Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Obsolete
Core Processor	CPU32+
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	25MHz
Co-Processors/DSP	Communications; CPM
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	10Mbps (1)
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	357-BBGA
Supplier Device Package	357-PBGA (25x25)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68360vr25l

tire word access will be retried. This is true even if the relinquish and retry was asserted on the second access and the first 8-bit access was completed normally.

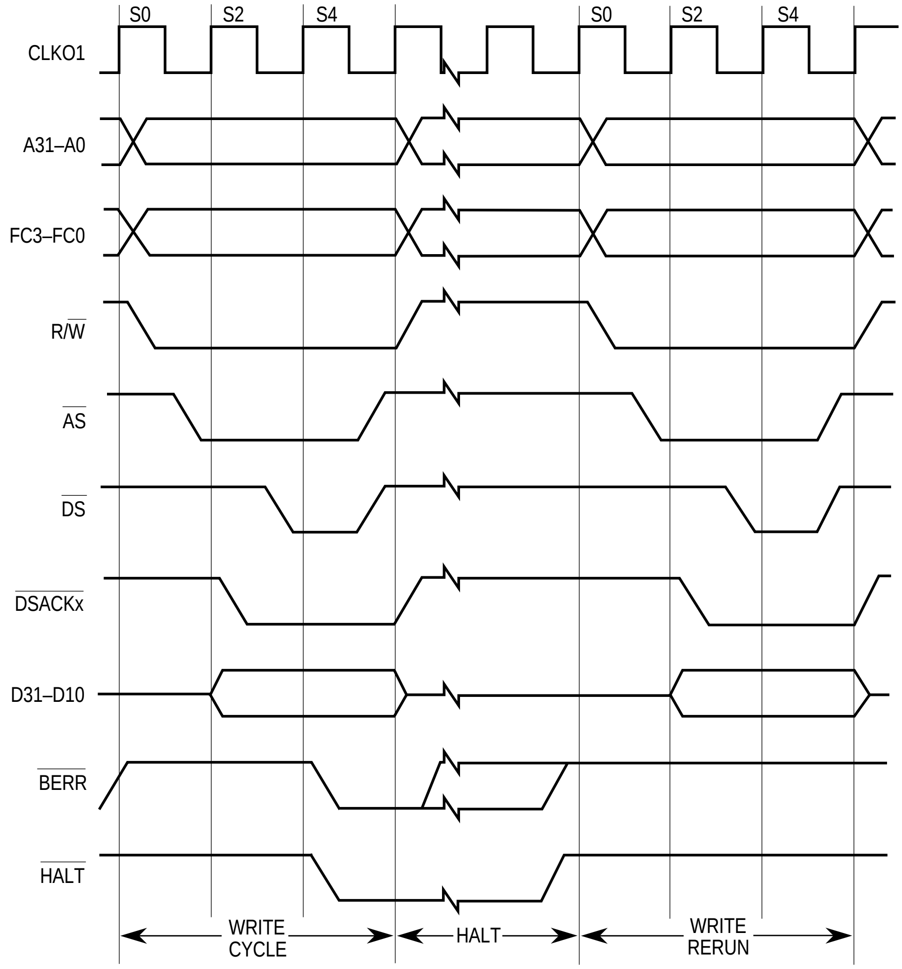


Figure 4-32. Late Retry Sequence

4.5.3 Halt Operation

When HALT is asserted and BERR is not asserted, the QUICC halts external bus activity at the next bus cycle boundary (see Figure 4-33). HALT by itself does not terminate a bus cycle. HALT affects external bus cycles only; thus, a program that does not require use of the external bus may continue executing until it requires use of the external bus.

Negating and reasserting HALT in accordance with the correct timing requirements provides a single step (bus cycle to bus cycle) operation. The single-cycle mode allows the user to proceed through (and debug) external QUICC operations, one bus cycle at a time. Since the occurrence of a bus error while HALT is asserted causes a retry operation, the user must anticipate retry cycles while debugging in the single-cycle mode. The single-step operation and the software trace capability allow the system debugger to trace single bus cycles, single instructions, or changes in program flow.

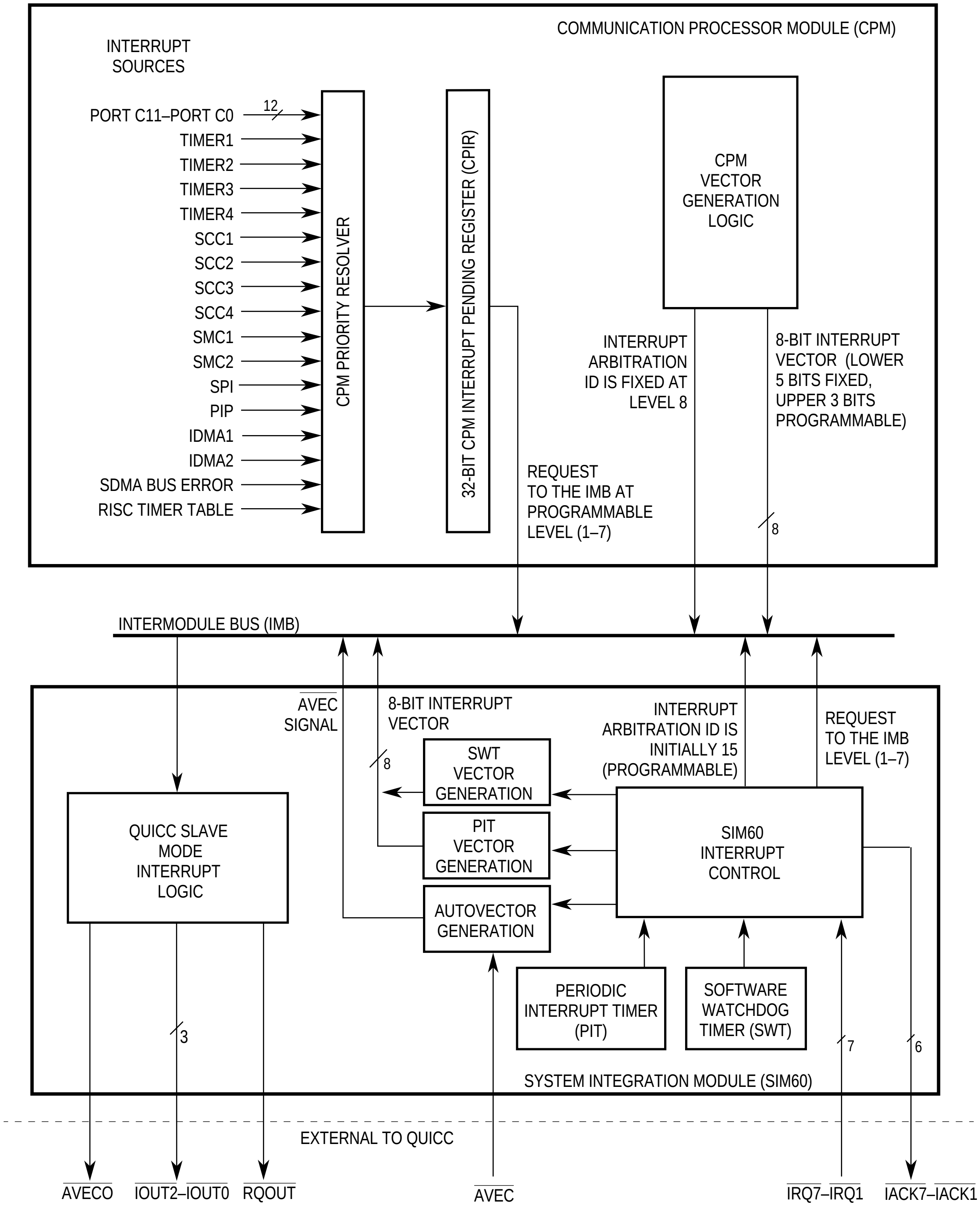

COMMUNICATION PROCESSOR MODULE (CPM)
INTERRUPT SOURCES
PORT C11–PORT C0
12
TIMER1
TIMER2
TIMER3
TIMER4
SCC1
SCC2
SCC3
SCC4
SMC1
SMC2
SPI
PIP
IDMA1
IDMA2
SDMA BUS ERROR
RISC TIMER TABLE
CPM PRIORITY RESOLVER
32-BIT CPM INTERRUPT PENDING REGISTER (CPIR)
CPM VECTOR GENERATION LOGIC
INTERRUPT ARBITRATION ID IS FIXED AT LEVEL 8
8-BIT INTERRUPT VECTOR (LOWER 5 BITS FIXED, UPPER 3 BITS PROGRAMMABLE)
REQUEST TO THE IMB AT PROGRAMMABLE LEVEL (1–7)
8
INTERMODULE BUS (IMB)
AVEC SIGNAL
8-BIT INTERRUPT VECTOR
8
INTERRUPT ARBITRATION ID IS INITIALLY 15 (PROGRAMMABLE)
REQUEST TO THE IMB LEVEL (1–7)
SWT VECTOR GENERATION
PIT VECTOR GENERATION
AUTOVECTOR GENERATION
QUICC SLAVE MODE INTERRUPT LOGIC
SIM60 INTERRUPT CONTROL
PERIODIC INTERRUPT TIMER (PIT)
SOFTWARE WATCHDOG TIMER (SWT)
3
7
6
SYSTEM INTEGRATION MODULE (SIM60)
EXTERNAL TO QUICC
AVECO
IOUT2–IOUT0
RQOUT
AVEC
IRQ7–IRQ1
IACK7–IACK1

Figure 6-3. QUICC Interrupt Structure

In the interrupt arbitration process, the module places its arbitration ID on the IMB. The arbitration ID ranges in value from 0 to 15. The SIM60 interrupt controller arbitration ID is initialized to 15 (the highest value), but may be lowered if desired. The higher arbitration value always wins.

DE—Destination Access Bus Error

The buffer was closed due to a bus error on the destination access. An interrupt (BED) will be generated, regardless of the I-bit. The RISC will clear the V-bit of this BD.

DA—Done Asserted During Transfer

The buffer was closed due to the assertion of $\overline{\text{DONEx}}$. An interrupt (DONE) will be generated, regardless of the I-bit. The RISC will clear the V-bit of this BD.

Data Length

The data length is the number of bytes that the IDMA should transfer from/to this BD's data buffer. The data length should be programmed to a value greater than zero.

Source Buffer Pointer

The source buffer pointer contains the address of the associated source data buffer. The buffer may reside in either internal or external memory.

NOTE

In single address mode when the source is a device, this field is ignored. In dual address mode when the source is a device, this field should contain the device address.

Destination Buffer Pointer

The destination buffer pointer contains the address of the associated destination data buffer. The buffer may reside in either internal or external memory.

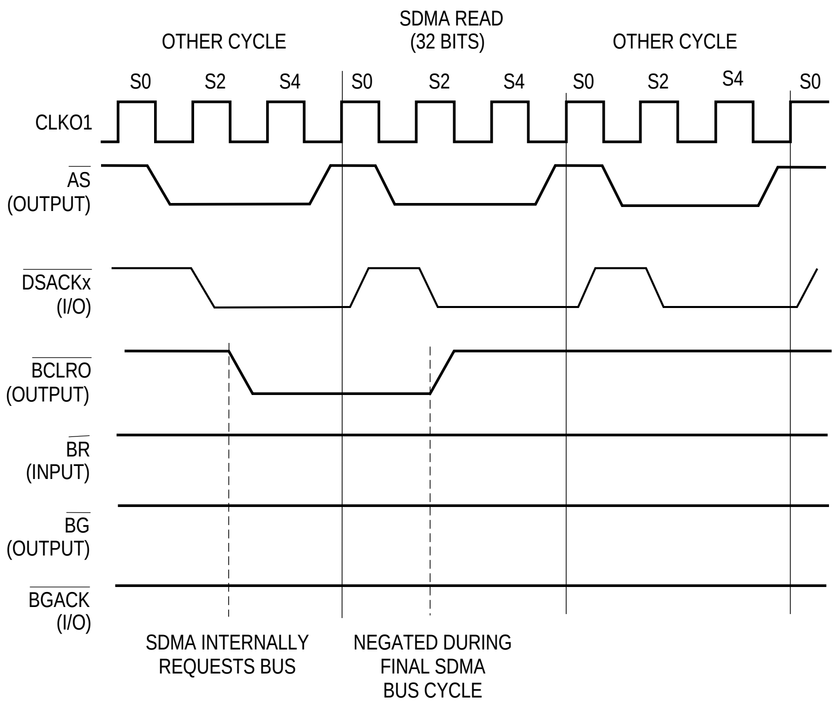
NOTE

In single address mode when the destination is a device, this field is ignored. In dual address mode when the destination is a device, this field should contain the device address.

7.6.4.2.3 IDMA Commands (INIT_IDMA). This command causes the RISC controller to reinitialize its IDMA internal state to the condition it had after a system reset. The IDMA BD pointer is reinitialized to the top of BD ring. When in the auto buffer and buffer chaining modes, the IDMA can be reset by setting the RST bit in the CMR and issuing the INIT_IDMA command. The INIT_IDMA command should only be executed in conjunction with the setting of the RST bit in the CMR.

7.6.4.3 STARTING THE IDMA. Once the channel has been initialized with all parameters required for a transfer operation, it is started by setting the STR bit in the CMR. After the channel has been started, any register that describes the current operation may be read but not modified (SAPR, DAPR, FCR, or BCR).

Once STR has been set, the channel is active and either accepts operand transfer requests in external mode or generates requests automatically in internal mode. When the first valid external request is recognized, the IDMA arbitrates for the bus. The $\overline{\text{DREQx}}$ input is ignored until STR is set.



NOTES:
1. The BCLRO signal is only asserted if the SDMA bus arbitration ID is greater than the BCLROID2–BCLROID0 bits in the SIM60 module configuration register.
2. The BR, BG, and BGACK signals are not affected by the SDMA bus arbitration process if the CPU32+ is enabled.

Figure 7-18. SDMA Bus Arbitration (Normal Operation)

The relative priority between the two IDMAs and the SDMA channels is user programmable. Regardless of system configuration, if the IDMA is a bus master when a higher priority SDMA channel needs to transfer over the bus, the SDMA will steal cycles from the IDMA with no arbitration overhead.

When the QUICC is in slave mode (CPU32+ is disabled) the SDMA can steal cycles from the IDMA with no arbitration overhead. See Section 4 Bus Operation for diagrams of bus arbitration by an internal master in slave mode.

7.7.2 SDMA Registers

The SDMA channels have one configuration register; otherwise, they are controlled transparently to the user, through the configuration of the SCCs, SMCs, and SPI. The only user-accessible registers associated with the SDMA are the SDMA configuration register (SDCR), SDMA address register (SDAR), a read-only register used for diagnostics in case of an SDMA bus error, and the SDMA status register (SDSR).

7.7.2.1 SDMA CONFIGURATION REGISTER (SDCR). The 16-bit SDCR is used to configure all 14 SDMA channels. It is always readable and writable in the supervisor mode, although writing the SDCR is not recommended unless the CP is disabled. SDCR is cleared at reset.

7.8.5.3 SI CLOCK ROUTE REGISTER (SICR). The 32-bit SICR is used to define the SCC clock sources. The clock source can be one of the four baud rate generators or an input from a bank of clock pins. The SICR appears to the user as a memory-mapped, read-write register and is cleared at reset.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GR4	SC4	R4CS			T4CS			GR3	SC3	R3CS			T3CS		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GR2	SC2	R2CS			T2CS			GR1	SC1	R1CS			T1CS		

GRx—Grant Support of SCCx

0 = SCCx transmitter does not support the grant mechanism. The grant is always asserted internally.

1 = SCCx transmitter supports the grant mechanism as determined by the GMx bit of its channel.

SCx—SCCx Connection

0 = SCCx is not connected to the multiplexed SI but is either connected directly to the NMSIx pins or is not used. The choice of general-purpose I/O port pins versus SCCn pins is made in the parallel I/O control register.

1 = SCCx is connected to the multiplexed SI. The NMSIx receive pins are available for other purposes.

RxCS—Receive Clock Source for SCCx

These bits are ignored when the SCCx is connected to the TSA (SCx = 1).

000 = SCCx receive clock is BRG1.
001 = SCCx receive clock is BRG2.
010 = SCCx receive clock is BRG3.
011 = SCCx receive clock is BRG4.
100 = SCCx receive clock for x = 1,2 is CLK1 and for x = 3,4 is CLK5.
101 = SCCx receive clock for x = 1,2 is CLK2 and for x = 3,4 is CLK6.
110 = SCCx receive clock for x = 1,2 is CLK3 and for x = 3,4 is CLK7.
111 = SCCx receive clock for x = 1,2 is CLK4 and for x = 3,4 is CLK8.

TxCS—Transmit Clock Source for SCCx

These bits are ignored when SCCx is connected to the TSA (SCx = 1).

000 = SCCx transmit clock is BRG1.
001 = SCCx transmit clock is BRG2.
010 = SCCx transmit clock is BRG3.
011 = SCCx transmit clock is BRG4.
100 = SCCx transmit clock for x = 1,2 is CLK1 and for x = 3,4 is CLK5.
101 = SCCx transmit clock for x = 1,2 is CLK2 and for x = 3,4 is CLK6.
110 = SCCx transmit clock for x = 1,2 is CLK3 and for x = 3,4 is CLK7.
111 = SCCx transmit clock for x = 1,2 is CLK4 and for x = 3,4 is CLK8.

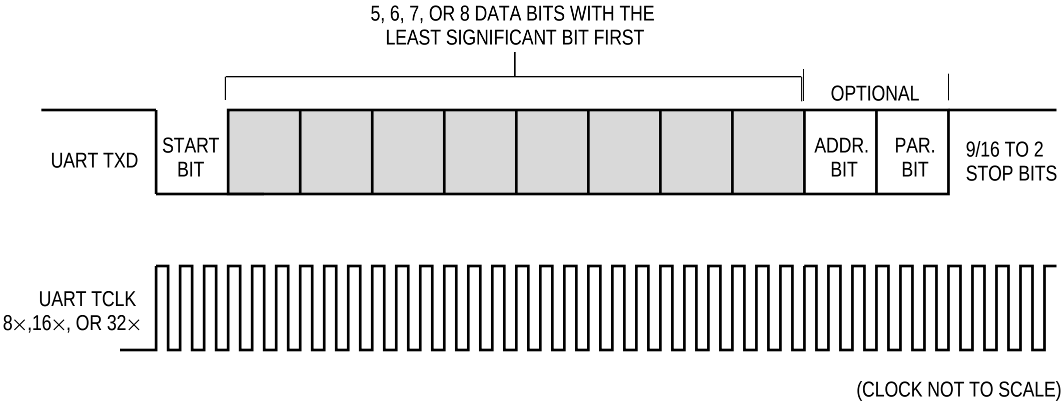


Figure 7-46. UART Character Format

Since the transmitter and receiver work asynchronously, there is no need to connect transmit and receive clocks. Instead, the receiver oversamples the incoming data stream (usually by a factor of 16) and uses some of these samples to determine the bit value. Traditionally the middle three samples of the sixteen samples are used. Two UARTs can communicate using a system like this if parameters, such as the parity scheme and character length, are the same for both transmitter and receiver.

When data is not transmitted in the UART protocol, a continuous stream of ones is transmitted, called the idle condition. Since the start bit is always a zero, the receiver can detect when real data is once again present on the line. UART also specifies an all-zeros character called a break, which is used to abort a character transfer sequence.

Many different protocols have been defined using asynchronous characters, but the most popular of these is the RS-232 standard. RS-232 specifies standard baud rates, handshaking protocols, and mechanical/electrical details. Another popular standard using the same character format is RS-485, which defines a balanced line system allowing longer cables than RS-232 links. Synchronous protocols like HDLC are sometimes defined to run over asynchronous links. Other protocols like Profibus extend the UART protocol to include LAN-oriented features such as token passing.

All the standards provide handshaking signals, but some systems require just three physical lines: transmit data, receive data, and ground.

Many proprietary standards have been built around the asynchronous character frame, and some even implement a multidrop configuration. In multidrop systems, more than two stations may be present on a network, with each having a specific address. Frames made up of many characters may be broadcast, with the first character acting as a destination address. To allow this, the UART frame is extended by one bit to distinguish between an address character and the normal data characters.

Additionally, a synchronous form of the UART protocol exists where start and stop bits are still present, but a clock is provided with each bit, so the oversampling technique is not required. This mode is called "isochronous" operation or, more often, synchronous UART.

In normal UART mode with 16× oversampling, the FSB bits (14–11) in the DSR are decoded as follows:

1111 =	Last Transmitted Stop Bit 16/16 (the default value after reset)
1110 =	Last Transmitted Stop Bit 15/16
1101 =	Last Transmitted Stop Bit 14/16
1100 =	Last Transmitted Stop Bit 13/16
1011 =	Last Transmitted Stop Bit 12/16
1010 =	Last Transmitted Stop Bit 11/16
1001 =	Last Transmitted Stop Bit 10/16
1000 =	Last Transmitted Stop Bit 9/16
0xxx =	Invalid. Do not use.

When the UART is configured for 32× oversampling, the FSB bits (14–11) in the DSR are decoded as follows:

1111 =	Last Transmitted Stop Bit 32/32 (the default value after reset)
1110 =	Last Transmitted Stop Bit 31/32
1101 =	Last Transmitted Stop Bit 30/32
1100 =	Last Transmitted Stop Bit 29/32
1011 =	Last Transmitted Stop Bit 28/32
1010 =	Last Transmitted Stop Bit 27/32
1001 =	Last Transmitted Stop Bit 26/32
1000 =	Last Transmitted Stop Bit 25/32
0111 =	Last Transmitted Stop Bit 24/32
0110 =	Last Transmitted Stop Bit 23/32
0101 =	Last Transmitted Stop Bit 22/32
0100 =	Last Transmitted Stop Bit 21/32
0011 =	Last Transmitted Stop Bit 20/32
0010 =	Last Transmitted Stop Bit 19/32
0001 =	Last Transmitted Stop Bit 18/32
0000 =	Last Transmitted Stop Bit 17/32

When the UART is configured for 8× oversampling, the FSB bits (14–11) in the DSR are decoded as follows:

1111 =	Last Transmitted Stop Bit 8/8 (the default value after reset)
1110 =	Last Transmitted Stop Bit 7/8
1101 =	Last Transmitted Stop Bit 6/8
1100 =	Last Transmitted Stop Bit 5/8
10xx =	Invalid. Do not use.
01xx =	Invalid. Do not use.
00xx =	Invalid. Do not use.

The UART receiver can always receive fractional stop bits. The next character's start bit may begin at any time after the three middle samples of the stop bit have been taken.

7.10.16.14 UART ERROR-HANDLING PROCEDURE. The UART controller reports character reception and transmission error conditions via the channel BDs, the error counters,

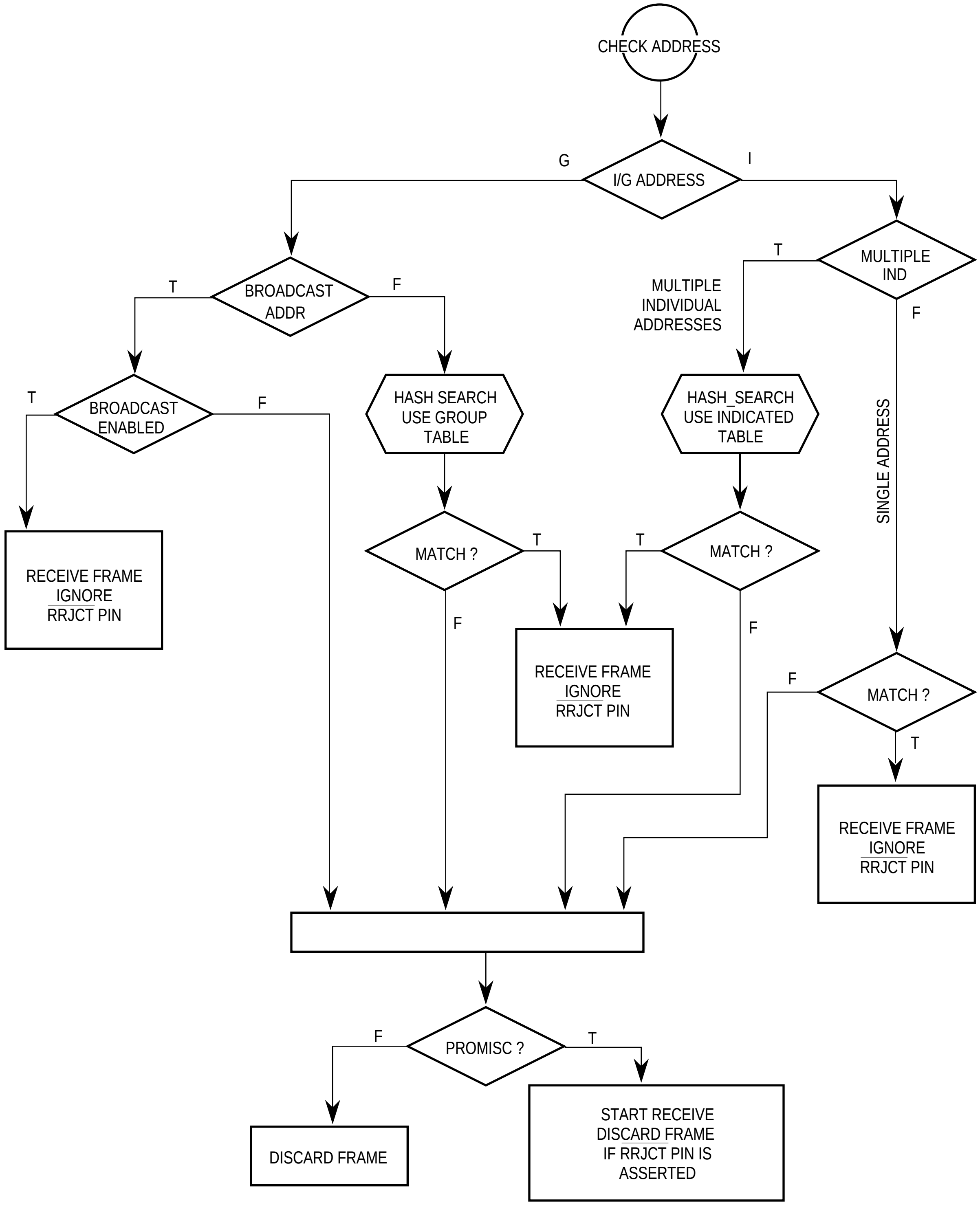

CHECK ADDRESS
I/G ADDRESS
G
I
BROADCAST ADDR
T
F
MULTIPLE IND
T
F
MULTIPLE INDIVIDUAL ADDRESSES
SINGLE ADDRESS
BROADCAST ENABLED
T
F
HASH SEARCH USE GROUP TABLE
HASH_SEARCH USE INDICATED TABLE
RECEIVE FRAME IGNORE RRJCT PIN
MATCH ?
T
F
MATCH ?
T
F
RECEIVE FRAME IGNORE RRJCT PIN
MATCH ?
F
T
RECEIVE FRAME IGNORE RRJCT PIN
PROMISC ?
F
T
DISCARD FRAME
START RECEIVE DISCARD FRAME IF RRJCT PIN IS ASSERTED

Figure 7-70. Ethernet Address Recognition Flowchart

7.10.23.12 HASH TABLE ALGORITHM. The hash table process used in the individual and group hash filtering operates as follows. The Ethernet controller maps any 48-bit address

7.11.2 General SMC Mode Register (SMCMR)

The operating mode of each SMC port is defined by the 16-bit, memory-mapped, read-write SMCMR. See the specific SMC protocol for more information on this register.

7.11.3 SMC Buffer Descriptors

When the SMCs are configured to operate in GCI mode, the memory structure for the SMCs is pre-defined to be one word long for transmit and one word long for receive. These one-word structures are detailed later when the GCI operation is described in more detail.

However, in UART and transparent modes of operation, the SMCs have a memory structure that is like that of the SCCs. The data associated with the SMCs is stored in buffers. Each buffer is referenced by BD organized in a buffer descriptor ring located in the dual-port RAM (see Figure 7-74).

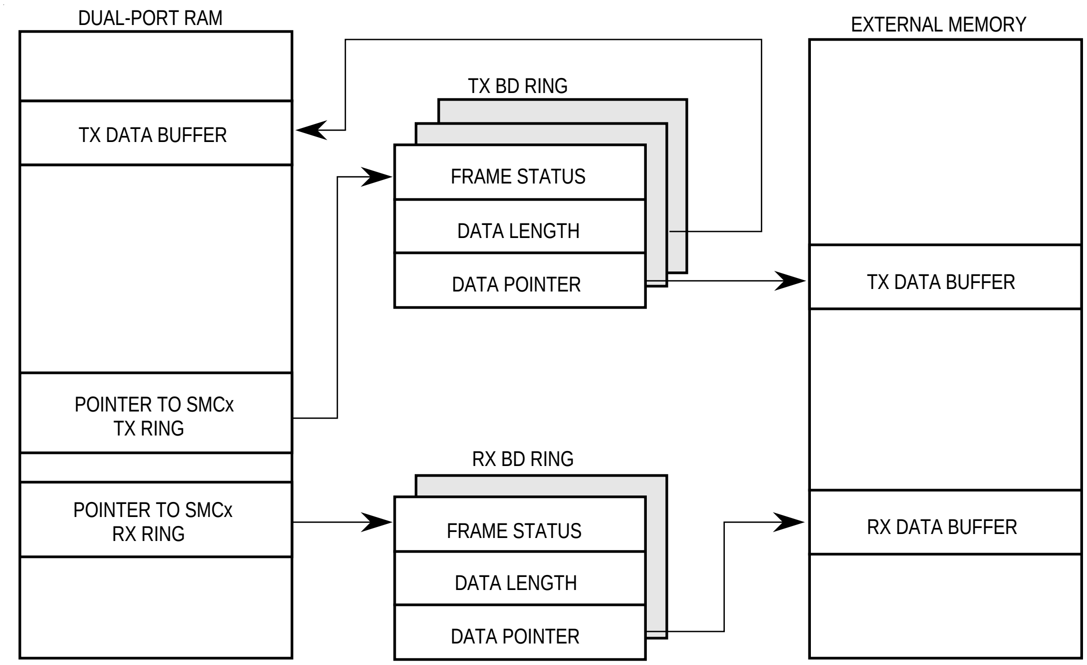


Figure 7-74. SMC Memory Structure

The BD ring allows the user to define buffers for transmission and buffers for reception. Each BD ring forms a circular queue. The CP confirms reception and transmission (or indicates error conditions) using the BDs to inform the processor that the buffers have been serviced.

The actual buffers may reside in either external memory or internal memory. Data buffers may reside in the parameter area of an SCC or SMC if that channel is not enabled.

7.11.4 SMC Parameter RAM

Each SMC parameter RAM area begins at the same offset from each SMC base area. The protocol-specific portions of the SMC parameter RAM are discussed in the specific protocol

7.11.7.5 SMC UART RECEPTION PROCESSING. When the CPU32+ core enables the SMC receiver in UART mode, it will enter hunt mode, waiting for the first character to arrive. Once the first character arrives, the first Rx BD is checked by the CP to see if it is empty. It then begins storing characters in the associated data buffer.

When the data buffer has been filled or the MAX_IDL timer has expired (assuming it was enabled), the SMC clears the E-bit in the BD and generates an interrupt if the I-bit in the BD is set. If the incoming data exceeds the length of the data buffer, the SMC will fetch the next BD in the table and, if it is empty, will continue to transfer data to this BD's associated data buffer.

If the CM bit is set in the Rx BD, the E-bit will not be cleared, allowing the associated data buffer to be overwritten automatically when the CP next accesses this data buffer.

7.11.7.6 SMC UART PROGRAMMING MODEL. An SMC configured as a UART uses the same data structure as in the other modes. The SMC UART data structure supports multi-buffer operation. The SMC UART allows the user to transmit break and preamble sequences. Overrun, parity, and framing errors are reported via the BDs. In its simplest form, the SMC UART can function in a character-oriented environment. Each character is transmitted with accompanying stop bits and parity (as configured by the user), and received into separate 1-byte buffers. Reception of each buffer may generate a maskable interrupt.

Many applications may want to take advantage of the message-oriented capabilities supported by the SMC UART by using linked buffers (in either receive or transmit). In this case, data is handled in a message-oriented environment; users can work on entire messages rather than operating on a character-by-character basis. A message may span several linked buffers. Each message can be both transmitted and received as a linked list of buffers without any intervention from the CPU32+, which achieves both ease in programming and significant savings in processor overhead.

In the message-oriented environment, the idle sequence is used as the message delimiter. The transmitter is able to generate an idle sequence before starting a new message, and the receiver is able to close a buffer upon detection of idle sequence.

7.11.7.7 SMC UART COMMAND SET. The following transmit and receive commands are issued to the CR.

7.11.7.7.1 Transmit Commands. The following paragraphs describe the SMC UART transmit commands.

STOP TRANSMIT Command. The channel STOP TRANSMIT command disables the transmission of characters on the transmit channel. If this command is received by the SMC UART controller during message transmission, transmission of that message is aborted. The SMC UART completes transmission of any data already transferred to its FIFO and shift register (up to two characters) and then stops transmitting data. The TBPTR is not advanced when this command is issued.

The SMC UART transmitter will transmit a programmable number of break sequences and then start to transmit idles. The number of break sequences (which may be zero) should be

SM—SMC Mode

00 = GCI or SCIT support
01 = Reserved
10 = UART
11 = Totally transparent operation (must be selected for SMC transparent operation)

DM—Diagnostic Mode

00 = Normal operation
01 = Local loopback mode
10 = Echo mode
11 = Reserved

TEN—SMC Transmit Enable

0 = SMC transmitter disabled
1 = SMC transmitter enabled

NOTES

Once the SMC transmit enable bit is cleared, the bit must not be reenabled for at least 3 serial clocks.

REN—SMC Receive Enable

0 = SMC receiver disabled
1 = SMC receiver enabled

7.11.10.11 SMC TRANSPARENT RECEIVE BUFFER DESCRIPTOR (RX BD). The CP reports information about the received data for each buffer using Rx BDs. The CP closes the current buffer, generates a maskable interrupt, and starts to receive data into the next buffer after one of the following events:

1. Detecting an overrun error
2. Detecting a full receive buffer
3. Issuing the ENTER HUNT MODE command

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET + 0	**E**	—	**W**	**I**	—	—	**CM**	—	—	—	—	—	—	—	OV	—
OFFSET + 2	DATA LENGTH															
OFFSET + 4	**RX DATA BUFFER POINTER**															
OFFSET + 6																

NOTE : Entries in boldface must be initialized by the user

E—Empty

0 = The data buffer associated with this Rx BD has been filled with received data, or data reception has been aborted due to an error condition. The CPU32+ core is free to examine or write to any fields of this Rx BD. The CP will not use this BD again while the E-bit remains zero.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET + 0	E	R							CHARACTER1							
OFFSET + 2	E	R							CHARACTER2							
OFFSET + 4	E	R							CHARACTER3							
OFFSET + 6																

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET + E	E	R							CHARACTER8							
OFFSET + 10	1	1							RCCM							
OFFSET +1 2									RCCR							

CHARACTER1-8—Control Character Values

These fields define control characters that should be compared to the incoming character. For less than 8 bits characters, the msb bits should be zero.

E—End of Table

0 = This entry is valid. The lower 8 bits will be checked against the incoming character.
1 = The entry is not valid. This must be the last entry in the Control Characters Table.

NOTE

In tables with 8 control characters this bit is always 0.

R—Reject character

0 = The character is not rejected but written into the receive buffer. The buffer is then closed and a new receive buffer is used if there is more data in the message. A maskable (I Bit in the Receive BD) interrupt is generated.
1 = If this character is recognized it will not be written to the receive buffer. Instead, it is written to the Received Control Characters Register (RCCR) and a maskable interrupt is generated. The current buffer is not closed when a control character is received with R set.

RCCM—Received Control Character Mask

The value in this register is used to mask the comparison of CHARACTER1 through CHARACTER8. The lower eight bits of RCCM correspond to the lower eight bits of CHARACTER1-8, and are decoded as follows.

0 = Mask this bit in the comparison of the incoming character, and CHARACTER1 through CHARACTER8.
1 = The address comparison on this bit proceeds normally. No masking takes place.

NOTE

The two most significant bits (bit 15 and bit 14) of RCCM must be set, or erratic operation may occur during the control character recognition process.

Table 8-1. Boundary Scan Control Bits

Name	Bit Number	Name	Bit Number	Name	Bit Number
g1.cntl	2	g10.cntl	32	pbhl.ctl	155
g2.cntl	5	add.cntl	59	pblh.ctl	159
g3.cntl	8	addh.cntl	83	pchh.ctl	170
g4.cntl	13	g11.cnt	89	pchl.ctl	174
g5.cntl	16	db.ctl	110	pclh.ctl	179
g6.cntl	18	pahh.ctl	131	g12.cntl	188
g7.cntl	23	pahl.ctl	136	g13.cntl	191
g8.cntl	27	palh.ctl	141	g14.cntl	194
g9.cntl	29	pbhh.ctl	150		

The boundary scan bit definitions are listed in Table 8-2.

The first column in the table defines the bit's ordinal position in the boundary scan register. The shift register cell nearest TDO (i.e., first to be shifted out) is defined as bit 0; the last bit to be shifted out is 195.

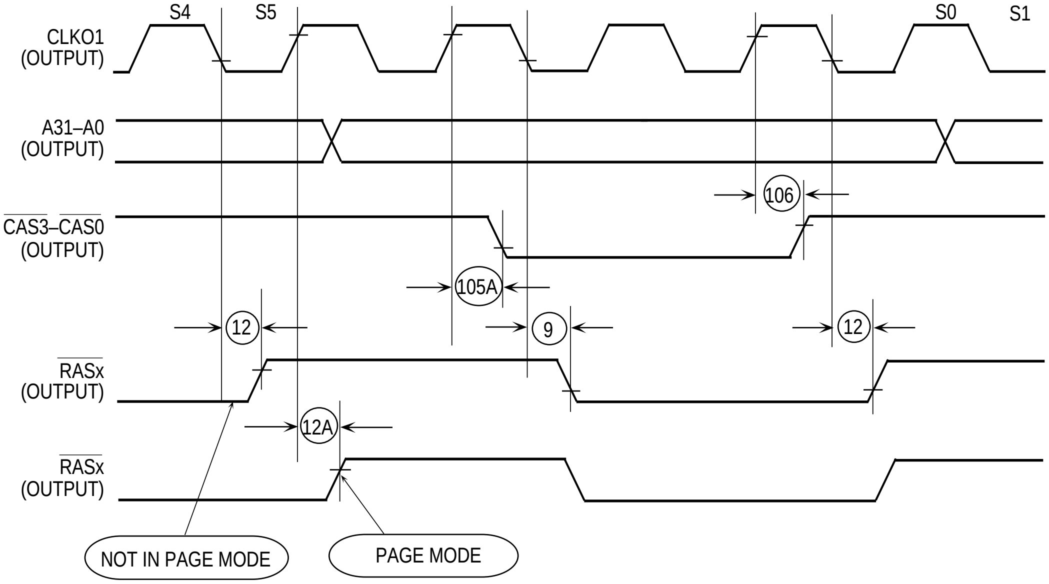
The second column references one of the four QUICC cell types depicted in Figure 8-3 through , which describe the cell structure for each type.

The third column lists the pin name for all pin-related cells or defines the name of bidirectional control register bits.

The fourth column lists the pin type for convenience, where TS-Output indicates a three-stateable output pin, I/O indicates a bidirectional pin, and OD-I/O denotes an open-drain bidirectional pin.

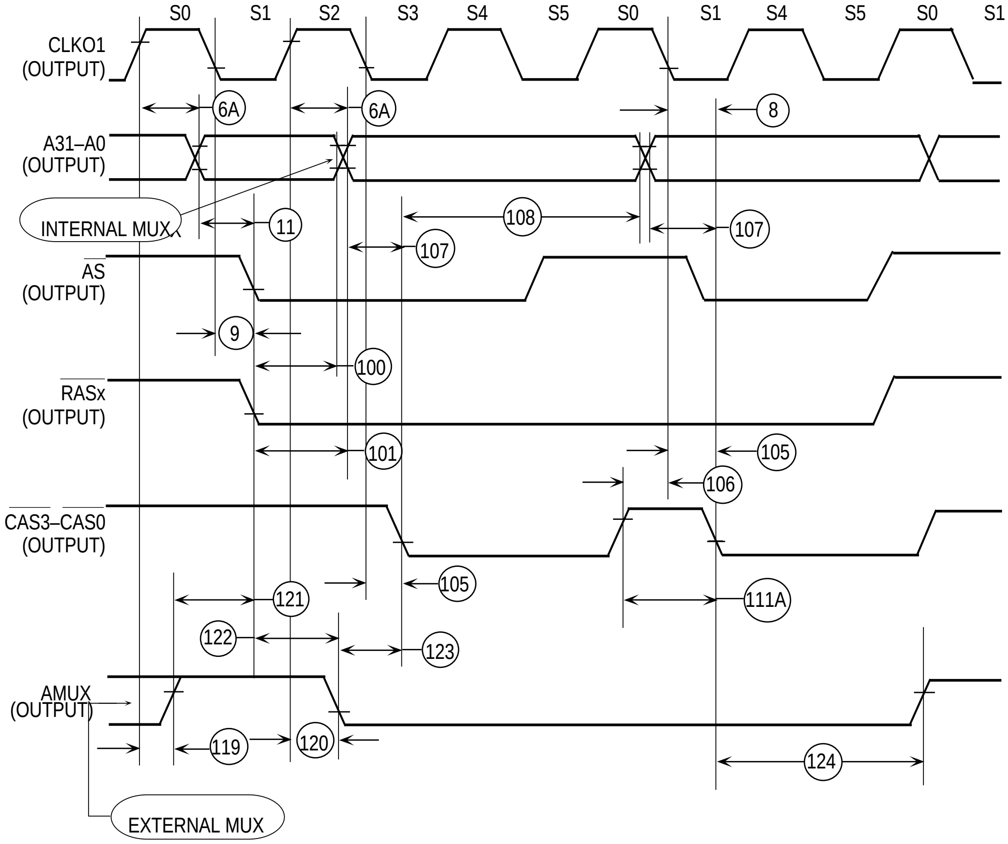
The last column indicates the associated boundary scan register control bit for bidirectional, three-state, and open-drain output pins.

Bidirectional pins include a single scan cell for data (IO.Cell) as depicted in Figure 8-6. These bits are controlled by the cell shown in Figure 8-5. The value of the control bit determines whether the bidirectional pin is an input or an output. One or more bidirectional data cells can be serially connected to a control cell as shown in Figure 8-7. Note that, when sampling the bidirectional data cells, the cell data can be interpreted only after examining the IO control cell to determine pin direction, and also note that the control cell captures the value of the following cell.



NOTE: All timing is shown with respect to 0.8-V and 2.0-V levels.

Figure 10-22. DRAM: Refresh Cycle



NOTE: All timing is shown with respect to 0.8-V and 2.0-V levels.

Figure 10-23. DRAM: Page-Mode—Page-Hit

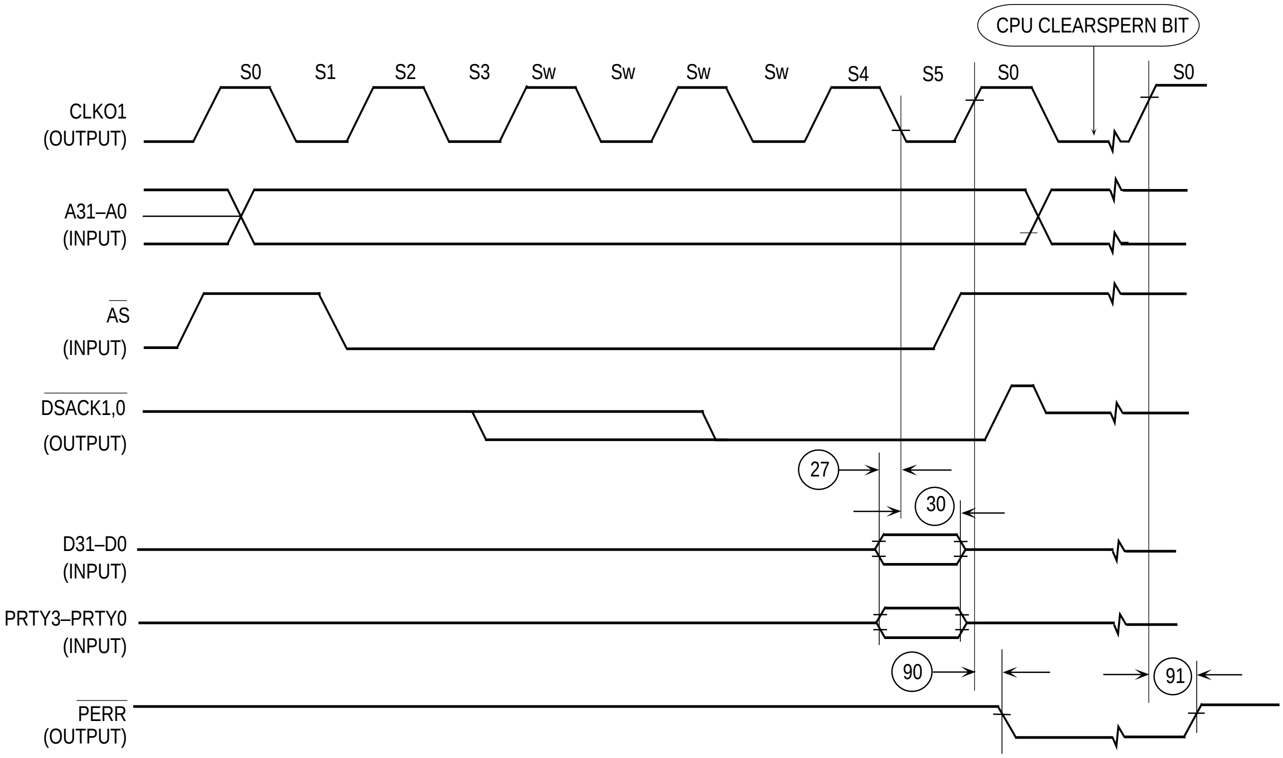


Figure 10-38. External MC68030/MC68360 Parity Bits Timing Diagram

10.15 040 BUS TYPE SLAVE MODE BUS ARBITRATION AC ELECTRICAL SPECIFICATIONS

(The electrical specifications in this document are preliminary. See Figure 10-39)

Num.	Characteristic	3.3 V or 5.0 V		5.0V		Unit
		25.0 MHz		33.34MHz		
		Min	Max	Min	Max	
231	Address, Transfer Attributes High Impedance to Clock High	7	—	6	—	ns
232[1]	Clock High to BG Low	—	20	—	15	ns
233	Clock High to BG High	4	20	4	15	ns
234	BB High to Clock High (040 output)	7	—	6	—	ns
235	BB High Impedance to Clock High (040 output)	0	—	0	—	ns
236	Clock High to BB Low (360 output)	—	20	—	15	ns
237	Clock High to BB High (360 output)	—	20	—	15	ns
238	Clock Low to BB High Impedance (360 output)	—	20	—	15	ns

NOTES:

1. BG remains low until either the SDMA or the IDMA requests the external bus.

11.3 PIN ASSIGNMENT—241-LEAD PIN GRID ARRAY (PGA)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
T	PA15	PA12	PA9	PA6	PA3	PA2	PB17	PB15	PB12	PB11	PB8	PB5	PB2	PC11	PC9	PC6	PC5	PC2
S	D2	D0	PA13	PA10	PA7	PA5	PA1	PB16	PB13	PB10	PB7	PB4	PB1	PC10	PC7	PC3	PC1	IRQ2
R	D4	D3	D1	PA14	PA11	PA8	PA4	PA0	PB14	PB9	PB6	PB3	PB0	PC8	PC4	PC0	IRQ3	IRQ1
Q	D7	D6	D5	GND	GND	GND	VCC	VCC	GND	GND	VCC	VCC	GND	GND	GND	IRQ5	BERR	RESETS
P	D10	D9	D8	GND	VCC	GND			NC				VCC	GND	GND	HALT	RMC	PERR
N	D13	D12	D11	GND	GND									GND	GND	AVEC	TDO	TMS
M	D16	D15	D14	GND											VCC	TD1	TCK	RESETH
L	D19	D18	D17	VCC											GND	TRST	BKPT	IRQ6
K	CLKO2	VCC	GND	VCCCLK	GNDCLK									VCC	VCC	IRQ4	BGACK	BG
J	CLKO1	D20	D22	GND	VCC									GND	GND	IFETCH	NC1	BR
H	D21	D23	D25	GND											GNDs2	NC2	BCLRO	OE
G	D24	D26	D28	VCC											VCC	IPIPE0	AS	IPIPE1
F	D27	D29	D31	GND	GND									VCC	GND	PRTY2	PRTY1	PRTY0
E	D30	FC3	FC0	A31	VCCSYN	GNDSYN			GND				VCC	GNDs1	VCC	NC3	DSACK1	PRTY3
D	FC2	FC1	A30	XFC	VCC	GND	GND	VCC	VCC	GND	GND	GND	GND	VCC	GND	CAS0	R/W	DSACK0
C	SIZ1	A29	EXTAL	MODCK1	A27	A23	A20	A17	A14	A8	A4	A0	CS7	CS4	CS1	CAS3	FREEZE	DS
B	SIZ0	A28	MODCK0	GND	A25	A22	A19	A16	A13	A10	A7	A5	A1	IRQ7	CS5	CS2	CAS2	CAS1
A		XTAL	NC3	A26	A24	A21	A18	A15	A12	A11	A9	A6	A3	A2	TRIS	CS6	CS3	CS0

MC68360/MC68EN360
(BOTTOM VIEW)

NOTE: Pin P9 "NC" is for guide purposes only.