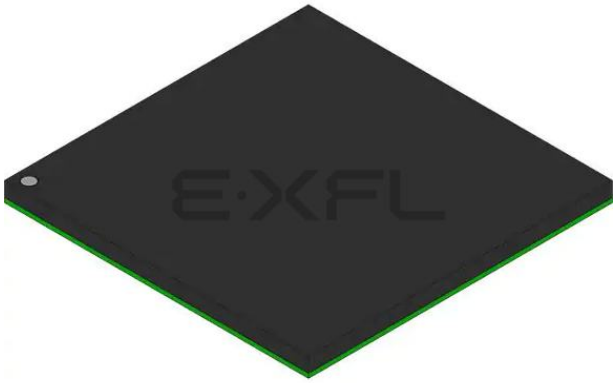


Welcome to E-XFL.COM

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	Active
Core Processor	CPU32+
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	33MHz
Co-Processors/DSP	Communications; CPM
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	10Mbps (1)
SATA	-
USB	-
Voltage - I/O	5V
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/pro/item?MUrl=&PartUrl=mc68360vr33l

1.2.1 CPU32+ Core

The CPU32+ core is a CPU32 that has been modified to connect directly to the 32-bit IMB and apply the larger bus width. Although the original CPU32 core had a 32-bit internal data path and 32-bit arithmetic hardware, its interface to the IMB was 16 bits. The CPU32+ core can operate on 32-bit external operands with one bus cycle. This allows the CPU32+ core to fetch a long-word instruction in one bus cycle and to fetch two word-length instructions in one bus cycle, filling the internal instruction queue more quickly. The CPU32+ core can also read and write 32-bits of data in one bus cycle.

Although the CPU32+ instruction timings are improved, its instruction set is identical to that of the CPU32. It will also execute the entire M68000 instruction set. It contains the same background debug mode (BDM) features as the CPU32. No new compilers, assemblers, or other software support tools need be implemented for the CPU32+; standard CPU32 tools can be used.

The CPU32+ delivers approximately 4.5 MIPS at 25 MHz, based on the standard (accepted) assumption that a 10-MHz M68000 delivers 1 VAX MIPS. If an application requires more performance, the CPU32+ can be disabled, allowing the rest of the QUICC to operate as an intelligent peripheral to a faster processor. The QUICC provides a special mode called MC68040 companion mode to allow it to conveniently interface to members of the M68040 family. This two-chip solution provides a 22-MIPS performance at 25 MHz.

The CPU32+ also offers automatic byte alignment features that are not offered on the CPU32. These features allow 16 or 32-bit data to be read or written at an odd address. The CPU32+ automatically performs the number of bus cycles required.

1.2.2 System Integration Module (SIM60)

The SIM60 integrates general-purpose features that would be useful in almost any 32-bit processor system. The term “SIM60” is derived from the QUICC part number, MC68360. The SIM60 is an enhanced version of the SIM40 that exists on the MC68340 and MC68330 devices.

First, new features, such as a DRAM controller and breakpoint logic, have been added. Second, the SIM40 was modified to support a 32-bit IMB as well as a 32-bit external system bus. Third, new configurations, such as slave mode and internal accesses by an external master, are supported.

Although the QUICC is always a 32-bit device internally, it may be configured to operate with a 16-bit data bus. Regardless of the choice of the system bus size, dynamic bus sizing is supported. Bus sizing allows 8-, 16-, and 32-bit peripherals and memory to exist in the 32-bit system bus mode and 8- and 16-bit peripherals and memory to exist in the 16-bit system bus mode.

Table 3-2. CPM Sub-Module Base Addresses

3	SMC1 Base	DPRBASE + $E80
4	SCC4 Base	DPRBASE + $F00
4	IDMA2 Base	DPRBASE + $F70
4	SMC2 Base	DPRBASE + $F80

3.3 INTERNAL REGISTERS MEMORY MAP

In addition to the internal dual-port RAM, there are a number of internal registers to support the functions of the various CPU32+ core peripherals. The internal registers (see Table 3-3 and Table 3-4) are memory-mapped registers offset from the register base (REGBASE) pointer. REGBASE (abbreviated REGB) = DPRBASE + 4K. All registers are located on the internal IMB.

NOTES

All registers that are underlined in the following tables are special registers called event registers. In these registers, bits are set by the QUICC and cleared by the user. To clear a bit, the user must write a one to that bit. For example, to clear bit 2 in SCCE1, the MOVE.B #$04,SCCE1 instruction may be used. Do NOT use read-modify-write instructions (such as BSET, BCLR, AND, OR, etc.) with these registers, or ALL bits in that register will inadvertently be cleared. See the individual register descriptions for more information.

All undefined and reserved bits within registers and parameter RAM values written by the user should be written with zero to allow for future enhancements to the device.

Bold letters mark registers that are restricted to supervisor access.

3.3.1 SIM Registers Memory Map

Table 3-3 lists the SIM registers memory map.

—CSR1 = $FF. Clear any CSR bits that are currently set.
—CMAR1 = $00. Disable interrupts for this example.
—CMR1 = $47A1. Internal maximum transfer rate; starts IDMA.

Bus Access #	Address (Hex)	Operation	No. Bytes	No. Bytes in DHR
1	$00000001	Read	1	1
2	$00000002	Read	2	3
3	$20000000	Write	2	1
4	$00000002	Write	1	0

Example 2. This more complicated example shows how packing is performed when the source and destination sizes are the same—long word. This example also shows the entire 7-byte DHR in use. The source address is $00000000, and the destination address is $20000003. The number of bytes to be transferred is 16.

IDMA channel 1 initialization required for this example:

- ICCR = $0720. Recommended normal configuration.
- FCR1 = $89. Source function code is 1000; destination function code is 1001.
- SAPR1 = $00000000. Source address.
- DAPR1 = $20000003. Destination address.
- BCR1 = $00000010. Byte transfer count.
- CSR1 = $FF. Clear any CSR bits that are currently set.
- CMAR1 = $00. Disable interrupts for this example.
- CMR1 = $4701. Internal maximum transfer rate; starts IDMA.

Bus Access #	Address (Hex)	Operation	No. Bytes	No. Bytes in DHR
1	$00000000	Read	4	4
2	$20000003	Write	1	3
3	$00000004	Read	4	7
4	$20000004	Write	4	3
5	$00000008	Read	4	7
6	$20000008	Write	4	3
7	$0000000C	Read	4	7
8	$2000000C	Write	4	3
9	$20000010	Write	2	1
10	$20000012	Write	1	0

Example 3. This example shows how packing operates when the source and destination sizes are different. The source address is $00000002, and the destination address is $20000002. The source size is long word, and the destination size is byte. The number of bytes to be transferred is 8.

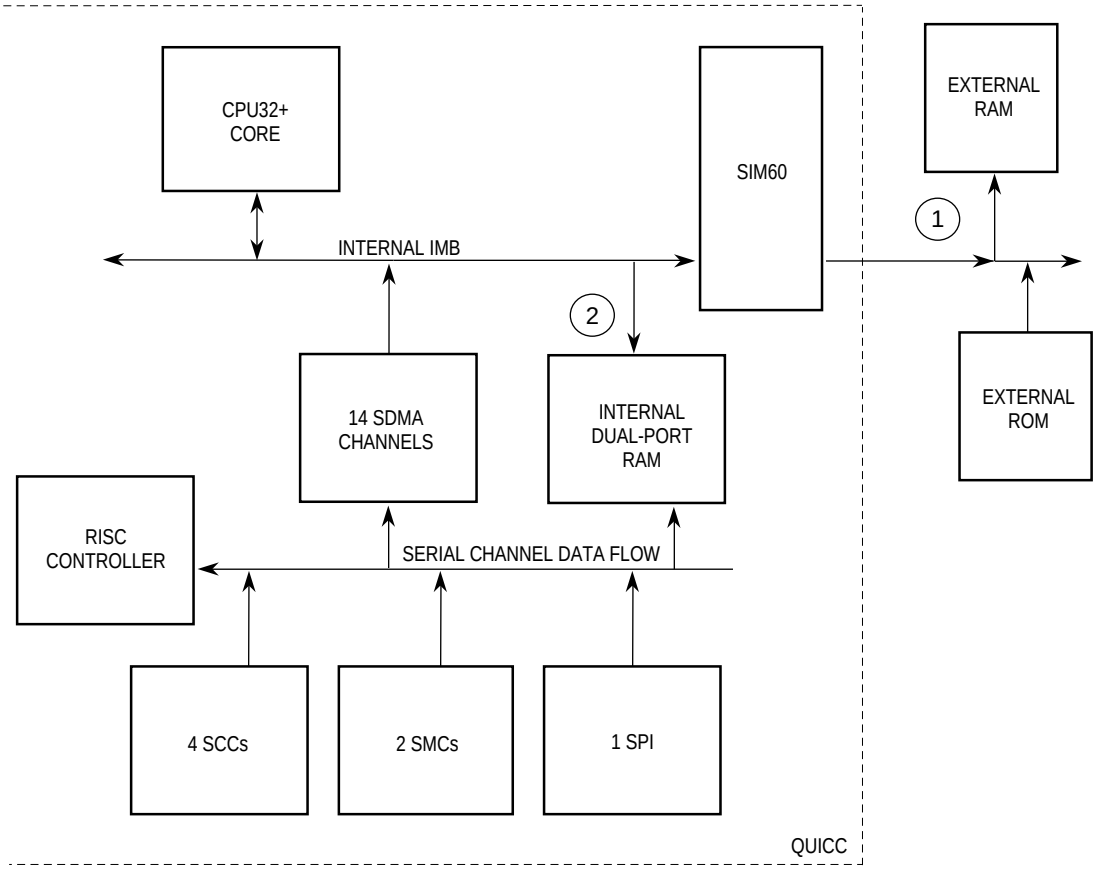


Figure 7-17. SDMA Data Paths

The read or write operation may take multiple bus cycles if the memory provides less than a 32-bit port size. For instance, a 32-bit long-word read from a 16-bit memory will take two SDMA bus cycles. As long as a higher priority bus master does not require the bus during an SDMA transfer, the entire operand (32 bits on reads and 8, 16, or 32 bits on writes) will be transferred in back-to-back bus cycles before the SDMA relinquishes the bus. If a higher priority bus master requests the bus during an operand transfer, it will be granted the bus at the end of that SDMA bus cycle. Once the higher priority bus master relinquishes the bus, the SDMA will reacquire the bus and continue any outstanding bus cycles.

The SDMA can steal cycles with no arbitration overhead when the QUICC is in master mode (i.e., the CPU32+ is enabled) and the external bus is not currently being held by an external master (see Figure 7-18). Note that in normal operation, the $\overline{BR}$, $\overline{BG}$, and $\overline{BGACK}$ signals are not affected by the SDMA; however, an indication of the SDMA internal bus request can be obtained from the $\overline{BCLRO}$ signal.

The SDMA will assert the $\overline{BCLRO}$ signal when it requests the bus if this capability is programmed in the SIM60 module configuration register and port E pin assignment register. $\overline{BCLRO}$ can be used to clear an external bus master from the external bus, if desired. For instance, $\overline{BCLRO}$ can be connected through logic to the external master's $\overline{HALT}$ signal, and then be negated externally when the external master's $\overline{AS}$ signal is negated. $\overline{BCLRO}$, as seen from the QUICC, is negated by the SDMA during its access to memory.

7.8.4.1 ONE MULTIPLEXED CHANNEL WITH STATIC FRAMES. With this configuration (see Figure 7-23), there are 64 entries in the SI RAM for transmit data and strobe routing and 64 entries for receive data and strobe routing. This configuration should be chosen when only one TDM is required and the routing on that TDM does not need to be changed dynamically.

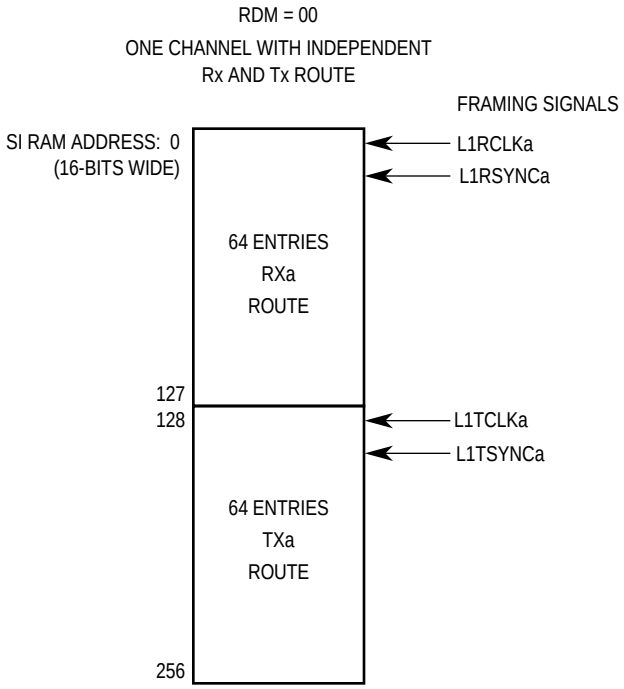


Figure 7-23. SI RAM: One TDM with Static Frames

7.8.4.2 ONE MULTIPLEXED CHANNEL WITH DYNAMIC FRAMES. With this configuration (see Figure 7-24), there is one multiplexed channel. The channel has 32 entries for transmit data and strobe routing and 32 entries for receive data and strobe routing. In each RAM, one of the partitions is the current-route RAM, and the other is a shadow RAM used to allow the user to change the serial routing. After programming the shadow RAM, the user sets the CSRx bit of the associated channel in the SI CR. When the next frame sync arrives, the SI will automatically exchange the current-route RAM for the shadow RAM. Refer to 7.8.4.7 SI RAM Dynamic Changes for more details on how to dynamically change the channel's route. This configuration should be chosen when only one TDM is required but the routing on that TDM may need to be changed dynamically.

5. PAPAR bits 6, 7, and 8 = 1. Configures L1TXDa, L1RXDa, and L1RCLKa.
6. PADIR bits 6 and 7 = 1. PADIR bit 8 = 0. Configures L1TXDa, L1RXDa, and L1RCLKa.
7. PCPAR bits 3, 10, and 11 = 1. Configures L1RQa, L1TSYNCa, and L1RSYNCa.
8. PCDIR bit 3 = 0. L1RQa is an input. L1TSYNCa will perform the L1GRa function and is therefore an output, but it does not need to be configured with a PCDIR bit. L1RSYNCa is an input, but it does not need to be configured with a PCDIR bit.
9. SIGMR = $04. Enable TDMa (one static TDM).
10. 1SICMR is not used.
11. 1SISTR and SIRP do not need to be read, but can be used for debugging information once the channels are enabled.
12. 1Enable the SCC1 for HDLC operation (to handle the LAPD protocol of the D channel), and set SCC2 and SCC4 as desired.

7.8.7 SI GCI Support

The normal mode of the GCI, also known as the ISDN-oriented modular rev 2.2 (IOM-2), and the SCIT are fully supported by the QUICC. The QUICC also supports the D channel access control in S/T interface terminals by using the command/indication (C/I) channel for that function.

The GCI bus consists of four lines: two data lines, a clock, and a frame synchronization line. Usually, an 8-kHz frame structure defines the various channels within the 256-kbps data rate. The QUICC can support two independent GCI buses and has independent receive and transmit sections for each one. The interface can also be used in a multiplexed frame structure on which up to eight physical layer devices multiplex their GCI channels. In this mode, the data rate would be 2048 kbps.

In the GCI bus, the clock rate is twice the data rate. The SI divides the input clock by two to produce the data clock.

The QUICC also has data strobe lines, and the 1× data rate clock L1CLKOx output pins. These signals are used for interfacing devices to GCI that do not support the GCI bus.

The GCI signals for each transmit and receive channel are as follows:

L1RSYNCx—Used as GCI sync signal; input to the QUICC. This signal indicates that the clock periods following the pulse designate the GCI frame.

L1RCLKx—Used as GCI clock; input to the QUICC. The L1RCLKx signal is twice the data clock.

L1RXDx—Used as GCI receive data; input to the QUICC.

L1TXDx—Used as GCI transmit data; open-drain output. Valid only for the bits that are supported by the IDL; three-stated otherwise.

L1CLKOx—Optional signal; output from QUICC. This 1× clock output can be used to clock devices that do not interface directly to GCI. If the double-speed clock

7. Receiving an address character when working in multidrop mode (The address character is written to the next buffer for software comparison.)

An example of the SCC UART Rx BD process is shown in Figure 7-48.

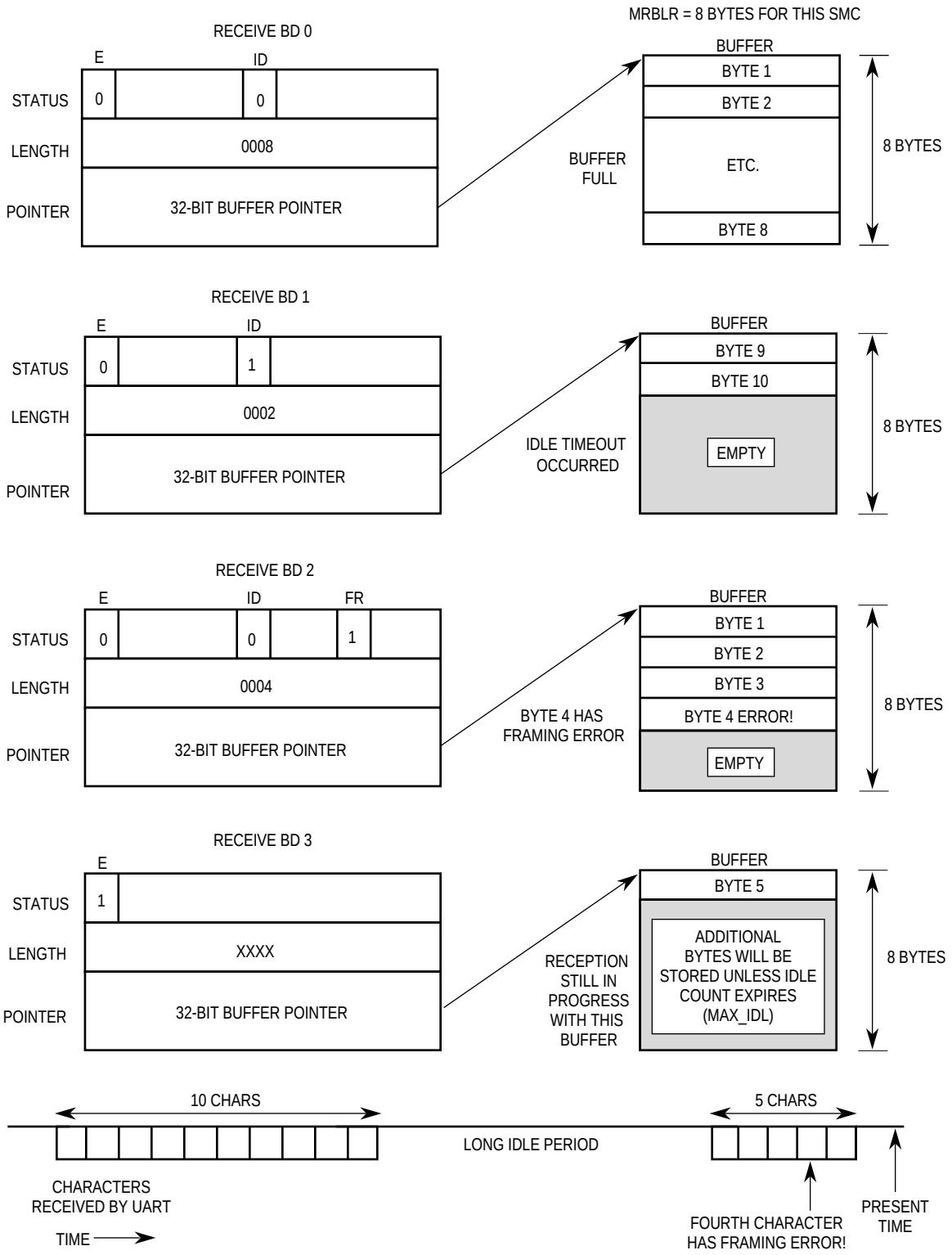


Figure 7-48. SCC UART Rx BD Example

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET + 0	E	—	W	I	L	F	CM	—	DE	—	LG	NO	AB	CR	OV	CD
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 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.

1 = The data buffer associated with this BD is empty, or reception is currently in progress. This Rx BD and its associated receive buffer are owned by the CP. Once the E-bit is set, the CPU32+ core should not write any fields of this Rx BD.

Bits 14, 8, 6—Reserved

W—Wrap (Final BD in Table)

0 = This is not the last buffer descriptor in the Rx BD table.

1 = This is the last buffer descriptor in the Rx BD table. After this buffer has been used, the CP will receive incoming data into the first BD in the table (the BD pointed to by RBASE). The number of Rx BD s in this table is programmable, and is determined only by the W-bit and the overall space constraints of the dual-port RAM.

I—Interrupt

0 = The RXB bit is not set after this buffer has been used, but RXF operation remains unaffected.

1 = The RXB or RXF bit in the HDLC event register will be set when this buffer has been used by the HDLC controller. These two bits may cause interrupts (if enabled).

L—Last in Frame

This bit is set by the HDLC controller when this buffer is the last in a frame. This implies the reception of a closing flag or reception of an error, in which case one or more of the CD, OV, AB, and LG bits are set. The HDLC controller will write the number of frame octets to the data length field.

0 = This buffer is not the last in a frame.

1 = This buffer is the last in a frame.

F—First in Frame

This bit is set by the HDLC controller when this buffer is the first in a frame.

0 = The buffer is not the first in a frame.

1 = The buffer is the first in a frame.

However, the R-bit will be cleared if an error occurs during transmission, regardless of the CM bit.

BR—BCS Reset

0 = The transmitter BCS accumulation is not reset.
1 = The transmitter BCS accumulation is reset (used for STX or SOH) before sending the data buffer.

TD—Transmit DLE

0 = No automatic DLE transmission is to occur before the data buffer.
1 = The transmitter will transmit a DLE character before sending the data buffer, which saves writing the first DLE to a separate data buffer when working in transparent mode. See the TR bit for information on control characters.

TR—Transparent Mode

0 = The transmitter will enter (remain in) the normal mode after sending the data buffer. In this mode, the transmitter will automatically insert SYNCs in an underrun condition.
1 = The transmitter enters or remains in transparent mode after sending the data buffer. In this mode, the transmitter automatically inserts DLE-SYNC pairs in the underrun condition. Underrun occurs when the BISYNC controller finishes a buffer with the L-bit set to zero and the next BD is not available. The transmitter also checks all characters before sending them; if a DLE is detected, another DLE is automatically sent. The user must insert a DLE or program the BISYNC controller to insert it (using TD) before each control character required. The transmitter will calculate the CRC16 BCS even if the BCS bit in the BISYNC mode register is programmed to LRC. The PTCRC should be initialized to CRC16 before setting this bit.

B—BCS Enable

0 = Buffer consists of characters to be excluded from the BCS accumulation.
1 = Buffer consists of characters to be included in the BCS accumulation.

The following status bits are written by the CP after it has finished transmitting the associated data buffer.

UN—Underrun

The BISYNC controller encountered a transmitter underrun condition while transmitting the associated data buffer.

CT—CTS Lost

CTS was lost during message transmission.

Data Length

The data length is the number of octets that the CP should transmit from this BD's data buffer. It is never modified by the CP. The data length should be greater than zero.

W—Wrap (Final BD in Table)

0 = This is not the last buffer descriptor in the Tx BD Table.
1 = This is the last buffer descriptor in the Tx BD Table. After this buffer has been used, the CP will receive incoming data into the first BD in the table (the BD pointed to by TBASE). The number of Tx BDs in this table is programmable, and is determined only by the wrap bit and the overall space constraints of the dual-port RAM.

I—Interrupt

0 = No interrupt is generated after this buffer has been serviced.
1 = The TX bit in the PIP event register will be set when this buffer has been serviced by the CP, which can cause an interrupt.

L—Last

0 = This buffer is not the last buffer of the frame.
1 = This buffer is the last buffer of the frame.

CM—Continuous Mode

0 = Normal Operation.
1 = The R-bit is not cleared by the CP after this buffer is closed, allowing the associated data buffer to be retransmitted automatically when the CP next accesses this BD. However, the R bit will be cleared if an error occurs during transmission

F—Fault

0 = The Fault status remained negated during transmission
1 = The Fault status was asserted during transmission

PE—Printer Error

0 = The PError status remained negated during transmission
1 = The PError status was asserted during transmission

S—Select Error

0 = The Select status remained asserted during transmission
1 = The Select status was negated during transmission

7.13.8.11 CENTRONICS TRANSMITTER EVENT REGISTER (PIPE) . When the Centronics Transmitter protocol is selected, the SMC2 event register is called the Centronics Transmitter event register. It is an 8-bit register which is used to report events recognized by the Centronics channel and generate interrupts. On recognition of an event, the Centronics controller will set its corresponding bit in the Centronics event register.

The Centronics event register is a memory-mapped register that may be read at any time. A bit is cleared by writing a one (writing a zero does not affect a bit's value). More than one bit may be cleared at a time. All unmasked bits must be cleared before the CP will clear the internal interrupt request. This register is cleared at reset.

rupt configuration register (CICR). The five LSBs are fixed in the CPIC, and are unique for each CPIC interrupt source.

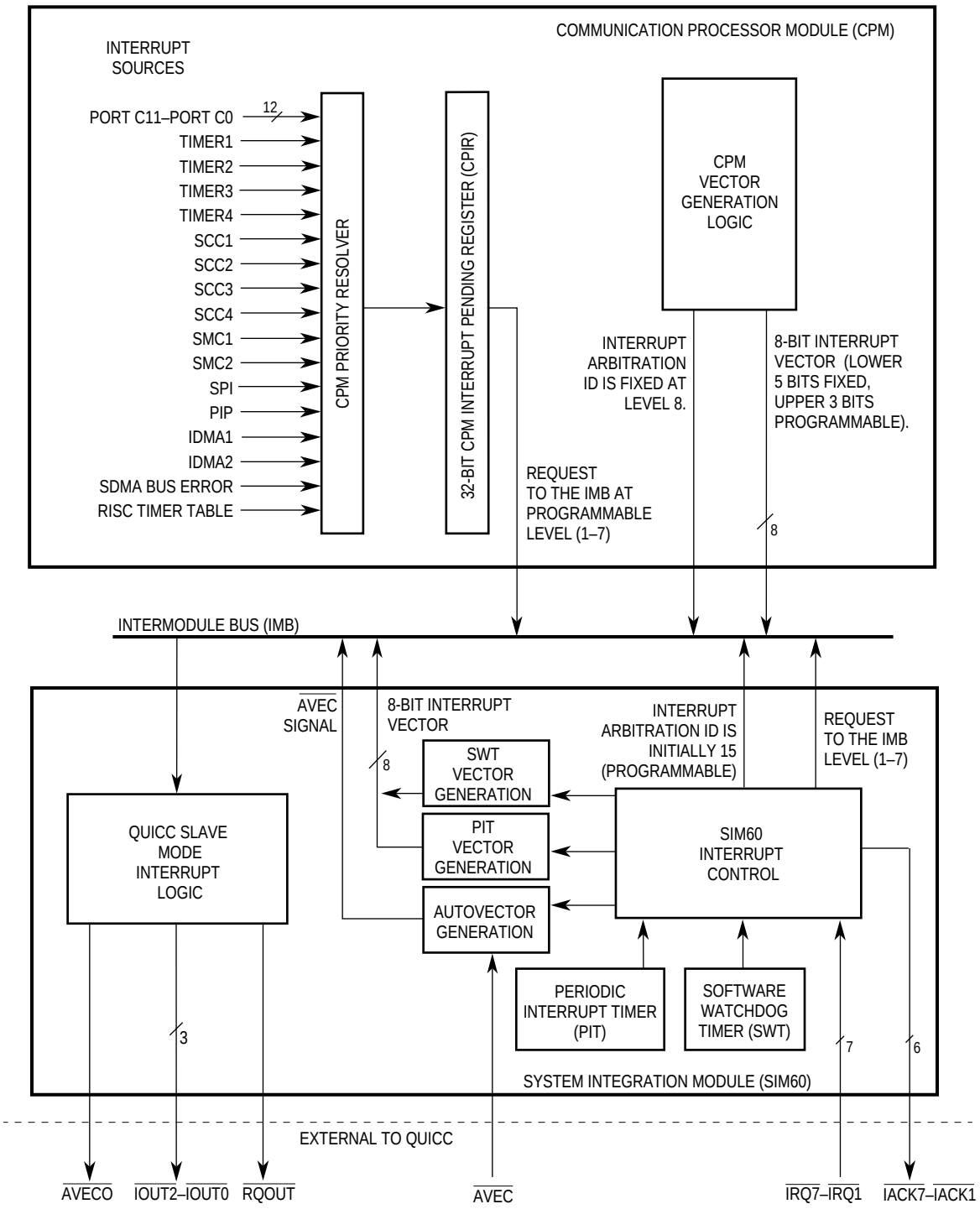


Figure 7-99. QUICC Interrupt Structure

The RBASE and TBASE values are new to the QUICC. They point to the start of the buffer descriptor tables, must be long-word aligned, and must point to the dual-port RAM area.

The RFCR and TFCR registers have one additional purpose and a different bit placement on the QUICC. If RFCR and TFCR are cleared on the MC68302 or not used with the MC68302, then the equivalent function on the QUICC can be implemented by writing $18 to the QUICC RFCR and TFCR. Note that on the QUICC there is an additional function code pin to signify DMA operation, and the suggestion of $18 uses this capability.

The MRBLR value has the same function on both devices. Additionally, if the receive FIFO is set to 32-bit-wide mode, the MRBLR value must be aligned to a long word.

The internal state parameter has the same function on both devices.

The MC68302 RBD# and TBD# have the same concept on the QUICC, except the parameters are called RBPTR and TBPTR and are now 16-bit values. They point to the current buffer descriptor; however, they are now offsets from the beginning location of the QUICC dual-port RAM. For example, the current transmit buffer descriptor location may be found by MBAR + TBPTR.

The internal data pointer parameter has the same function on both devices.

The internal byte count parameter has the same function on both devices.

9.3.4.2.3 Protocol-Dependent Parameter RAM Values. These values are very similar between devices, although new functions are often added on the QUICC. Where possible, any parameter for a given protocol that is the same for the MC68302 and the QUICC carries the same name on both devices.

The following points note changes between the MC68302 and the QUICC protocol-dependent parameter RAM for a given protocol:

In the UART mode, if the MAX_IDL entry is programmed to $0000 on the QUICC, the function will be disabled.

In the UART mode, the ability to send special control characters, such as XOFF, no longer requires the CHARACTER8 entry. Rather, a new entry, called the TOSEQ entry, has been created. The programming of the TOSEQ entry, however, is the same as the old MC68302 CHARACTER8 entry.

In UART mode, note the new parameters for a receive character mask (RCCM) and a function that times the length of a receive break (BRKLN).

The HDLC parameter RAM is the same, except for two new entries: RFTHR and RFCNT. They allow the user to reduce the number of received frame interrupts generated, and should be used in higher data rate applications.

The BISYNC parameter RAM is unchanged.

DDCMP is a microcode RAM product on the QUICC. The port of DDCMP from the MC68302 is not discussed.

V.110 is not supported on the QUICC.

if DRAM is used elsewhere in the system. The QUICC does not support bursting by the MC68EC030.

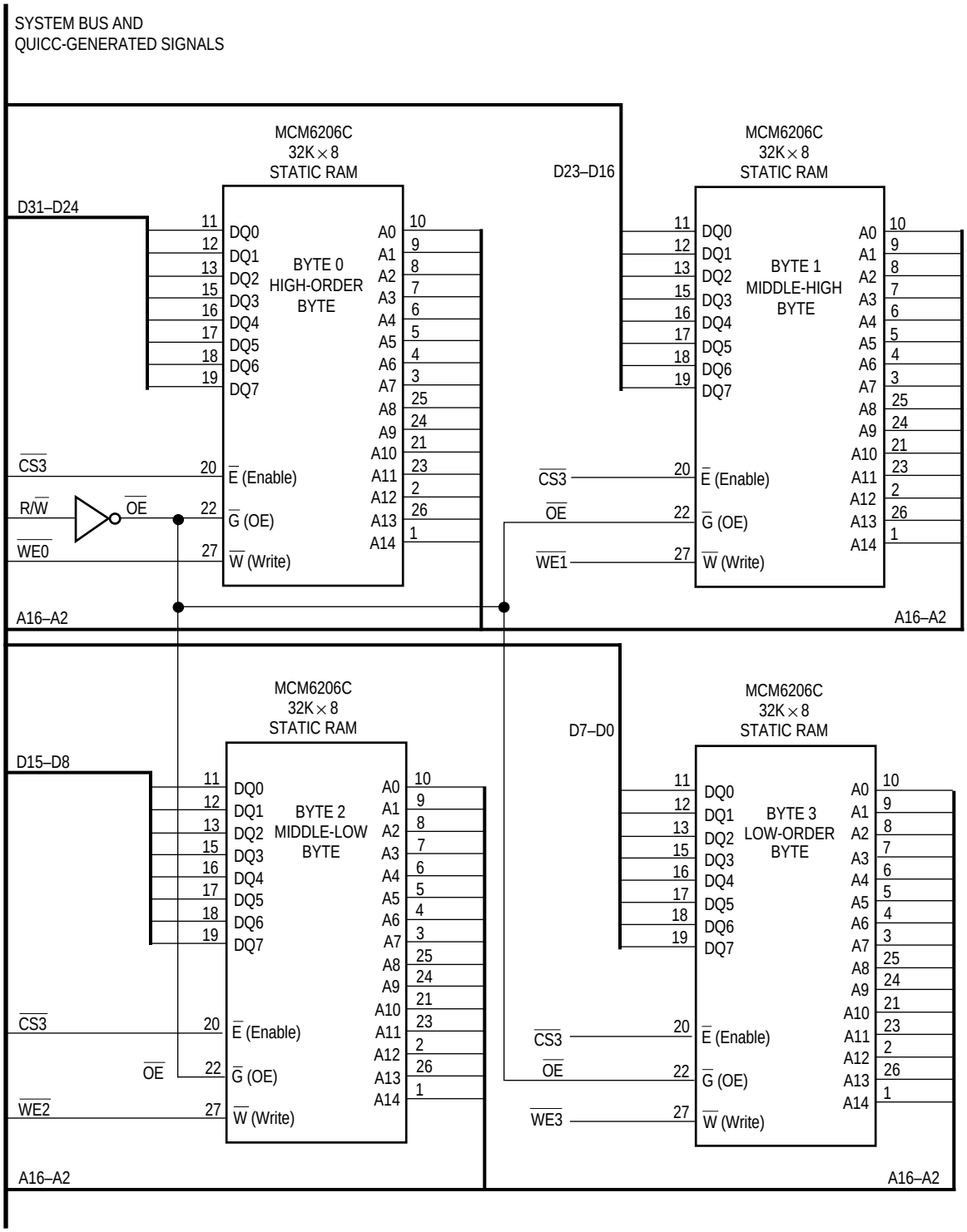


Figure 9-30. 128-Kbyte Static RAM Bank—32 Bits Wide

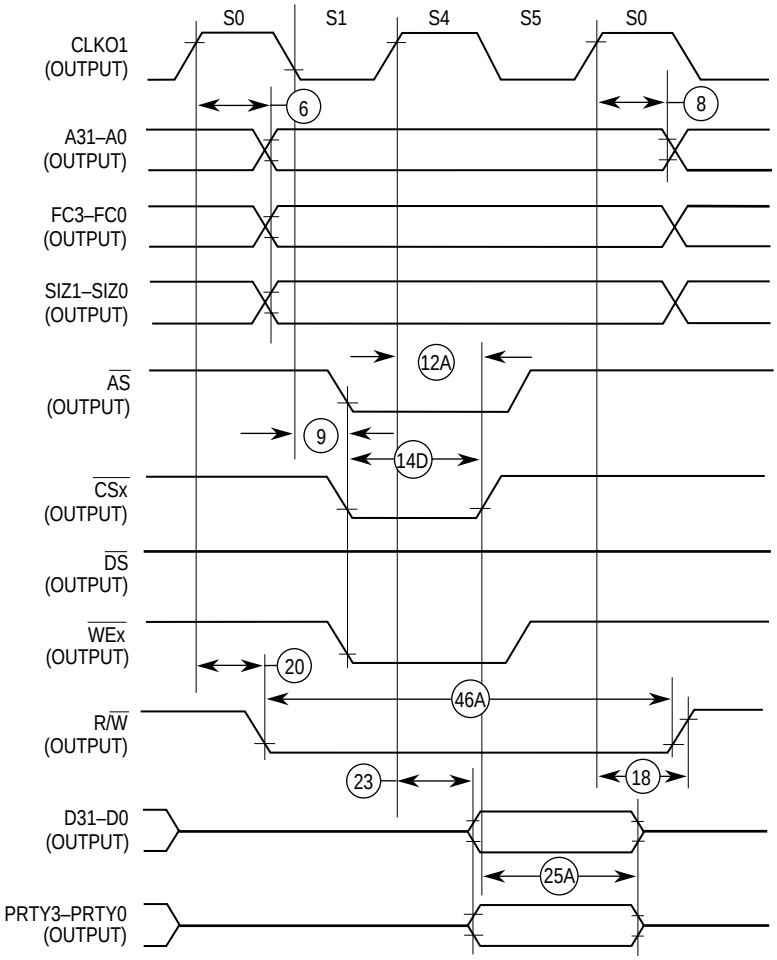
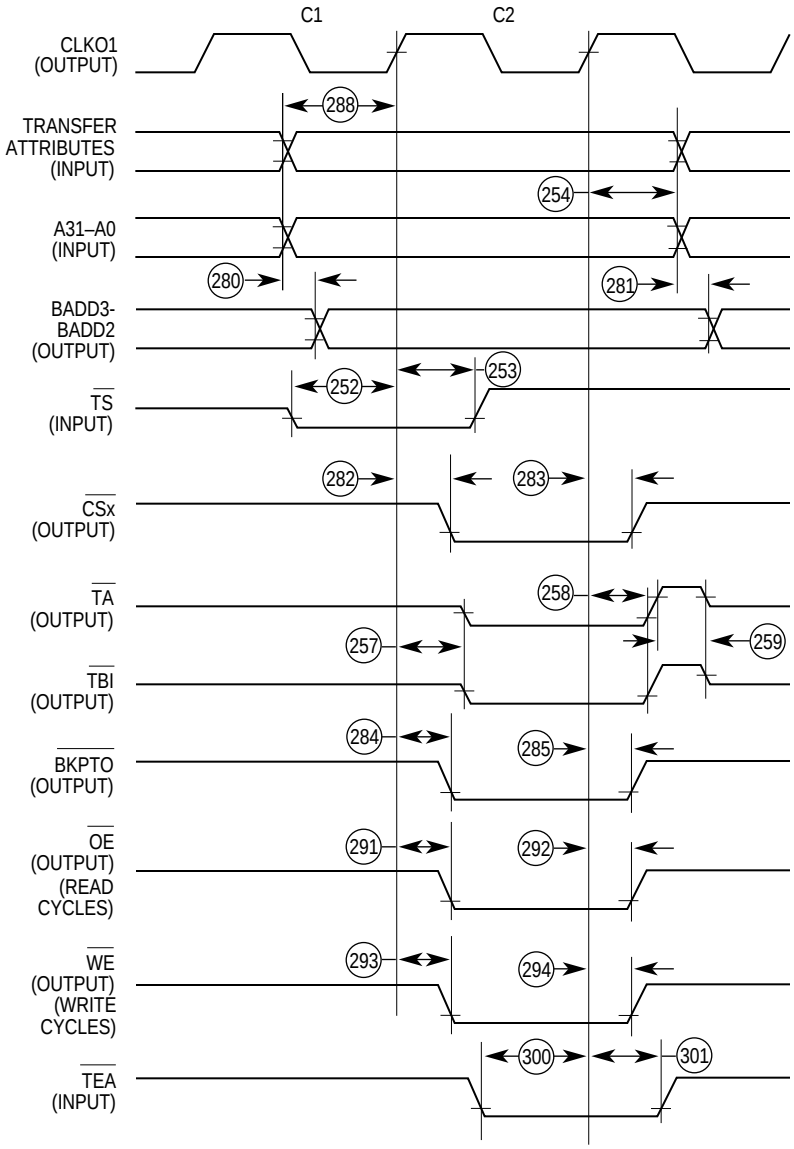


Figure 10-10. SRAM: Fast Termination Write Cycle (CSNTQ = 1)



NOTE: MC68040 Transfer Attribute Signals = SIZx, TTx, TMx, R/W, LOCK

Figure 10-44. MC68040 SRAM Read/Write Cycles (TSS40 =0, CSNT40 = 0)

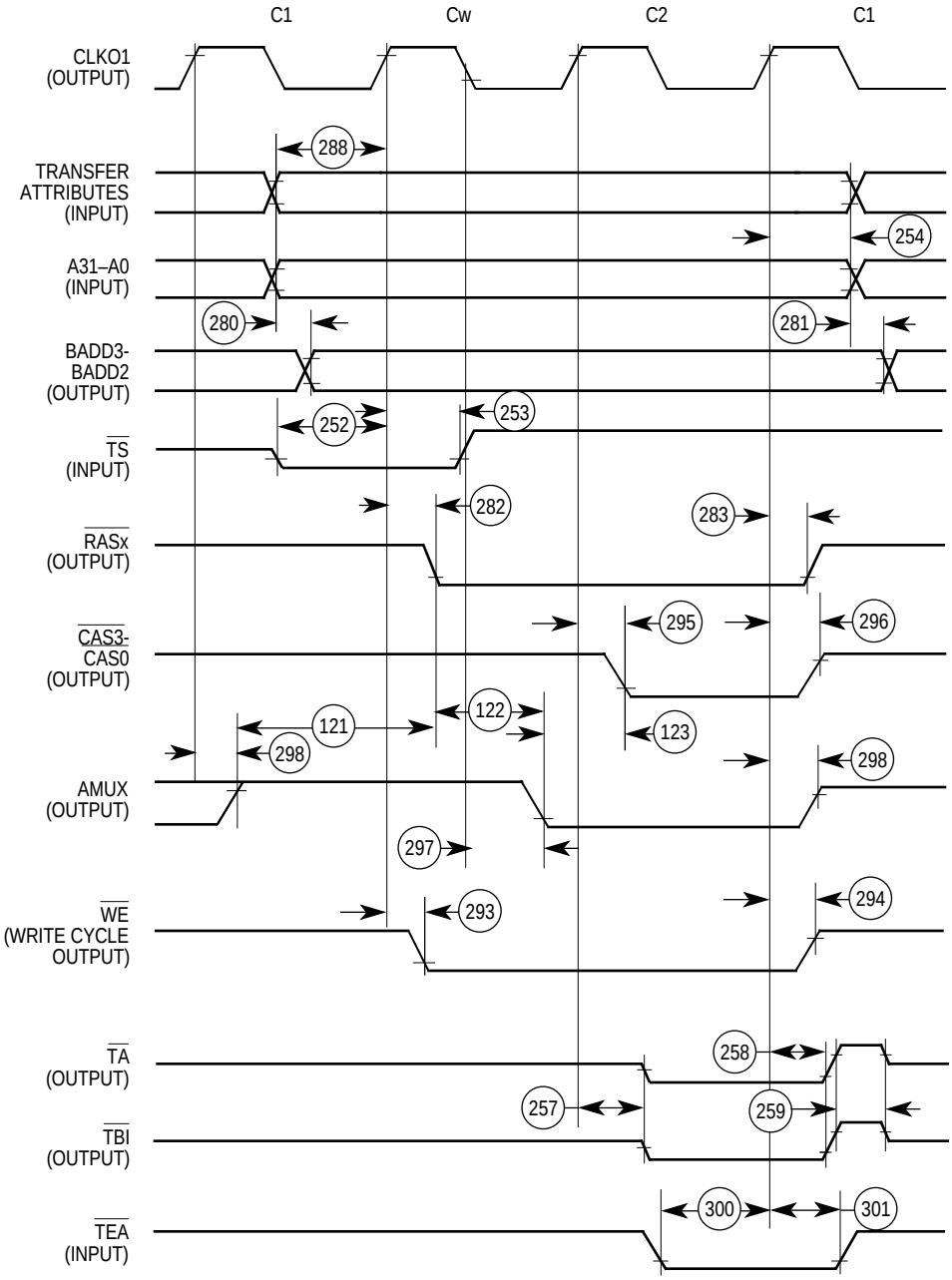


Figure 10-46. External MC68040 DRAM Cycles Timing Diagram

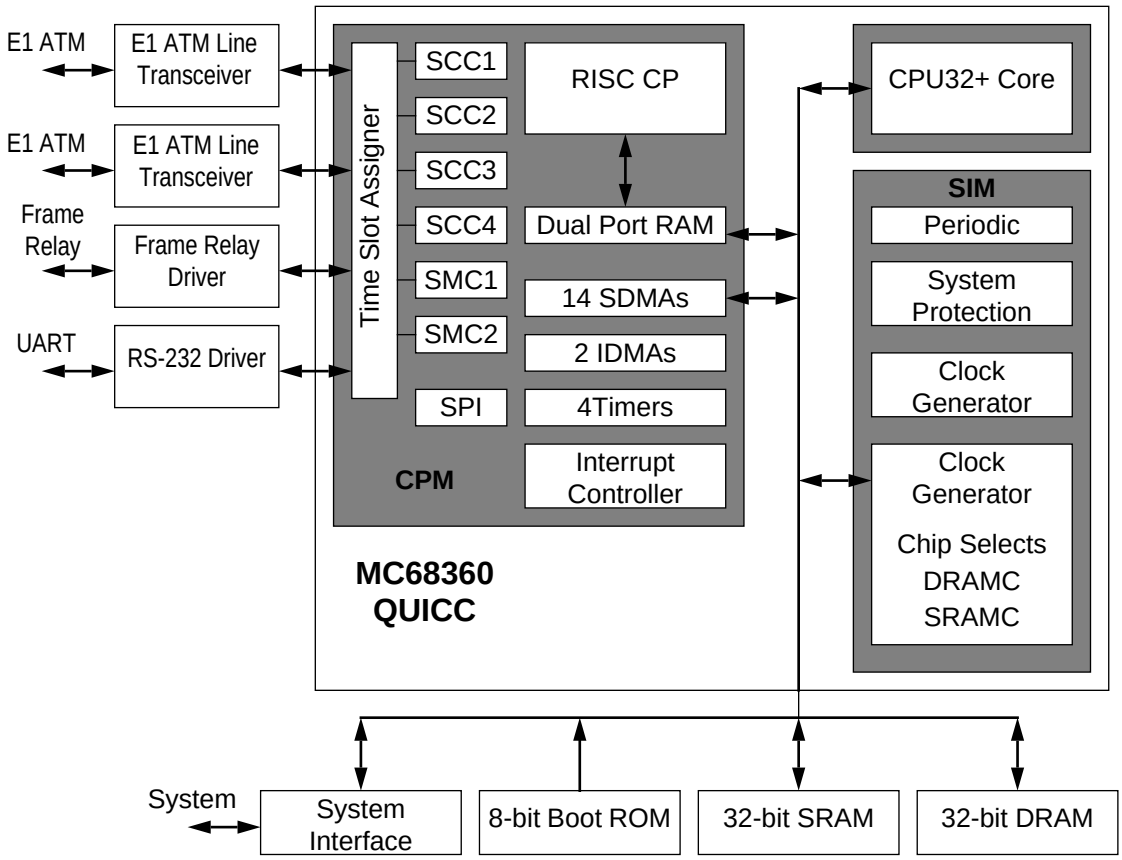


Figure C-2. ATM/Frame Relay Interwork System

- Empty cells transmitted when there are no pending data transfers.
- Empty cells and cells with non-matching headers are automatically discarded on receive.
- Scrambling option is provided utilizing the self-synchronizing $X^{43} + 1$ scrambling polynomial.
- Incoming cells with incorrect HECs are received and marked.
- Bandwidth reservation mechanism in the transmitter to allow mixing of data and isochronous services.
- CAM support on reception for handling many connections.

  —Consumes 1280 bytes of the QUICC's internal memory.

C.3.2 Performance

At 25 Mhz, an aggregate ATOM1 bandwidth of 10 Mbps divided among the 4 SCCs consumes 100% of the processing power of the RISC communications engine. If only a percentage of the total available ATOM1 bandwidth is used, the remaining RISC processing power can be used to run other protocols on other channels. Table C-3 shows the possible QUICC configuration.

INDEX

T