E

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





### DDRE[7:0] — Data Direction Register E Bits

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

1 = Corresponding port E pin configured as output

0 = Corresponding port E pin configured as input

#### NOTE

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



To transmit a message, the CPU08 has to identify an available transmit buffer which is indicated by a set transmit buffer empty (TXE) flag in the MSCAN08 transmitter flag register (CTFLG) (see 24.13.7 MSCAN08 Transmitter Flag Register).

The CPU08 then stores the identifier, the control bits and the data content into one of the transmit buffers. Finally, the buffer has to be flagged ready for transmission by clearing the TXE flag.

The MSCAN08 then will schedule the message for transmission and will signal the successful transmission of the buffer by setting the TXE flag. A transmit interrupt is generated<sup>(1)</sup> when TXE is set and can be used to drive the application software to re-load the buffer.

In case more than one buffer is scheduled for transmission when the CAN bus becomes available for arbitration, the MSCAN08 uses the local priority setting of the three buffers for prioritisation. For this purpose, every transmit buffer has an 8-bit local priority field (PRIO). The application software sets this field when the message is set up. The local priority reflects the priority of this particular message relative to the set of messages being emitted from this node. The lowest binary value of the PRIO field is defined as the highest priority.

The internal scheduling process takes place whenever the MSCAN08 arbitrates for the bus. This is also the case after the occurrence of a transmission error.

When a high priority message is scheduled by the application software, it may become necessary to abort a lower priority message being set up in one of the three transmit buffers. As messages that are already under transmission cannot be aborted, the user has to request the abort by setting the corresponding abort request flag (ABTRQ) in the transmission control register (CTCR). The MSCAN08 will then grant the request, if possible, by setting the corresponding abort request acknowledge (ABTAK) and the TXE flag in order to release the buffer and by generating a transmit interrupt. The transmit interrupt handler software can tell from the setting of the ABTAK flag whether the message was actually aborted (ABTAK = 1) or sent (ABTAK = 0).

# 24.5 Identifier Acceptance Filter

The Identifier Acceptance Registers (CIDAR0-3) define the acceptance patterns of the standard or extended identifier (ID10-ID0 or ID28-ID0). Any of these bits can be marked 'don't care' in the Identifier Mask Registers (CIDMR0-3).

A filter hit is indicated to the application on software by a set RXF (Receive Buffer Full Flag, see 24.13.5 MSCAN08 Receiver Flag Register (CRFLG)) and two bits in the Identifier Acceptance Control Register (see 24.13.9 MSCAN08 Identifier Acceptance Control Register). These Identifier Hit Flags (IDHIT1-0) clearly identify the filter section that caused the acceptance. They simplify the application software's task to identify the cause of the receiver interrupt. In case that more than one hit occurs (two or more filters match) the lower hit has priority.

A very flexible programmable generic identifier acceptance filter has been introduced to reduce the CPU interrupt loading. The filter is programmable to operate in four different modes:

 Single identifier acceptance filter, each to be applied to a) the full 29 bits of the extended identifier and to the following bits of the CAN frame: RTR, IDE, SRR or b) the 11 bits of the standard identifier plus the RTR and IDE bits of CAN 2.0A/B messages. This mode implements a single filter for a full length CAN 2.0B compliant extended identifier. Figure 24-3 shows how the 32-bit filter bank (CIDAR0-3, CIDMR0-3) produces a filter 0 hit.

<sup>1.</sup> The transmit interrupt will occur only if not masked. A polling scheme can be applied on TXE also.



Figure 24-7. Clocking Scheme

A programmable prescaler is used to generate out of the MSCAN08 clock the time quanta (Tq) clock. A time quantum is the atomic unit of time handled by the MSCAN08.

$$f_{Tq} = \frac{f_{MSCANCLK}}{Presc value}$$

A bit time is subdivided into three segments<sup>(1)</sup> (see Figure 24-8).

- SYNC\_SEG: This segment has a fixed length of one time quantum. Signal edges are expected to happen within this section.
- Time segment 1: This segment includes the PROP\_SEG and the PHASE\_SEG1 of the CAN standard. It can be programmed by setting the parameter TSEG1 to consist of 4 to 16 time quanta.
- Time segment 2: This segment represents PHASE\_SEG2 of the CAN standard. It can be programmed by setting the TSEG2 parameter to be 2 to 8 time quanta long.

Bit rate = 
$$\frac{f_{Tq}}{No. of time quanta}$$

The synchronization jump width (SJW) can be programmed in a range of 1 to 4 time quanta by setting the SJW parameter.

<sup>1.</sup> For further explanation of the underlying concepts please refer to ISO/DIS 11 519-1, Section 10.3.



#### MSCAN Controller (MSCAN08)

Addr	Register		Bit 7	6	5	4	3	2	1	Bit 0
\$05b0	IDR0	Read: Write:	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3
\$05b1	IDR1	Read: Write:	ID2	ID1	ID0	RTR	IDE(=0)			
\$05b2	IDR2	Read: Write:								
\$05b3	IDR3	Read: Write:								
				Unimplemented						

= Unimplemented

Figure 24-12. Standard Identifier Mapping

### 24.12.2 Identifier Registers

The identifiers consist of either 11 bits (ID10–ID0) for the standard, or 29 bits (ID28–ID0) for the extended format. ID10/28 is the most significant bit and is transmitted first on the bus during the arbitration procedure. The priority of an identifier is defined to be highest for the smallest binary number.

### SRR — Substitute Remote Request

This fixed recessive bit is used only in extended format. It must be set to 1 by the user for transmission buffers and will be stored as received on the CAN bus for receive buffers.

### IDE — ID Extended

This flag indicates whether the extended or standard identifier format is applied in this buffer. In case of a receive buffer, the flag is set as being received and indicates to the CPU how to process the buffer identifier registers. In case of a transmit buffer, the flag indicates to the MSCAN08 what type of identifier to send.

1 = Extended format, 29 bits

0 = Standard format, 11 bits

### **RTR** — Remote Transmission Request

This flag reflects the status of the remote transmission request bit in the CAN frame. In case of a receive buffer, it indicates the status of the received frame and supports the transmission of an answering frame in software. In case of a transmit buffer, this flag defines the setting of the RTR bit to be sent.

1 = Remote frame

0 = Data frame

### 24.12.3 Data Length Register (DLR)

This register keeps the data length field of the CAN frame.

### DLC3–DLC0 — Data Length Code Bits

The data length code contains the number of bytes (data byte count) of the respective message. At transmission of a remote frame, the data length code is transmitted as programmed while the number of transmitted bytes is always 0. The data byte count ranges from 0 to 8 for a data frame. Table 24-5 shows the effect of setting the DLC bits.



- **pull** An instruction that copies into the accumulator the contents of a stack RAM location. The stack RAM address is in the stack pointer.
- **pullup** A transistor in the output of a logic gate that connects the output to the logic 1 voltage of the power supply.
- pulse-width The amount of time a signal is on as opposed to being in its off state.
- **pulse-width modulation (PWM)** Controlled variation (modulation) of the pulse width of a signal with a constant frequency.
- **push** An instruction that copies the contents of the accumulator to the stack RAM. The stack RAM address is in the stack pointer.
- **PWM period** The time required for one complete cycle of a PWM waveform.
- **RAM** Random access memory. All RAM locations can be read or written by the CPU. The contents of a RAM memory location remain valid until the CPU writes a different value or until power is turned off.
- **RC circuit** A circuit consisting of capacitors and resistors having a defined time constant.
- read To copy the contents of a memory location to the accumulator.
- register A circuit that stores a group of bits.
- **reserved memory location** A memory location that is used only in special factory test modes. Writing to a reserved location has no effect. Reading a reserved location returns an unpredictable value.
- reset To force a device to a known condition.
- **ROM** Read-only memory. A type of memory that can be read but cannot be changed (written). The contents of ROM must be specified before manufacturing the MCU.
- SCI See "serial communication interface module (SCI)."
- serial Pertaining to sequential transmission over a single line.
- serial communications interface module (SCI) A module in the M68HC08 Family that supports asynchronous communication.
- serial peripheral interface module (SPI) A module in the M68HC08 Family that supports synchronous communication.
- set To change a bit from logic 0 to logic 1; opposite of clear.
- **shift register** A chain of circuits that can retain the logic levels (logic 1 or logic 0) written to them and that can shift the logic levels to the right or left through adjacent circuits in the chain.
- signed A binary number notation that accommodates both positive and negative numbers. The most significant bit is used to indicate whether the number is positive or negative, normally logic 0 for positive and logic 1 for negative. The other seven bits indicate the magnitude of the number.
- **software** Instructions and data that control the operation of a microcontroller.



Glossary

software interrupt (SWI) — An instruction that causes an interrupt and its associated vector fetch.

- SPI See "serial peripheral interface module (SPI)."
- **stack** A portion of RAM reserved for storage of CPU register contents and subroutine return addresses.
- stack pointer (SP) A 16-bit register in the CPU08 containing the address of the next available storage location on the stack.
- **start bit** A bit that signals the beginning of an asynchronous serial transmission.
- status bit A register bit that indicates the condition of a device.
- stop bit A bit that signals the end of an asynchronous serial transmission.
- subroutine A sequence of instructions to be used more than once in the course of a program. The last instruction in a subroutine is a return from subroutine (RTS) instruction. At each place in the main program where the subroutine instructions are needed, a jump or branch to subroutine (JSR or BSR) instruction is used to call the subroutine. The CPU leaves the flow of the main program to execute the instructions in the subroutine. When the RTS instruction is executed, the CPU returns to the main program where it left off.
- **synchronous** Refers to logic circuits and operations that are synchronized by a common reference signal.
- TIM See "timer interface module (TIM)."
- timer interface module (TIM) A module used to relate events in a system to a point in time.
- timer A module used to relate events in a system to a point in time.
- toggle To change the state of an output from a logic 0 to a logic 1 or from a logic 1 to a logic 0.
- tracking mode Mode of low-jitter PLL operation during which the PLL is locked on a frequency. Also see "acquisition mode."
- two's complement A means of performing binary subtraction using addition techniques. The most significant bit of a two's complement number indicates the sign of the number (1 indicates negative). The two's complement negative of a number is obtained by inverting each bit in the number and then adding 1 to the result.
- unbuffered Utilizes only one register for data; new data overwrites current data.
- **unimplemented memory location** A memory location that is not used. Writing to an unimplemented location has no effect. Reading an unimplemented location returns an unpredictable value. Executing an opcode at an unimplemented location causes an illegal address reset.
- V —The overflow bit in the condition code register of the CPU08. The CPU08 sets the V bit when a two's complement overflow occurs. The signed branch instructions BGT, BGE, BLE, and BLT use the overflow bit.
- **variable** A value that changes during the course of program execution.