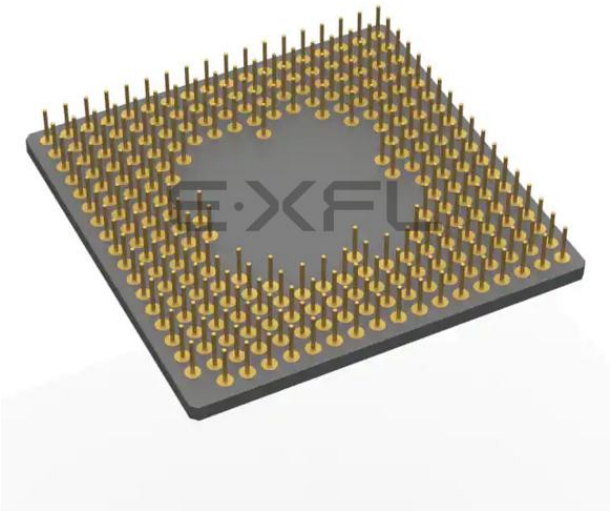


Welcome to E-XFL.COM

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	241-BEPGA
Supplier Device Package	241-PGA (47.24x47.24)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68360rc25l

Table 4-4. Address Offset Encoding

A1	A0	Offset
0	0	+0 Byte
0	1	+1 Byte
1	0	+2 Bytes
1	1	+3 Bytes

Table 4-5 lists the bytes required on the data bus for read cycles. The entries shown as OPx are portions of the requested operand that are read during that bus cycle and are defined by SIZ0, SIZ1, A0, and A1 for the bus cycle. Bytes labeled x are “don’t cares” and are not required during that read cycle.

Table 4-5. Data Bus Requirements for Read Cycles

Transfer Size	Size		Address		Long-Word Port External Data Bytes Required				Word Port External Data Bytes Required		Byte Port External Data Bytes Required
	SIZ1	SIZ0	A1	A0	D31:D24	D23:D16	D15:D8	D7:D0	D31:D24	D23:D16	D31:D24
Byte	0	1	0	0	OP3	x	x	x	OP3	x	OP3
	0	1	0	1	x	OP3	x	x	x	OP3	OP3
	0	1	1	0	x	x	OP3	x	OP3	x	OP3
	0	1	1	1	x	x	x	OP3	x	OP3	OP3
Word	1	0	0	0	OP2	OP3	x	x	OP2	OP3	OP2
	1	0	0	1	x	OP2	OP3	x	x	OP2	OP2
	1	0	1	0	x	x	OP2	OP3	OP2	OP3	OP2
	1	0	1	1	x	x	x	OP2	x	OP2	OP2
3 Bytes	1	1	0	0	OP1	OP2	OP3	x	OP1	OP2	OP1
	1	1	0	1	x	OP1	OP2	OP3	x	OP1	OP1
	1	1	1	0	x	x	OP1	OP2	OP1	OP2	OP1
	1	1	1	1	x	x	x	OP1	x	OP1	OP1
Long Word	0	0	0	0	OP0	OP1	OP2	OP3	OP0	OP1	OP0
	0	0	0	1	x	OP0	OP1	OP2	x	OP0	OP0
	0	0	1	0	x	x	OP0	OP1	OP0	OP1	OP0
	0	0	1	1	x	x	x	OP0	x	OP0	OP0

Table 4-6 lists the combinations of SIZ0, SIZ1, A0, and A1 and the corresponding pattern of the data transfer for write cycles from the internal multiplexer of the QUICC to the external data bus. Bytes labeled x are “don't care.”

Figure 4-4 shows the transfer of a long-word operand to a word port. In the first bus cycle, the QUICC places the four operand bytes on the external bus. Since the address is long-word aligned in this example, the multiplexer follows the pattern in the entry of Table 4-6 corresponding to SIZ0, SIZ1, A0, A1 = 0000. The port latches the data on bits D16–D31 of the data bus, asserts $\overline{\text{DSACK1}}$ ($\overline{\text{DSACK0}}$ remains negated), and the QUICC terminates the bus cycle. It then starts a new bus cycle with SIZ0, SIZ1, A0, A1 = 1010 to transfer the remaining

16 bits. SIZ0 and SIZ1 indicate that a word remains to be transferred; A0 and A1 indicate that the word corresponds to an offset of two from the base address. The multiplexer follows the pattern corresponding to this configuration of the size and address signals and places the two least significant bytes of the long word on the word portion of the bus (D16–D31). The bus cycle transfers the remaining bytes to the word-size port. Figure 4-5 shows the timing of the bus transfer signals for this operation.

Table 4-6. QUICC Internal to External Data Bus Multiplexer—Write Cycle

Transfer Size	Size		Address		External Data Bus Connection			
	SIZ1	SIZ0	A1	A0	D31:D24	D23:D16	D15:D8	D7:D0
Byte	0	1	0	0	OP3	x	x	x
	0	1	0	1	OP3	OP3	x	x
	0	1	1	0	OP3	x	OP3	x
	0	1	1	1	OP3	OP3	x	OP3
Word	1	0	0	0	OP2	OP3	x	x
	1	0	0	1	OP2	OP2	OP3	x
	1	0	1	0	OP2	OP3	OP2	OP3
	1	0	1	1	OP2	OP2	x	OP2
3 Bytes	1	1	0	0	OP1	OP2	OP3	x
	1	1	0	1	OP1	OP1	OP2	OP3
	1	1	1	0	OP1	OP2	OP1	OP2
	1	1	1	1	OP1	x	OP2	OP1
Long Word	0	0	0	0	OP0	OP1	OP2	OP3
	0	0	0	1	OP0	OP0	OP1	OP2
	0	0	1	0	OP0	OP1	OP0	OP1
	0	0	1	1	OP0	OP0	x	OP0

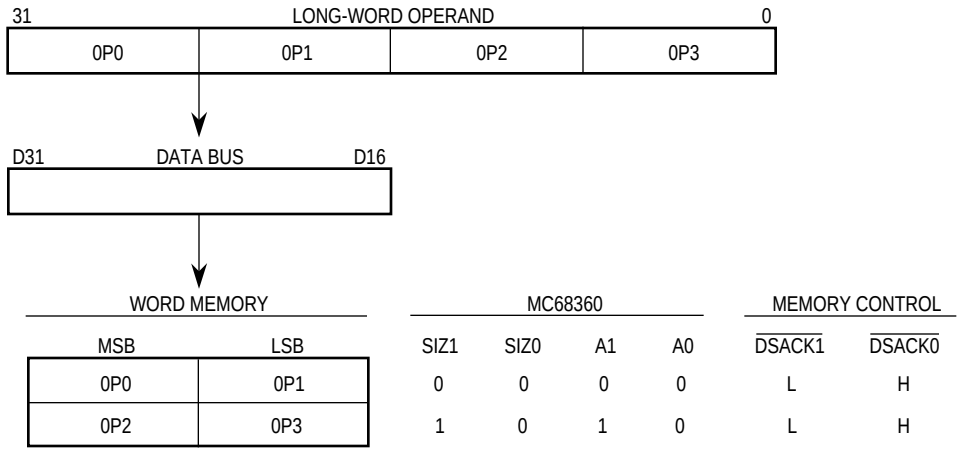


Figure 4-4. Example of Long-Word Transfer to Word Port

Once an external device receives the bus and asserts $\overline{\text{BGACK}}$, it should negate $\overline{\text{BR}}$. If $\overline{\text{BR}}$ remains asserted after $\overline{\text{BGACK}}$ is asserted, the QUICC assumes that another device is requesting the bus and prepares to issue another $\overline{\text{BG}}$.

4.6.4 Bus Arbitration Control

The bus arbitration control unit in the QUICC is implemented with a finite state machine. As discussed previously, all asynchronous inputs to the QUICC are internally synchronized in a maximum of two cycles of the clock. As shown in Figure 4-37, input signals labeled R and A are internally synchronized versions of $\overline{\text{BR}}$ and $\overline{\text{BGACK}}$, respectively. The $\overline{\text{BG}}$ output is labeled G, and the internal high-impedance control signal is labeled T. If T is true, the address, data, and control buses are placed in the high-impedance state after the next rising edge following the negation of $\overline{\text{AS}}$ and $\overline{\text{RMC}}$. All signals are shown in positive logic (active high), regardless of their true active voltage level. The state machine shown in Figure 4-37 does not have a state 1 or state 4.

State changes occur on the next rising edge of the clock after the internal signal is valid. The $\overline{\text{BG}}$ signal transitions on the rising edge of the clock after a state is reached during which G changes. The bus control signals (controlled by T) are driven by the QUICC immediately following a state change, when bus mastership is returned to the QUICC. State 0, in which G and T are both negated, is the state of the bus arbiter while the QUICC is bus master. R and A keep the arbiter in state 0 as long as they are both negated.

The QUICC does not allow arbitration of the external bus during the $\overline{\text{RMC}}$ sequence. For the duration of this sequence, the QUICC ignores the $\overline{\text{BR}}$ input. If mastership of the bus is required during an $\overline{\text{RMC}}$ operation, $\overline{\text{BERR}}$ must be used to abort the $\overline{\text{RMC}}$ sequence.

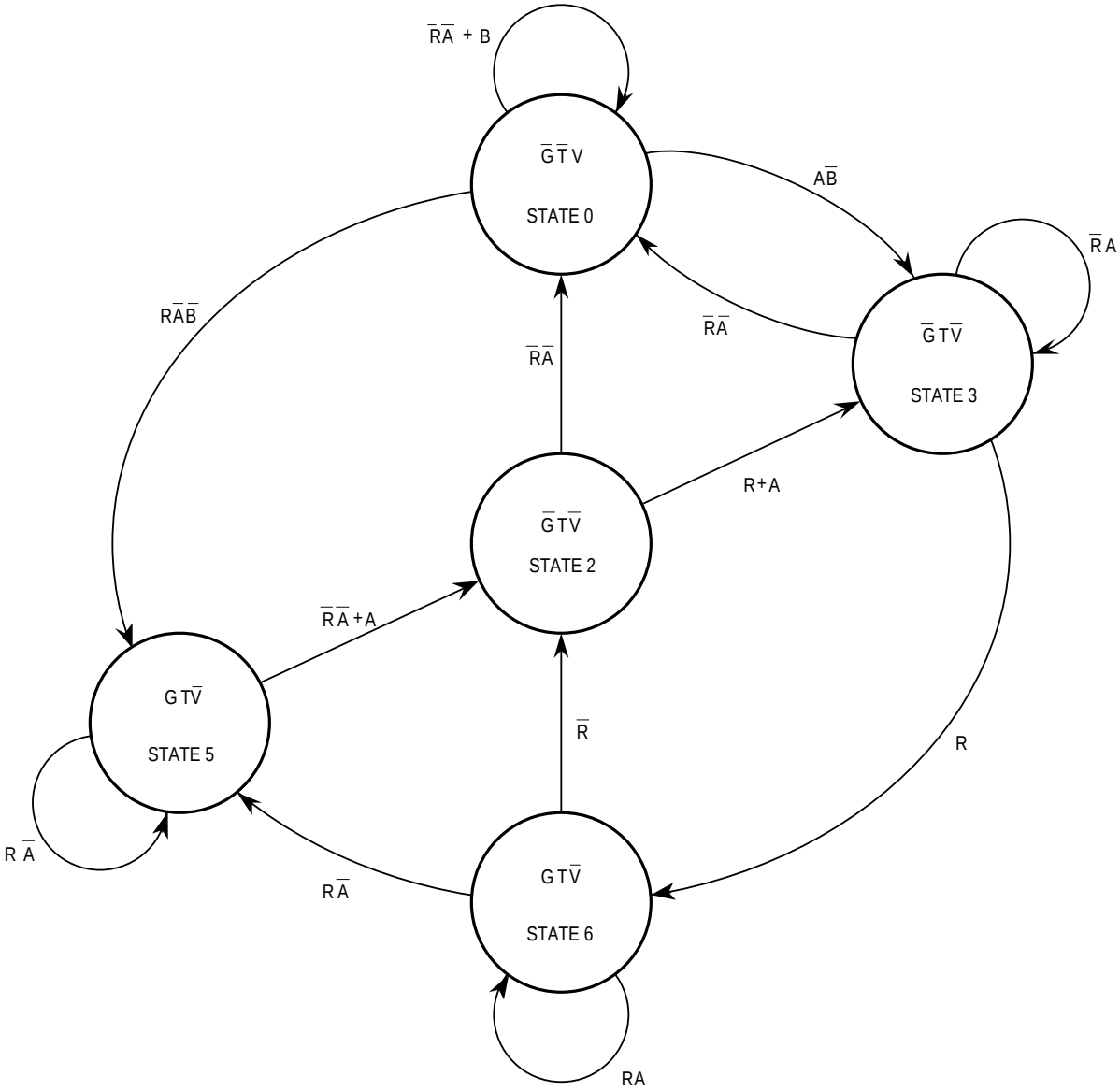


Figure 4-37. Bus Arbitration State Diagram

4.6.5 Slave (Disable CPU32+) Mode Bus Arbitration

When configured in the slave mode, the QUICC follows the bus arbitration mechanism described in 4.6 Bus Arbitration. When acting as one or more of the QUICC internal masters (refresh cycles, IDMA, and SDMA), the QUICC will output the $\overline{BR}$ signal. Systems that include several devices that can become bus master require external circuitry to assign priorities to the devices, so that when two or more external devices attempt to become bus

Table 5-10. Program Control Operations

Instruction	Operand Syntax	Operand Size	Operation
Conditional			
Bcc	⟨label⟩	8, 16, 32	If condition true, then PC + d ⇒ PC
DBcc	Dn, ⟨label⟩	16	If condition false, then Dn – 1 ⇒ PC; if Dn ≠ (– 1), then PC + d ⇒ PC
Scc	⟨ea⟩	8	If condition true, then destination bits are set to 1; else destination bits are cleared to 0
Unconditional			
BRA	⟨label⟩	8, 16, 32	PC + d ⇒ PC
BSR	⟨label⟩	8, 16, 32	SP – 4 ⇒ SP; PC ⇒ (SP); PC + d ⇒ PC
JMP	⟨ea⟩	none	Destination ⇒ PC
JSR	⟨ea⟩	none	SP – 4 ⇒ SP; PC ⇒ (SP); destination ⇒ PC
NOP	none	none	PC + 2 ⇒ PC
Returns			
RTD	#⟨d⟩	16	(SP) ⇒ PC; SP + 4 + d ⇒ SP
RTR	none	none	(SP) ⇒ CCR; SP + 2 ⇒ SP; (SP) ⇒ PC; SP + 4 ⇒ SP
RTS	none	none	(SP) ⇒ PC; SP + 4 ⇒ SP

To specify conditions for change in program control, condition codes must be substituted for the letters "cc" in conditional program control opcodes. Condition test mnemonics are given below. Refer to 5.3.3.10 Condition Tests for detailed information on condition codes.

—CC — Carry clear
—CS — Carry set
—EQ — Equal
—F — False*
—GE — Greater or equal
—GT — Greater than
—HI — High
—LE — Less or equal
LS — Low or same
LT — Less than
MI — Minus
NE — Not equal
PL — Plus
T — True
VC — Overflow clear
VS — Overflow set
—*Not applicable to the Bcc instruction

5.3.3.9 SYSTEM CONTROL INSTRUCTIONS. Privileged instructions, trapping instructions, and instructions that use or modify the CCR provide system control operations. All of these instructions cause the processor to flush the instruction pipeline. Table 5-11 summarizes the instructions. The preceding list of condition tests also applies to the TRAPcc instruction. Refer to 5.3.3.10 Condition Tests for detailed information on condition codes.

the bus is not being monitored. Each method requires a slightly different serial logic design to avoid spurious serial clocks.

Figure 5-24 represents the timing required for asserting $\overline{\text{BKPT}}$ during a single bus cycle.

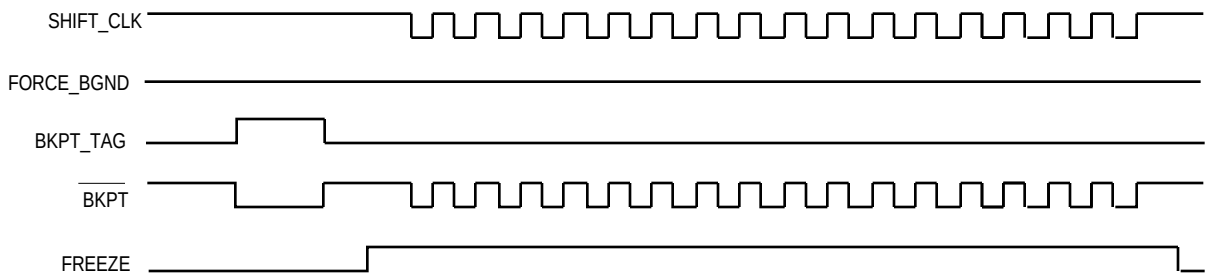


Figure 5-24. $\overline{\text{BKPT}}$ Timing for Single Bus Cycle

Figure 5-25 depicts the timing of the $\overline{\text{BKPT}}$/FREEZE method. In both cases, the serial clock is left high after the final shift of each transfer. This technique eliminates the possibility of accidentally tagging the prefetch initiated at the conclusion of a BDM session. As mentioned previously, all timing within the CPU is derived from the rising edge of the clock; the falling edge is effectively ignored.

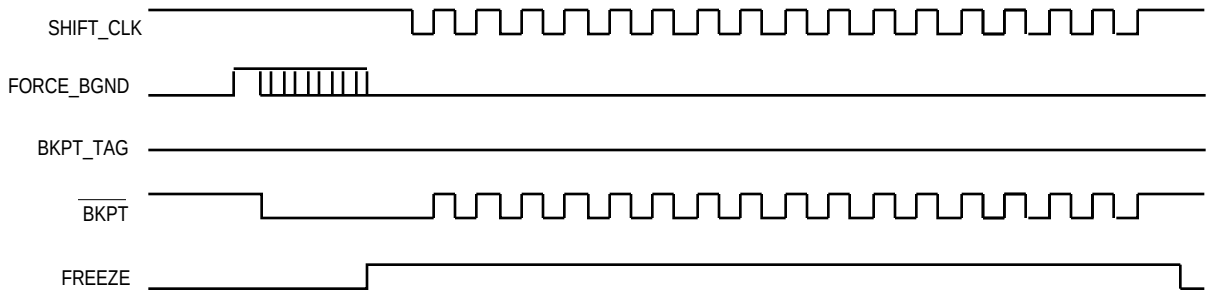


Figure 5-25. $\overline{\text{BKPT}}$ Timing for Forcing BDM

Figure 5-26 represents a sample circuit providing for both $\overline{\text{BKPT}}$ assertion methods. As the name implies, FORCE_BGND is used to force a transition into BDM by the assertion of $\overline{\text{BKPT}}$. FORCE_BGND can be a short pulse or can remain asserted until FREEZE is asserted. Once asserted, the set-reset latch holds $\overline{\text{BKPT}}$ low until the first SHIFT_CLK is applied.

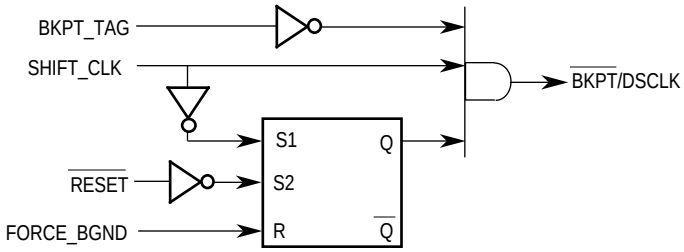


Figure 5-26. $\overline{\text{BKPT}}$/DSCLK Logic Diagram

The IBPTR entry points to the next BD that the IDMA will transfer data to when it is in IDLE state or points to the current BD during transfer processing. After a reset or when the end of an IDMA BD table is reached, the CP initializes this pointer to the value programmed in the IBASE entry.

ISTATE and ITEMP are for RISC use only.

7.6.4.2.2 IDMA Buffer Descriptors (BDs). Source addresses, destination addresses, and byte counts are presented to the RISC controller using special IDMA BDs. The RISC controller reads the BDs, programs the IDMA channel, and notifies the CPU32+ about the completion of a buffer transfer using the IDMA BDs. This concept is like that used for the serial channels on the QUICC, except that the BD is larger to contain additional information.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET + 0	V	—	W	I	L	—	CM	—	—	—	—	—	—	SE	DE	DA
OFFSET + 2																
OFFSET + 4	**DATA LENGTH**															
OFFSET + 6																
OFFSET + 8	**SOURCE DATA BUFFER POINTER**															
OFFSET + A																
OFFSET + C	**DESTINATION DATA BUFFER POINTER**															
OFFSET + E																

NOTE: Entries in boldface must be initialized by the user.

The following bits are prepared by the user before transfer and are set by the RISC controller after the buffer has been transferred.

V—Valid

0 = The data buffers associated with this BD are not currently ready for transfer. The user is free to manipulate this BD or its associated data buffer. When it is not in auto buffer mode, the RISC controller clears this bit after the buffer has been transferred (or after an error condition is encountered).

1 = The data buffers have been prepared for transfer by the user. (Note that only one data buffer needs to be prepared if the source/destination is a peripheral device.) It may be only the source data buffer when the destination is a device or the destination data buffer when the source is a device. No fields of this BD may be written by the user once this bit is set.

NOTE

The only difference between auto buffer mode and buffer chaining mode is that the V-bit is not cleared by the RISC controller in the auto buffer mode. Auto buffer mode is enabled by the CM bit.

register refers to the PSMR of the SCC when that SCC is configured for BISYNC mode. This register is cleared at reset. Some of the PSMR bits can be modified on the fly (i.e., while the receiver and transmitter are enabled).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NOS				CRC		RBCS	RTR	RVD	DRT	—		RPM		TPM	

NOS—Minimum Number of SYNCs Between or Before Messages

If NOS3–NOS0 = 0000, then 1 SYN1–SYN2 pair will be transmitted; if NOS3–NOS0 = 1111, then 16 SYN1–SYN2 pairs will be transmitted. The SYN1–SYN2 pair is defined in the DSR. The entire SYN1–SYN2 pair will always be transmitted regardless of the setting of the SYNL bits in the GSMR. The NOS bits may be modified on the fly.

CRC—CRC Selection

00 = Reserved.
01 = CRC16 (BISYNC). (X16 + X15 + X2 + 1). The PRCRC and PTCRC registers should be initialized to a preset value of all zeros or all ones before the channel is enabled. In both cases, the transmitter sends the calculated CRC non-inverted, and the receiver checks the CRC against zero. Eight-bit data characters (without parity) are configured when CRC16 is chosen.
10 = Reserved
11 = LRC (sum check). (BISYNC). For even LRC, the PRCRC and PTCRC registers should be initialized to zero before the channel is enabled. For odd LRC, the PR-CRC and PTCRC registers should be initialized to ones.
The receiver will check character parity when BCS is programmed to LRC and the receiver is not in transparent mode. The transmitter will transmit character parity when BCS is programmed to LRC and the transmitter is not in transparent mode. Use of parity in BISYNC assumes the use of 7-bit data characters.

RBCS—Receive Block Check Sequence

The BISYNC receiver internally stores two BCS calculations with a byte delay (eight serial clocks) between them. This enables the user to examine a received data byte and then decide whether or not it should be part of the BCS calculation. This is useful when control character recognition and stripping are to be performed in software. The bit should be set (or reset) within the time taken to receive the following data byte. When this bit is reset, the BCS calculations exclude the latest fully received data byte. When RBCS is set, the BCS calculations continue normally.

0 = Disable receive BCS
1 = Enable receive BCS

RTR—Receiver Transparent Mode

0 = The receiver is placed in normal mode with SYNC stripping and control character recognition operative.
1 = The receiver is placed in transparent mode. SYNCs, DLEs, and control characters are only recognized after a leading DLE character. The receiver will calculate the CRC16 sequence, even if it is programmed to LRC while in transparent mode. PR-CRC should be initialized to the CRC16 preset value before setting this bit.

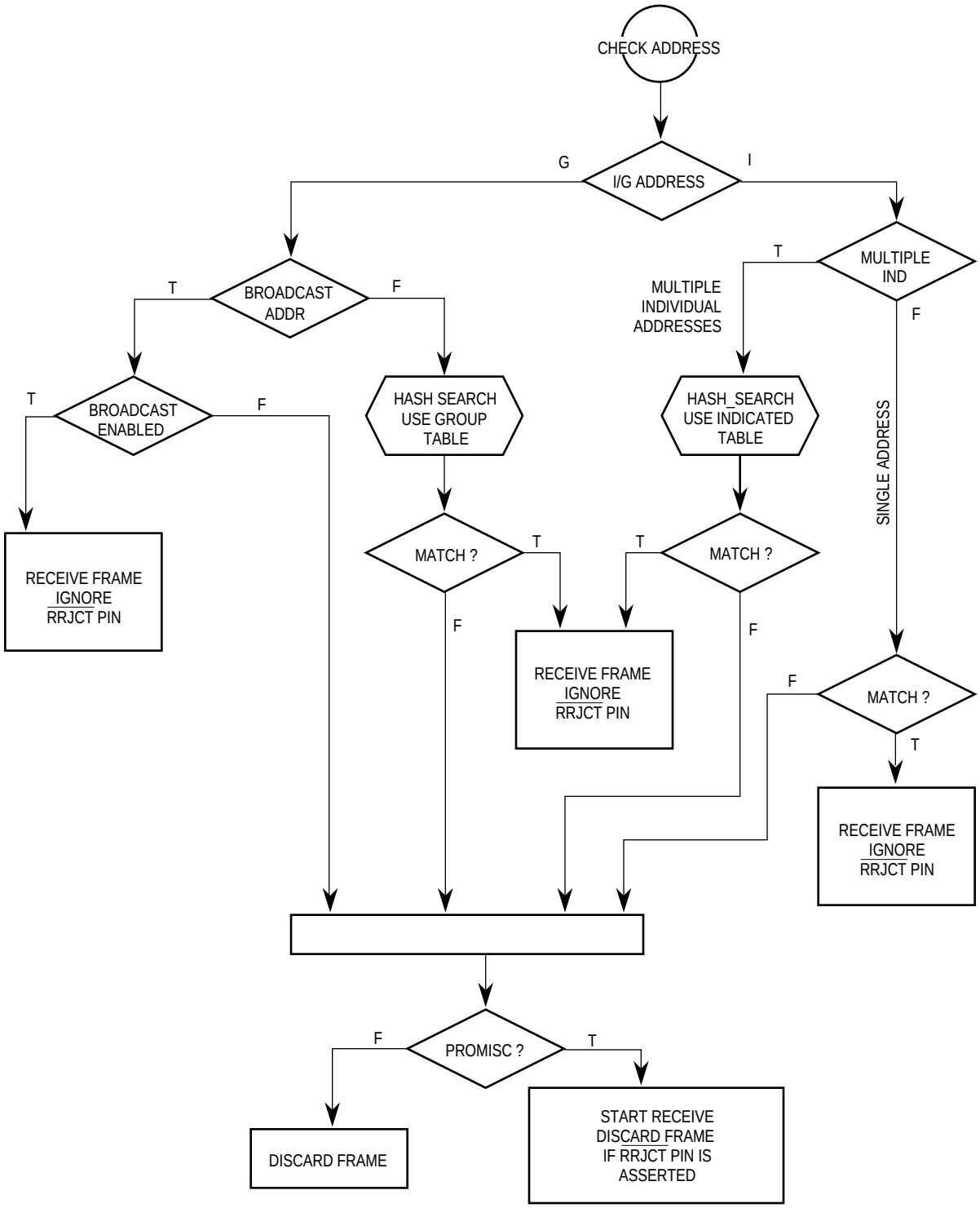


Figure 7-70. Ethernet Address Recognition Flowchart

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

into one of 64 bins. The 64 bins are represented by 64 bits stored in GADDR1–4 or IADDR1–4.

When the SET GROUP ADDRESS command is executed, the Ethernet controller maps the selected 48-bit address into one of the 64 bits. This is performed by passing the 48-bit address through the on-chip 32-bit CRC generator and selecting 6 bits of the CRC-encoded result to generate a number between 1 and 64. Bits 31–30 of the CRC result select one of the four GADDRs or IADDRs, and bits 29–26 of the CRC result select the bit within the selected register.

When a frame is received by the Ethernet controller, the same process is used. If the CRC generator selects a bit that is set in the group/individual hash table, the frame is accepted; otherwise, it is rejected. The result is that if eight group addresses are stored in the hash table, and random group addresses are received, the hash table prevents roughly 56/64 (or 87.5%) of the group address frames from reaching memory. Those that do reach memory must be further filtered by the processor to determine if they truly contain one of the eight desired addresses.

Better performance is achieved by using the group hash table and individual hash table at the same time. For instance, if eight group and eight physical addresses are stored in their respective hash tables, 87.5% of all frames (not just group address frames) are prevented from reaching memory.

The effectiveness of the hash table declines as the number of addresses increases. For instance, with 128 addresses stored in a 64-bin hash table, the vast majority of the hash table bits will be set, preventing only a small fraction of the frames from reaching memory. In such instances, an external CAM is advised if the extra bus utilization cannot be tolerated. See 7.10.23.7 CAM Interface for more details.

NOTE

The hash tables cannot be used to reject frames that match a set of entered addresses because unintended addresses will be mapped to the same bit in the hash table. Thus, an external CAM must be used to implement this function.

7.10.23.13 INTERPACKET GAP TIME. The minimum interpacket gap time for back-to-back transmission is 9.6 µs. The receiver can receive back-to-back frames with this minimum spacing. In addition, after the backoff algorithm, the transmitter will wait for carrier sense to be negated before retransmitting the frame. The retransmission will begin 9.6 µs after carrier sense is negated if carrier sense stays negated for at least 6.4 µs.

7.10.23.14 COLLISION HANDLING. If a collision occurs during frame transmission, the Ethernet controller will continue the transmission for at least 32 bit times, transmitting a JAM pattern consisting of 32 ones. If the collision occurs during the preamble sequence, the JAM pattern will be sent after the end of the preamble sequence.

If a collision occurs within 64 byte times, the retry process is initiated. The transmitter will wait a random number of slot times. A slot time is 512 bit times (52 µs). If collision occurs

Table 7-18. Centronics Receiver Parameter RAM

Address	Name	Width	Description
PIP Base+12	R_CNT	Word	Rx Internal Byte Count
PIP Base+14	RTEMP	Long	Rx Temp
PIP Base+18	Res	Word	Reserved
PIP Base+1C	Res	Word	Reserved
PIP Base+20	Res	Word	Reserved
PIP Base+22	Res	Word	Reserved
PIP Base+24	Res	Long	Reserved
PIP Base+28	MAX_SL	Word	Maximum Silence period
PIP Base+2a	SL_CNT	Word	Silence counter
PIP Base+2c	CHARCTER1	Word	CONTROL character 1
PIP Base+2E	CHARCTER2	Word	CONTROL character 2
PIP Base+30	CHARCTER3	Word	CONTROL character 3
PIP Base+32	CHARCTER4	Word	CONTROL character 4
PIP Base+34	CHARCTER5	Word	CONTROL character 5
PIP Base+36	CHARCTER6	Word	CONTROL character 6
PIP Base+38	CHARCTER7	Word	CONTROL character 7
PIP Base+3A	CHARCTER8	Word	CONTROL character 8
PIP Base+3C	RCCM	Word	Receive Control Character Mask
PIP Base+3E	RCCR	Word	Receive Character Control Register

Certain parameter RAM values above (marked in **bold face**) need to be initialized by the user before the PIP is enabled; the others are initialized/written by the CP. Once initialized, most parameter RAM values will not need to be accessed in user software since most of the activity is centered around the transmit buffer descriptors, not the parameter RAM.

7.13.8.14 BUFFER DESCRIPTOR TABLE POINTER (RBASE). The RBASE entry defines the starting location in the dual-port RAM for the PIP receiver's set of buffer descriptors. This provides a great deal of flexibility in how BDs are partitioned. By programming the RBASE entry and by setting the "wrap" bit in the last BD, the user may select how many BDs to allocate for the receive function. The user must initialize RBASE before enabling the channel.

NOTE

.RBASE should contain a value that is divisible by 8.

7.13.8.15 CENTRONICS FUNCTION CODE REGISTER (CFCR). The FC entry contains the value that the user would like to appear on the function code pins (FC3-0) when the associated SDMA channel accesses memory. It also controls the byte ordering convention to be used in the transfers.

follow. (In embedded control applications, most users leave the CPU in supervisor mode permanently.)

Step 3: Write the VBR

Many users initialize the ROM to contain an initial exception vector table. If so, the user should write the CPU vector base register (VBR) to point to the starting location of the table. (The exception vector table defines where to find the routines that handle interrupts, bus errors, traps, etc.) Note, however, that exceptions will not be ready to be handled properly until the stack pointer points to addressable RAM.

Step 4: Write the MBAR

The module base address register (MBAR), which always exists at a fixed address, determines where the 8-Kbyte block of QUICC internal RAM and internal peripherals are to be mapped. To put the 8-Kbyte block at $700000, write $00700001 to the MBAR. The MBAR must be accessed before any other QUICC internal peripheral or RAM location. Remember that the MBAR must be written in a special way as described in the Section 6 System Integration Module (SIM60). (If multiple QUICCs exist in the system, it may be necessary to write the MBARE before writing MBAR.)

Step 5: Verify a Dual-Port RAM Location

First, verify that the MBAR address was programmed correctly. This can be done by testing one of the dual-port RAM locations. Write $33 and $CC to location $700000 and verify that these values can be correctly read. Location $700000 is the beginning of the QUICC internal RAM.

Step 6: Is This a Power-Up Reset?

Next, determine the cause of the reset from the reset status register (RSR). The RSR is normally cleared by the user after it is read (ones are written to RSR). If this is not a power-up reset, the user may wish to take actions other than the ones listed or notify the system of the cause of an unexpected reset to aid in debugging. Many of the following steps are only necessary after a power-on reset. See Section 4 Bus Operation for more details on the effects of the different types of resets.

Step 7: Deal with the Clock Synthesizer

The next step is to set up the clock synthesizer, which is located in the SIM60. The three registers that control the clock synthesizer are the CLKOCR, the PLLCR, and the CDVCR.

If a low-speed external crystal is used (such as 32 kHz or 4 MHz), multiply the QUICC clock frequency up to the desired speed (e.g., 25 MHz). If an external oscillator is used to provide the exact system frequency, then this step is not needed.

The clock synthesizer has other options, such as SyncCLK and BRGCLK dividers, as well as the ability to divide the general system clock frequency. These are used in low-power applications. At this time, leave all these options in their default conditions. These options should only be enabled when the rest of the application code is in a more stable state.

QUICC supports internally (e.g., the level of the SIM60 and the level of the CPM). If such a condition occurs, $\overline{\text{BERR}}$ will be asserted by the QUICC.

9.8.1.9 SOFTWARE WATCHDOG. If desired, the MC68EC030 can program the QUICC software watchdog to generate a level 7 interrupt or a system reset. In this application, the software watchdog is configured in software to generate a reset so that the breakpoint logic can use level 7 interrupts. No additional hardware is required because the connection between the reset pins of the QUICC and the MC68EC030 is already made.

9.8.1.10 PERIODIC INTERVAL TIMER. If desired, the MC68EC030 can use the periodic interval timer on the QUICC to generate a system interrupt, such as for a real-time kernel. No additional hardware is required for this function.

9.8.1.11 MC68EC030 CACHING CONFIGURATION. The MC68EC030 can cache or not cache data and program memory as desired. However, it is strongly advisable not to cache the data that is accessed by the QUICC serial channels because of the overhead incurred every time the cached data area is written.

9.8.1.12 DOUBLE BUS FAULT. In slave mode, the QUICC double bus fault monitor is not operational.

9.8.1.13 JTAG AND THREE-STATE. The QUICC provides JTAG ports, commonly known as JTAG. This interface uses five pins: TMS, TDI, TDO, TCK, and $\overline{\text{TRST}}$. TMS and TDI are left unconnected because they have internal pullups. The JTAG ports of both parts are disabled in this application; however, the capability could be easily added.

When the QUICC is in master mode, it provides a $\overline{\text{TRIS}}$ pin that allows all outputs on the device to be three-stated. In slave mode, this feature is not available since the QUICC is a peripheral of the system.

9.8.1.14 QUICC SERIAL PORTS. The functions on QUICC parallel I/O ports A, B, and C may be used as desired in this application and have no bearing on the MC68EC030 interface. However, any unused parallel I/O pins should be configured as outputs, so they are not left floating.

9.8.2 Memory Interfaces

In this application, a number of memory arrays have been developed for EPROM, flash EPROM, EEPROM, SRAM, and DRAM. Each memory interface can be attached to the system bus as desired.

One issue not discussed is the decision of whether external buffers are needed on the system bus. This issue depends on the number of memory arrays used in the design and possibly the layout (i.e., capacitance) of the system bus.

Another issue left to the user is the number of wait states used with each memory system. This depends on the memory speed, whether external buffers are used, and the loading on the system bus pins. (The QUICC provides capacitance de-rating figures to calculate the effect of more or less capacitance on the AC Timing Specifications.)

10.7 AC Electrical Specifications Control Timing

(GND = 0 Vdc, TA = 0 to 70°C; The electrical specifications in this document are preliminary; See Figure 10-2)

Num.	Characteristic	Symbol	3.3 V or 5.0 V		5.0 V		Unit
			25 MHz		33.34 MHz		
			Min	Max	Min	Max	
	System Frequency	f_{sys}	dc[1]	25.00		33.34	MHz
	Crystal Frequency	f_{XTAL}	25	6000	25	6000	kHz
	On-Chip VCO System Frequency	f_{sys}	20	50	20	67	MHz
	Start-up Time With external clock (oscillator disabled) or after changing the multiplication factor MF.	t_{pll}		2500			clks
	CLKO1–2 stability	Δ_{CLK}	TBD	TBD			%
1	CLKO1 Period	t_{cyc}	40	—	30	—	ns
1A	EXTAL Duty Cycle, MF	t_{dcyc}	40	60	40	60	%
1C	External Clock Input Period	t_{EXTcyc}	40	—	30	—	ns
2, 3	CLKO1 Pulse Width (Measured at 1.5v)	t_{CW1}	19	—	14	—	ns
2A, 3A	CLKO2 Pulse Width (Measured at 1.5v)	t_{CW2}	9.5	—	7	—	ns
4, 5	CLKO1 Rise and Fall Times (Full Drive)	t_{Crf1}	—	2	—	2	ns
4A, 5A	CLKO2 Rise and Fall Times (Full Drive)	t_{Crf2}	—	2	—	2	ns
5B	EXTAL to CLKO1 Skew—PLL enabled (MF<5)	t_{EXTP1}		a		a	ns
5C	EXTAL to CLKO2 Skew—PLL enabled (MF<5)	t_{EXTP2}		a		a	ns
5D	CLKO1 to CLKO2 Skew	t_{CSKW}		a		a	ns

Note: 1. Note that the minimum VCO frequency and the PLL default values put some restrictions on the minimum system frequency.

a: The following calculation should be used to determine the actual value for specifications 5B, 5C and 5D.

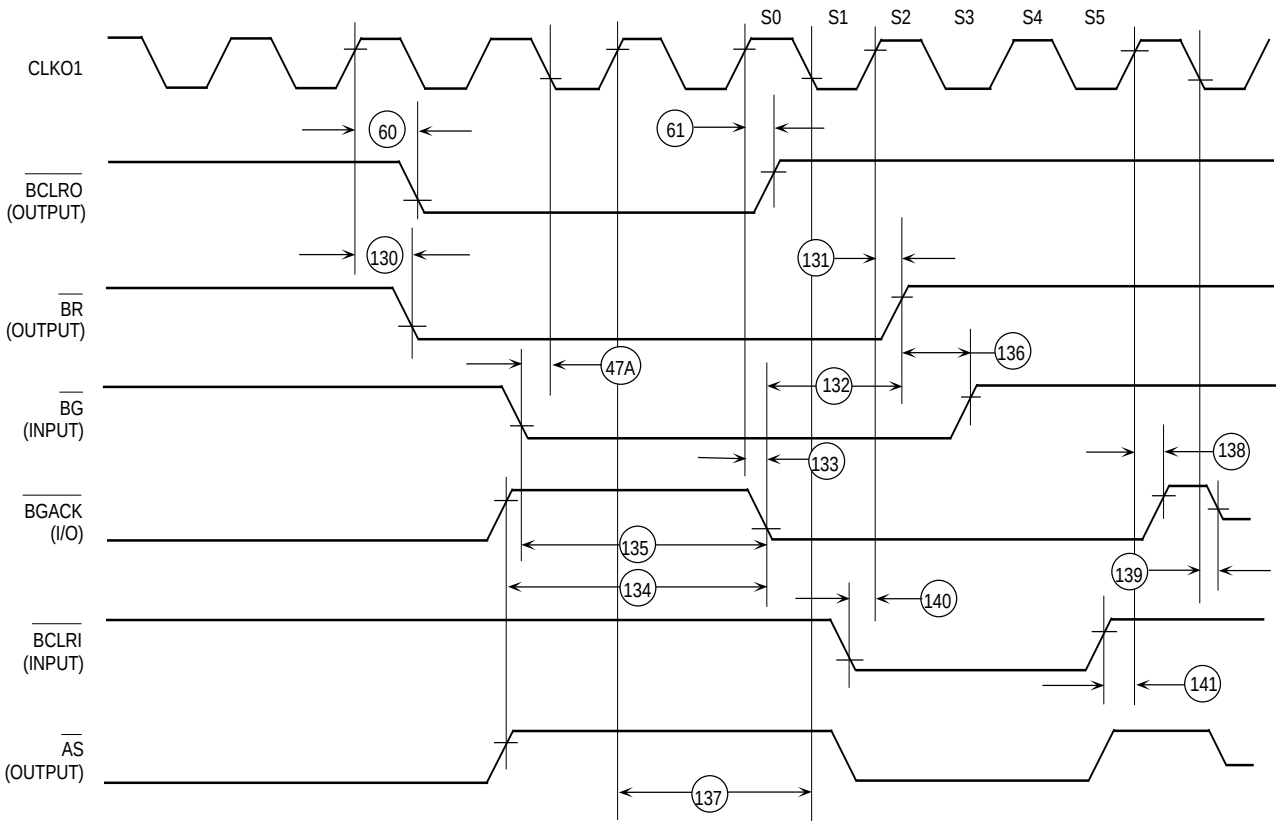
5B : 25Mhz +/-(0.9ns + 0.25 x (rise time)) (1.4ns@rise=2ns; 1.9ns@rise=4ns)

33Mhz +/-(0.5ns + 0.25 x (rise time)) (1ns@rise=2ns; 1.5ns@rise=4ns)

5C : 25/33Mhz +/-(2ns + 0.25 x (rise time)) (2.5ns@rise=2ns; 3ns@rise=4ns)

5D : 25Mhz +/-(3ns + 0.5 x (rise time)) (4ns@rise=2ns; 5ns@rise=4ns)

33Mhz +/-(2.5ns + 0.5 x (rise time)) (3.5ns@rise=2ns; 4.5ns@rise=4ns)



NOTE: Diagram does not apply to MC68040 companion mode.

Figure 10-25. MC68360 Slave Mode Asynchronous Arbitration

10.14 030/QUICC BUS TYPE SRAM/DRAM CYCLES AC ELECTRICAL SPECIFICATIONS

(The electrical specifications in this document are preliminary. See Figure 10-33–Figure 10-37)

Num.	Characteristic	3.3 V or 5.0 V 25.0 MHz		5.0V 33.34MHz		Unit
		Min	Max	Min	Max	
200	AS Low to OE Low (Read Cycle)	—	20	—	15	ns
201	AS High to OE High	—	20	—	15	ns
202	AS Low to WE Low (Write Cycle)	—	20	—	15	ns
203	AS High to WE High	—	20	—	15	ns
204	CLKO1 High to DSACK High	—	20	—	15	ns
205	AS Low to $\overline{CS}$ Low	—	22	—		ns
206	AS High to $\overline{CS}$ High	—	20	—	15	ns
207	AS Low to DSACK Low	—	27	—	22	ns
208	AS High to $\overline{CAS}$ High	—	20	—	15	ns
210	AS Low to BKPTO Low	—	25	—	23	ns
211	AS High to BKPTO High	—	30	—	25	ns
212	Data Valid to PRTY3–0 Valid	—	20	—	15	ns
213	Data Invalid to PRTY3–0 Invalid	5	—	3.75	—	ns
214	R/W valid to AS Low	0	—	0	—	ns
215	AS High to R/W Invalid	0	—	0	—	ns
216	AS Assert to Parity Driven	4	—	3	—	ns
217	AS Negate to Parity Invalid	4	20	—	15	ns

NOTES:
1Synchronous specifications above are valid only when BSTM=1.

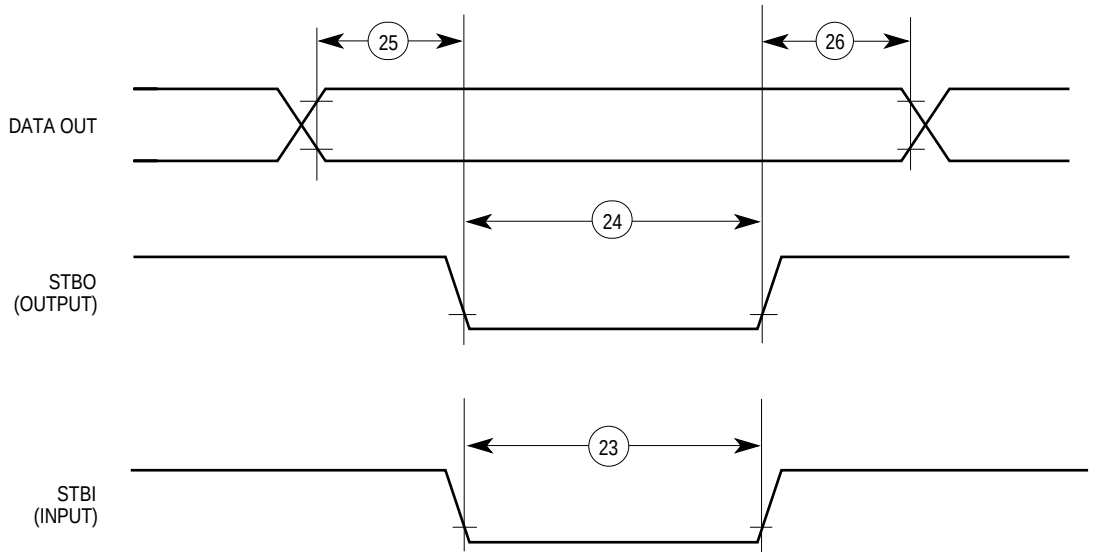


Figure 10-54. PIP Tx (Pulse Mode)

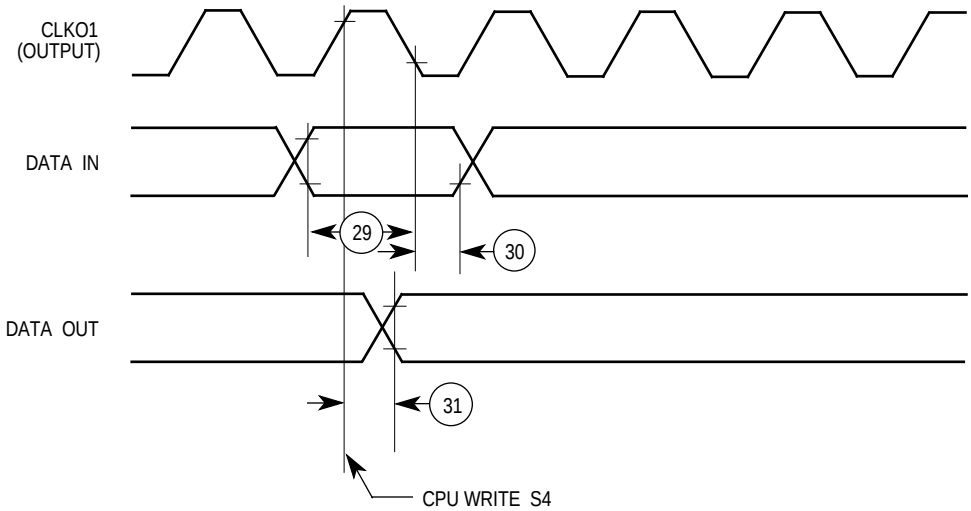


Figure 10-55. Parallel I/O Data-in/Data-Out Timing Diagram

10.20 INTERRUPT CONTROLLER AC ELECTRICAL SPECIFICATIONS

(The electrical specifications in this document are preliminary. See Figure 10-56 and Figure 10-57)

Num.	Characteristic	3.3 V or 5.0 V		5.0 V		Unit
		25.0 MHz		33.34 MHz		
		Min	Max	Min	Max	
35	Port C Interrupt Pulse Width Low (Edge Triggered Mode)	70	—	55	—	ns
36	Minimum Time Between Active edges PortC	70	—	55	—	clk
37	Clock High to IOUT Valid (Slave Mode)	—	20	—	17	ns
38	Clock High to RQOUT Valid (Slave Mode)	—	20	—	17	ns

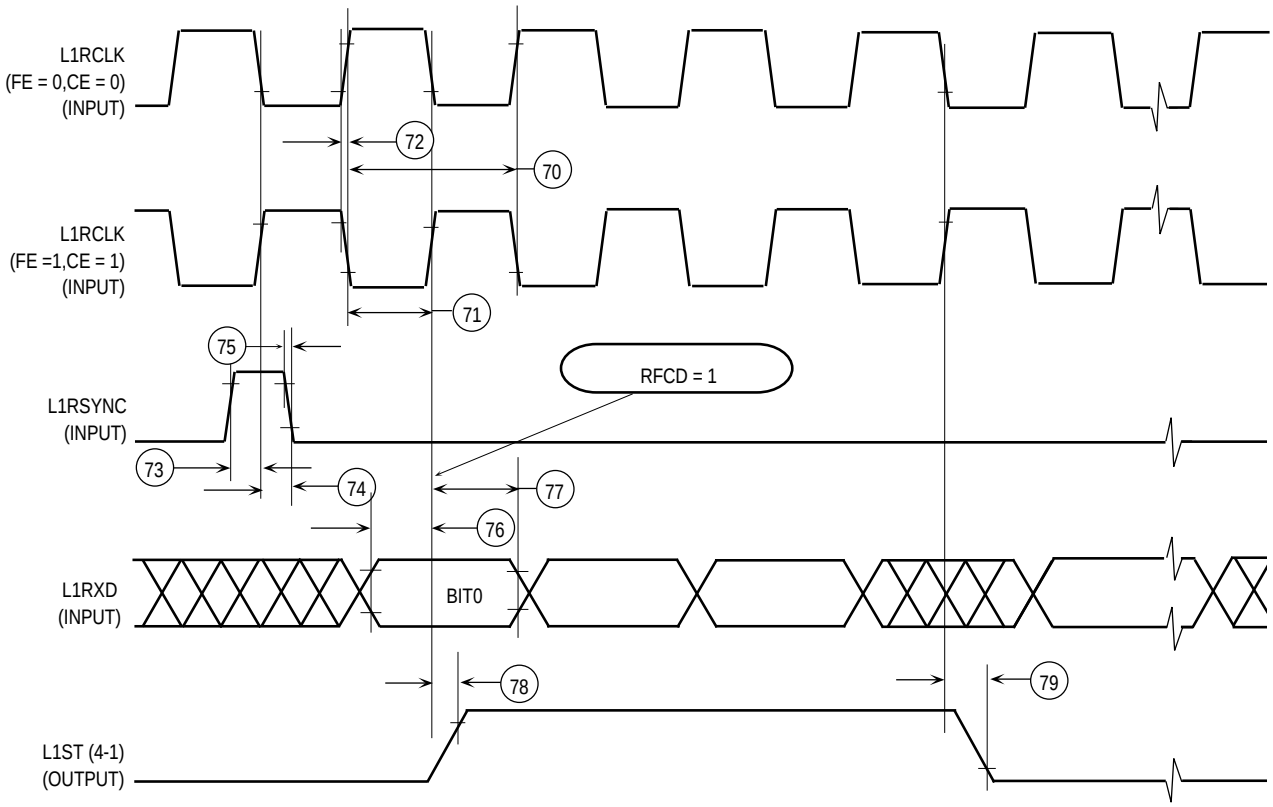


Figure 10-60. SI Receive Timing with Normal Clocking (DSC = 0)

10.24 SCC IN NMSI MODE—EXTERNAL CLOCK ELECTRICAL SPECIFICATIONS

(The electrical specifications in this document are preliminary. See Figure 10-65–Figure 10-67)

Num.	Characteristic	3.3 V or 5.0 V		5.0 V		Unit
		25.0 MHz		33.34 MHz		
		Min	Max	Min	Max	
100[1]	RCLK1 and TCLK1 Width High	CLKO1	—	CLKO1	—	
101	RCLK1 and TCLK1 Width Low	CLKO1 + 5nS	—	CLKO1 + 5nS	—	
102	RCLK1 and TCLK1 Rise/Fall Time	—	15	—	15	ns
103	TXD1 Active Delay (From TCLK1 Falling Edge)	0	50	0	50	ns
104	RTS1 Active/Inactive Delay (From TCLK1 Falling Edge)	0	50	0	50	ns
105	CTS1 Setup Time to TCLK1 Rising Edge	5	—	5	—	ns
106	RXD1 Setup Time to RCLK1 Rising Edge	5	—	5	—	ns
107[2]	RXD1 Hold Time from RCLK1 Rising Edge	5	—	5	—	ns
108	CD1 Setup Time to RCLK1 Rising Edge	5	—	5	—	ns

NOTES:
1.The ratio SyncCLK/RCLK1 and SyncCLK/TCLK1 must be greater or equal to 2.25/1
2.Also applies to $\overline{CD}$ and $\overline{CTS}$ hold time when they are used as an external sync signals.

10.25 SCC IN NMSI MODE—INTERNAL CLOCK ELECTRICAL SPECIFICATIONS

(The electrical specifications in this document are preliminary. See Figure 10-65–Figure 10-67)

Num.	Characteristic	3.3 V or 5.0 V		5.0 V		Unit
		25.0 MHz		33.34 MHz		
		Min	Max	Min	Max	
100[1]	RCLK1 and TCLK1 Frequency	0	8.3	0	11	MHz
102	RCLK1 and TCLK1 Rise/Fall Time	—	—	—	—	ns
103	TXD1 Active Delay (From TCLK1 Falling Edge)	0	30	0	30	ns
104	RTS1 Active/Inactive Delay (From TCLK1 Falling Edge)	0	30	40	—	ns
105	CTS1 Setup Time to TCLK1 Rising Edge	40	—	40	—	ns
106	RXD1 Setup Time to RCLK1 Rising Edge	40	—	0	—	ns
107[2]	RXD1 Hold Time from RCLK1 Rising Edge	0	—	40	—	ns
108	CD1 Setup Time to RCLK1 Rising Edge	40	—	0	30	ns

NOTES:
1.The ratio SyncCLK/RCLK1 and SyncCLK/TCLK1 must be greater or equal to 3/1
2.Also applies to $\overline{CD}$ and $\overline{CTS}$ hold time when they are used as an external sync signals.