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	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	-40°C ~ 85°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=mc68360crc25l

SECTION 4
BUS OPERATION

This section provides a functional description of the system bus, the signals that control it, and the bus cycles provided for data transfer operations. It also describes the error and halt conditions, bus arbitration, and reset operation. Operation of the external bus is the same whether the QUICC or an external device is the bus master; the names and descriptions of bus cycles are from the viewpoint of the bus master. For exact timing specifications, refer to Section 10 Electrical Characteristics.

NOTE

The bus operation of the QUICC is very similar to the bus operation of the MC68030 and the MC68340. Much of the text and figures of the bus operation of those devices is common to this section.

The QUICC also supports the MC68EC040 (or other M68040 family members) as an external bus master. The MC68EC040 can access QUICC registers and use QUICC peripherals. The QUICC has a glueless MC68EC040 interface and special logic for acting as the MC68EC040 memory controller, interrupt controller, and the provider of system protection logic. The MC68EC040 bus operation is described in the *M68040 User Manual*. When the QUICC is the bus master of an M68040 system, its bus operation remains the same when it is the only bus master in the system. See 4.6.7 Internal Accesses for a description and timing diagram of the MC68EC040 internal read/write cycles (i.e., MC68EC040 reading/writing the QUICC) and interrupt acknowledge cycles. See 6.11 General-Purpose Chip-Select Overview (SRAM Banks) and 6.12 DRAM Controller Overview (DRAM Banks) for more information on the timing diagrams of MC68EC040 DRAM and SRAM accesses.

The QUICC architecture supports byte, word, and long-word operands allowing access to 8-, 16-, and 32-bit data ports through the use of asynchronous cycles controlled by the size outputs (SIZ1, SIZ0) and data size acknowledge inputs ($\overline{\text{DSACK1}}$, $\overline{\text{DSACK0}}$).

The QUICC allows byte, word, and long-word operands to be located in memory on any byte boundary. For a misaligned transfer, more than one bus cycle may be required to complete the transfer, regardless of port size. For a port less than 32 bits wide, multiple bus cycles may be required for an operand transfer due to either misalignment or a port width smaller than the operand size. Instruction words and their associated extension words must be aligned on word boundaries. The user should be aware that misalignment of word or long-word operands can cause the CPU32+ to perform multiple bus cycles for operand transfers; therefore, processor performance is optimized if word and long-word memory operands are

Xn	Index register
[An]	Address extension
cc	Condition code
d#	Displacement
	Example: d_{16} is a 16-bit displacement
⟨ea⟩	Effective address
#⟨data⟩	Immediate data; a literal integer
label	Assembly program label
list	List of registers
	Example: D3–D0
[...]	Bits of an operand
	Examples: [7] is bit 7; [31:24] are bits 31–24
(...)	Contents of a referenced location
	Example: (Rn) refers to the contents of Rn
CCR	Condition code register (lower byte of SR)
	X—extend bit
	N—negative bit
	Z—zero bit
	V—overflow bit
	C—carry bit
PC	Program counter
SP	Active stack pointer
SR	Status register
SSP	Supervisor stack pointer
USP	User stack pointer
FC	Function code
DFC	Destination function code register
SFC	Source function code register
+	Arithmetic addition or postincrement
–	Arithmetic subtraction or predecrement
/	Arithmetic division or conjunction symbol
×	Arithmetic multiplication
=	Equal to

$$\text{PIT period} = \frac{\text{PITR count value}}{16}$$

This gives a range from 62.5 ms, with a PITR value of $01, to 15.94 s, with a PITR value of $FF.

For a fast calculation of PIT period using a 32.768-kHz crystal, the following equations can be used:

With prescaler disabled:

PIT period = PITR (122 μs)

With prescaler enabled:

PIT period = PITR (62.5 ms)

6.3.2.2 USING THE PIT AS A REAL-TIME CLOCK. The PIT can be used as a real-time clock interrupt by setting it up to generate an interrupt with a 1-second period. When using a 32.768-kHz (or 4.192-MHz) crystal, the PITR should be loaded with a value of 16 decimal ($10) with the prescaler enabled to generate interrupts at a 1-sec rate.

6.3.3 Freeze Support

FREEZE is asserted by the CPU32+ if a breakpoint is encountered with background mode enabled. Refer to Section 5 CPU32+ for more information on the background mode. When FREEZE is asserted, the double bus fault monitor and spurious interrupt monitor continue to operate normally. However, the SWT, the bus monitor, and the PIT may be affected. Setting the FRZ1 bit in the MCR disables the SWT and the PIT when FREEZE is asserted. Setting the FRZ0 bit in the MCR disables the bus monitor when FREEZE is asserted.

If the CONFIG pins are configured with the CPU32+ core enabled, then one clock after reset is complete, the CONFIG2 pin will become the FREEZE output. Thus, the pin will start driving low one clock after reset. It will then be asserted (high) if the freeze condition occurs. If the CONFIG pins configure the QUICC to slave mode, then the FREEZE output is not available.

6.3.4 Low-Power Stop Support

Executing the LPSTOP instruction provides reduced power consumption when the QUICC is idle, with only the SIM remaining active. Operation of the SIM60 is controlled by the PLLCR. LPSTOP disables the clock to the SWT in the low state. The SWT, which remains stopped until the LPSTOP mode is ended, begins to run again on the next rising clock edge.

NOTE

When the CPU32+ executes the STOP instruction (as opposed to LPSTOP), the SWT continues to run. If the SWT is enabled, it issues a reset or interrupt when its timeout occurs.

(CLKO1 and CLKO2). The PLL synchronizes these clock signals to each other. These clock signals are discussed in the following paragraphs.

6.5.5.1 SPCLK. SPCLK is supplied to the PIT and SWT sub-modules in the SIM60. SPCLK is always the EXTAL frequency or EXTAL/128, depending on the configuration of the divide-by-128 prescaler. When EXTAL frequency > 10 MHz is selected by configuration pins (MODCK1-MODCK0 = 01), then the PLL is clocked with the EXTAL frequency and SPCLK is EXTAL/128. (i1616.e. When MODCK is 11 --> SPCLK is equal to EXTAL).

6.5.5.2 GENERAL SYSTEM CLOCK. This basic clock is supplied to all other modules and sub-modules on the QUICC, including the CPU32+, the RISC controller, and most other features in the communication processor module (CPM). The general system clock also supplies the SIMCLK to the SIM60 in normal device operation. The general system clock is the same as the CLKO1 frequency, and the CLKO2 signal is 2× the general system clock in normal device operation. The general system clock defaults to VCO/2 = 25 MHz (assuming a 25-MHz system frequency).

The frequency of the general system clock can be changed dynamically with the CDVCR, as shown in Figure 6-7. This configuration is called slow-go mode.

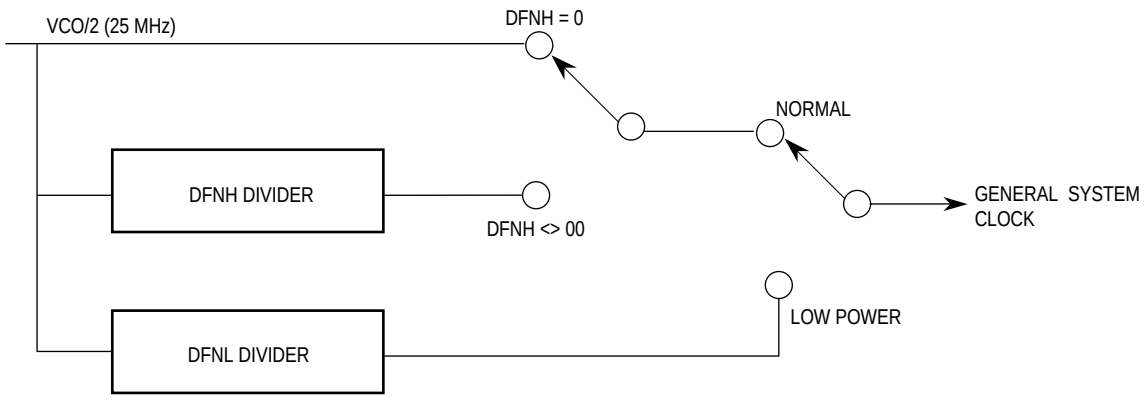


Figure 6-7. General System Clock Select

The general system clock can be operated at three frequencies. Normal operation is the highest frequency (25 MHz in a 25-MHz system). The general system clock can also be operated at a low frequency and a high frequency. The definition of low is made in the DFNL value in CDVCR; the definition of high is made in the DFNH value in CDVCR.

The frequency of the general system clock can be changed dynamically by software. The user may simply cause the general system clock to switch to its low frequency. However, in some applications, there is a need for high frequency during certain periods. An example is in interrupt routines, etc., that need more performance than the low frequency operation, but must consume less power than in normal operation. The SIM60 allows a method to automatically switch between low and high frequency operation.

6.11 GENERAL-PURPOSE CHIP-SELECT OVERVIEW (SRAM BANKS)

Any memory bank that is not used to control DRAM may be used as a general-purpose chip select, including pins $\overline{CS0}$–$\overline{CS7}$. This bank is called an SRAM bank. These pins may be used to support external memory such as SRAM, EPROM, flash EPROM, EEPROM, and peripherals.

The SRAM banks also have some unique features not available in the DRAM banks. First, upon system reset, a global (boot) chip select is available. This provides a boot ROM chip select before the system is fully configured. Second, the SRAM banks offer two-clock accesses to external SRAM. Finally, each SRAM bank supports a choice of the port size of its memory or peripheral to be 8, 16, or 32 bits with proper $\overline{DSACK}$ generation for those port sizes. Thus, an 8-bit EPROM may be used with a 32-bit SRAM, etc.

6.11.1 Associated Registers

The general-purpose chip selects are controlled by the global memory register (GMR) and the memory controller status register (MSTAT). There is one GMR and MSTAT in the memory controller. Additionally, each SRAM bank has a base register (BR) and an option register (OR).

The GMR is used to control global parameters for both SRAM and DRAM banks.

The MSTAT reports write protect violations and parity errors for both SRAM and DRAM banks.

The BR and the OR for each of the general-purpose chip selects program most of the features. The BR contains a valid (V) bit to indicate that the register information for that chip select is valid.

6.11.2 8-, 16-, and 32-Bit Port Size Configuration

The general-purpose chip selects support dynamic bus sizing. Defined 8-bit ports are accessible on both odd and even addresses when connected to data bus bits 31–24; defined 16-bit ports can be accessed as odd bytes, even bytes, or even words when connected to data bus bits 31–16; and defined 32-bit ports can be accessed as odd bytes, even bytes, odd words, and even words or long words on long-word boundaries. The port size is specified by the SPS bits in the OR.

6.11.3 Write Protect Configuration

The WP bit in each BR can restrict write access to its range of addresses. Any attempt to write this area will result in the WPER bit being set in the MSTAT.

6.11.4 Programmable Wait State Configuration

The general-purpose chip selects support internal DSACKx generation. They allow fast two-clock accesses to external memory by an internal bus master; from zero-wait-state accesses (3 clocks) up to 14-wait-state accesses (17 clocks) are allowed for internal bus masters. For external bus masters, two-clock accesses are not allowed, but 14 wait states may be programmed. Additionally, if the EMWS bit is set in the GMR, the chip selects can

NOTE

An access to a region that has no V-bit set may cause a bus monitor timeout.

0 = This DRAM/SRAM bank is invalid.
1 = This DRAM/SRAM bank is valid.

NOTE

Following a system reset, the V-bit is set in BR0 if the global chip select is enabled. See the CONFIG pins for more details.

WP—Write Protection

This bit can restrict write accesses within the address range of a BR. An attempt to write to the range of addresses specified in a BR that has this bit set can cause the $\overline{\text{BERR}}$ signal to be asserted by the bus monitor logic (if enabled), causing termination of this cycle.

0 = Both read and write accesses are allowed.
1 = Only read accesses are allowed. The $\overline{\text{RAS}}$/$\overline{\text{CS}}$ signal, $\overline{\text{TA}}$, and $\overline{\text{DSACK}}$ will not be asserted by the QUICC on write cycles to this memory bank. WPER will be set in the MSTAT register if a write to this memory bank is attempted.

PAREN—Parity Checking Enable

This bit is used to enable checking of parity on either an SRAM or DRAM bank.

0 = Parity checking is disabled.
1 = Parity checking is enabled.

NOTE

Parity checking is not possible for asynchronous external masters.

CSNTQ—$\overline{\text{CS}}$ Negate Timing QUICC (SRAM Bank Only)

This bit is used to determine when $\overline{\text{CS}}$ is negated during an internal QUICC or external QUICC/MC68030-type bus master write cycle. This is helpful to meet address/data hold time requirements for slow memories and peripherals (see Figure 6-13 and Figure 6-14).

0 = $\overline{\text{CS}}$ is negated normally (as late as possible).
1 = $\overline{\text{CS}}$ is negated one phase earlier, but the cycle length is not affected.

NOTE

CSNTQ is ignored for an SRAM cycle by an external master if the SYNC bit is cleared. CSNTQ = 1 is not valid for external DSACK assertion

CSNT40—$\overline{\text{CS}}$ Negate Timing MC68EC040 (SRAM Bank Only)

This bit is used to determine when $\overline{\text{CS}}$ is negated during an MC68EC040 write cycle. This is helpful to meet address/data hold time requirements (see Figure 6-15).

0 = $\overline{\text{CS}}$ is negated normally (as late as possible).
1 = $\overline{\text{CS}}$ is negated one phase earlier, but the cycle length is not affected.

ue of 0 stored in these bits gives a timer tick of 1 × (1024) = 1024 general system clocks. A value of 63 (decimal) stored in these bits gives a timer tick of 64 × (1024) = 65536 general system clocks.

Bits 7-0—Reserved - set to zero.

7.1.2 RISC Microcode Revision Number

The RISC controller writes a revision number stored in its ROM to a dual-port RAM location called REV_num. REV_num is located in the miscellaneous parameter RAM. The other locations are reserved for future use. The microcode rivision number only reflect the revision of the micro code. It dose not always refrect the MASK number.

Address	Name	Width	Description
Misc Base + 00	REV_num	Word	Microcode Revision Number
Misc Base + 02	RES	Word	Reserved
Misc Base + 04	RES	Long	Reserved
Misc Base + 08	RES	Long	Reserved

7.2 COMMAND SET

The host processor (CPU32+ or other external processor) issues commands to the RISC by writing to the command register (CR). The CR only needs to be accessed on rare occasions. For instance, to terminate the transmission of a frame by an SCC without waiting until the end of the frame, a STOP TX command can be issued to an SCC through the command register. The commands are described in general terms in the following paragraphs; they are described in specific terms when the protocol or feature is described in detail.

The host should set the FLG bit in the CR when it issues commands. The CP clears FLG after completing the command to indicate to the host that it is ready for the next command. Subsequent commands to the CR may be given only after FLG is cleared. The software reset command (issued by setting the RST bit) may be given regardless of the state of FLG, but the host should still set FLG when setting RST.

The CR, a 16-bit, memory-mapped, read-write register, is cleared by reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RST	—			OPCODE				CH NUM				—			FLG

RST—Software Reset Command

This bit is set by the host and cleared by the CP. On execution of this command, the RST bit and the FLG bit are cleared within two general system clocks. The RISC reset routine is approximately 60 clocks long, but the user can begin initialization of the CP immediately after this command is given. This command is useful when the host wants to reset the registers and parameters for all the channels (SCCs, SMCs, SPI, and PIP) as well as the RISC processor and RISC timer tables. This command does not affect the serial interface (SI) or the parallel I/O registers.

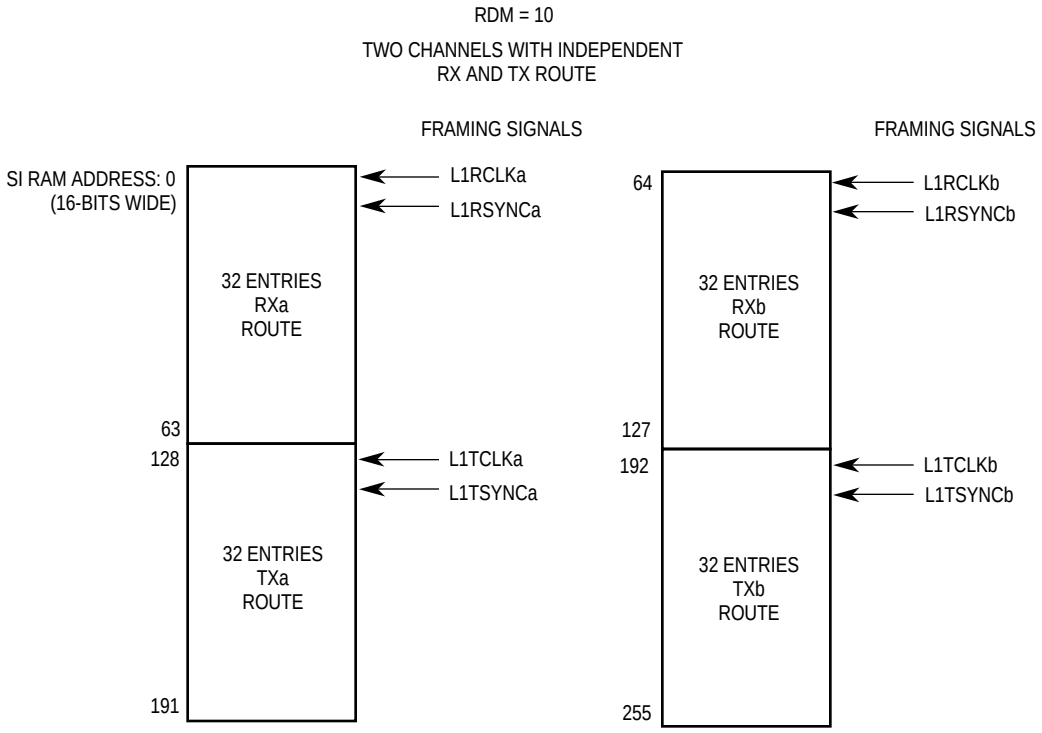


Figure 7-25. SI RAM: Two TDMs with Static Frames

7.8.4.4 TWO MULTIPLEXED CHANNELS WITH DYNAMIC FRAMES. With this configuration (see Figure 7-26), there are two multiplexed channels. Each channel has 16 entries for transmit data and strobe routing and 16 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 two TDMs are required and the routing on each TDM may need to be changed dynamically.

receiver begins receiving data. This behavior is similar to the MC68302 totally transparent mode behavior when the EXSYN bit in its SCC mode register is set.

SYNL—Sync Length (BISYNC and Transparent Mode Only)

These bits determine the operation of an SCC receiver that is configured for BISYNC or totally transparent operation only. See the data synchronization register definition in the BISYNC and totally transparent descriptions for more information.

00 = The sync pattern in the DSR is not used. An external sync signal is used instead ($\overline{CD}$ pin asserted).
01 = 4-bit sync. The receiver will synchronize on a 4-bit sync pattern stored in DSR. This character and additional syncs can be programmed to be stripped using the SYNC character in the parameter RAM. The transmitter will transmit the entire contents of the DSR prior to each frame.
10 = 8-bit sync. This option should be chosen along with the BISYNC protocol to implement mono-sync. The receiver will synchronize on an 8-bit sync pattern stored in DSR. The transmitter will transmit the entire contents of the DSR prior to each frame.
11 = 16-bit sync. Also called BISYNC. The receiver will synchronize on a 16-bit sync pattern stored in DSR. The transmitter will transmit the DSR prior to each frame.

RTSM—$\overline{RTS}$ Mode

This bit may be changed on the fly.

0 = Send idles between frames as defined by the protocol and the Tend bit. $\overline{RTS}$ is negated between frames (default).
1 = Send flags/syncs between frames according to the protocol. $\overline{RTS}$ is always asserted whenever the SCC is enabled.

RSYN—Receive Synchronization Timing (Valid for a Totally Transparent Channel Only)

0 = Normal operation.
1 = If CDS = 1, then the $\overline{CD}$ pin should be asserted on the second bit of the receive frame, rather than the first. This configuration matches the behavior of the MC68302 totally transparent receiver when its EXSYN bit is set; it is included on the QUICC for compatibility.

EDGE—Clock Edge

The EDGE bits determine the clock edge used by the DPLL for adjusting the receive sample point due to jitter in the received signal. The selection of the EDGE bits is ignored in the UART protocol or the x1 mode of the RDCR bits.

00 = Both the positive and negative edges are use for changing the sample point (default).
01 = Positive edge. Only the positive edge of the received signal is used for changing the sample point.
10 = Negative edge. Only the negative edge of the received signal is used for changing the sample point.
11 = No adjustment is made to the sample points.

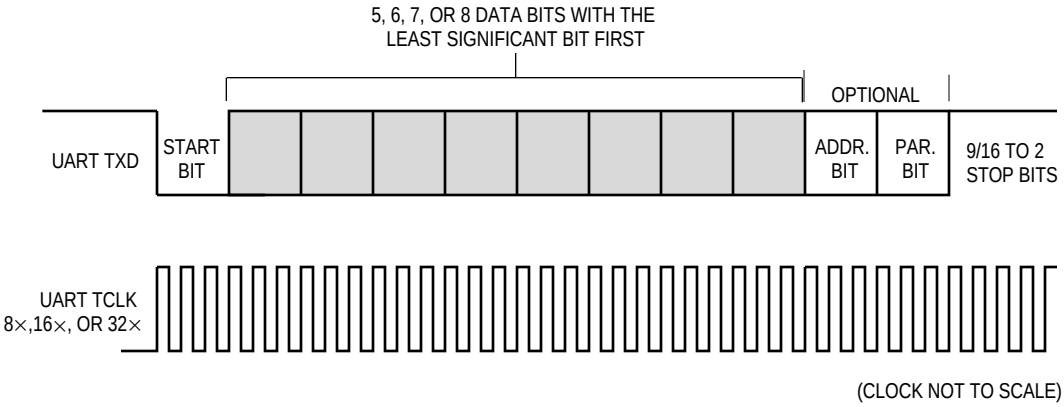


Figure 7-46. UART Character Format

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.

miscuous mode, the user can use the Miss bit to quickly determine whether the frame was destined to this station. This bit is valid only if the L bit is set.

0 = The frame was received because of an address recognition hit.
1 = The frame was received because of promiscuous mode.

LG—Rx Frame Length Violation

A frame length greater than the maximum defined for this channel was recognized (only the maximum-allowed number of bytes is written to the data buffer).

NO—Rx Nonoctet Aligned Frame

A frame that contained a number of bits not divisible by 8 was received, and the CRC check that occurred at the preceding byte boundary generated an error.

SH—Short Frame

A frame length that was less than the minimum defined for this channel was recognized.

This indication is possible only if the RSH bit is set in the PSMR.

CR—Rx CRC Error

This frame contains a CRC error.

OV—Overrun

A receiver overrun occurred during frame reception.

CL—Collision

This frame was closed because a collision occurred during frame reception. This bit will be set only if a late collision occurred or if the RSH bit is enabled in the PSMR. The late collision definition is determined by the LCW bit in the PSMR.

Data Length

The data length is the number of octets written by the CP into this BD's data buffer. It is written by the CP once as the buffer is closed.

When this BD is the last BD in the frame (L = 1), the data length contains the total number of frame octets (including four bytes for CRC).

NOTE

The actual amount of memory allocated for this buffer should be greater than or equal to the contents of the MRBLR.

Rx Data Buffer Pointer

The receive buffer pointer, which always points to the first location of the associated data buffer, may reside in either internal or external memory. This pointer must be divisible by 4.

7.10.23.19 ETHERNET TRANSMIT BUFFER DESCRIPTOR (TX BD). Data is presented to the Ethernet controller for transmission on an SCC channel by arranging it in buffers ref-

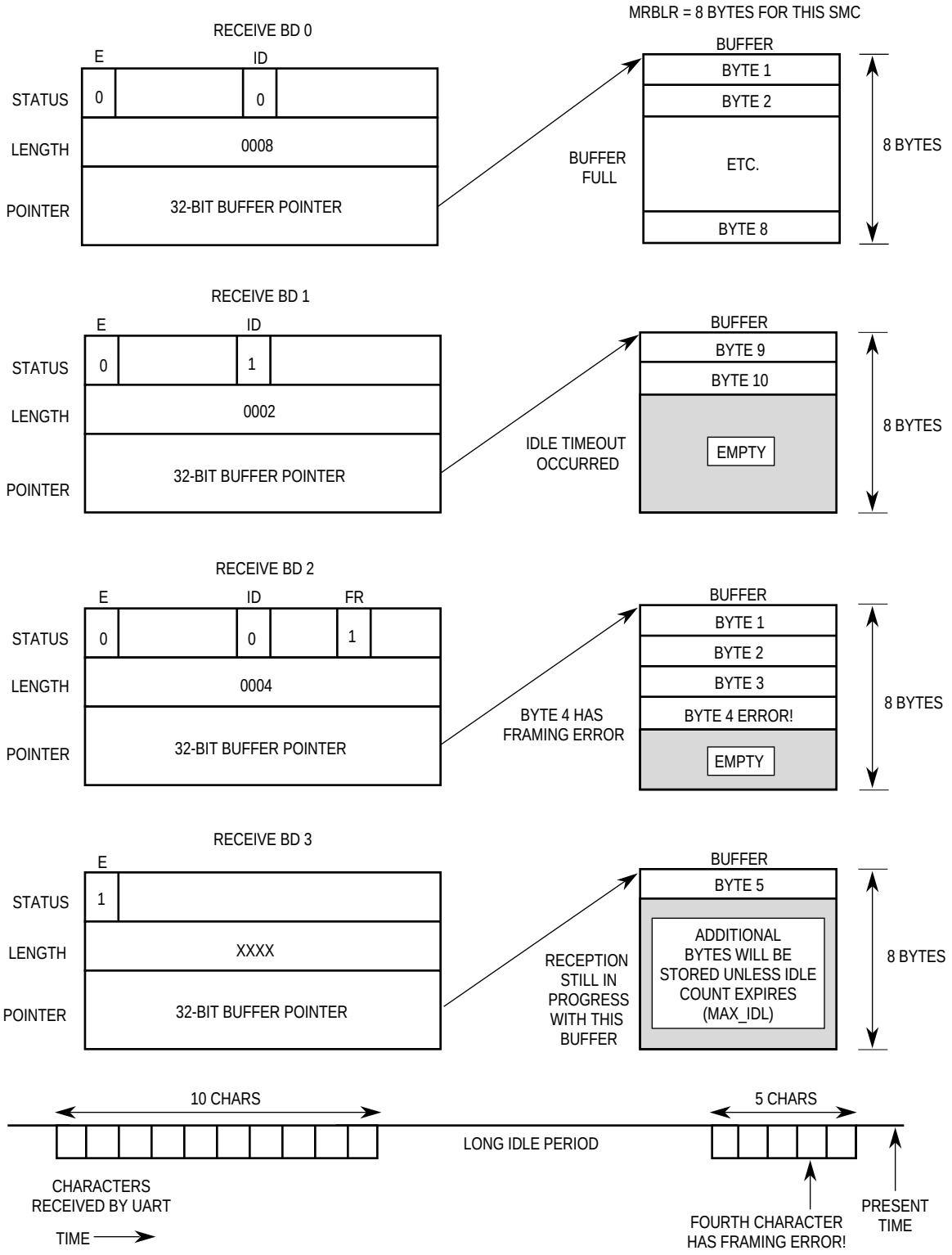


Figure 7-76. SMC UART Rx BD Example

7.11.7.13 SMC UART TRANSMIT BUFFER DESCRIPTOR (TX BD). Data is presented to the CP for transmission on an SMC channel by arranging it in buffers referenced by the

Bits 7, 5, 3—Reserved.

BRKe—Break End

The end of break sequence was detected. This indication will be no sonner than after one idle bit is received following a break sequence.

BRK—Break Character Received

A break character was received. If a very long break sequence occurs, this interrupt will occur only once after the first all-zeros character is received.

BSY—Busy Condition

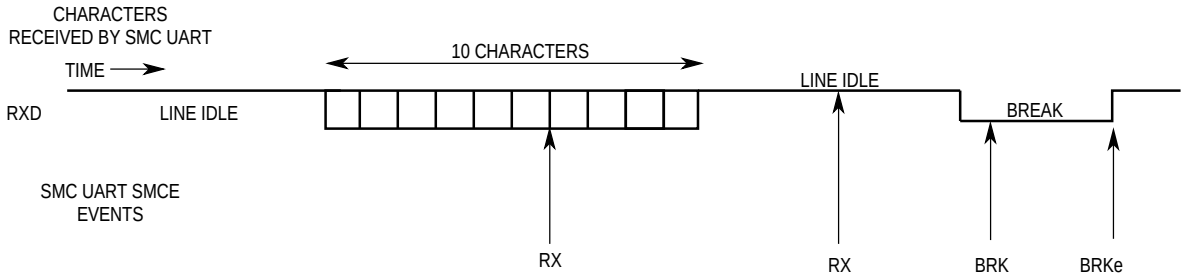
A character was received and discarded due to lack of buffers. This bit is be set no sooner than the middle of the last stop bit of the first receive character for which there is no available buffer. Reception continues when an empty buffer is provided.

TX—Tx Buffer

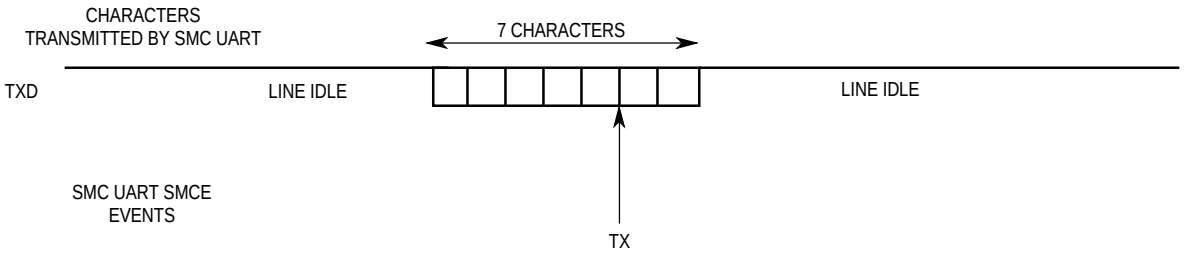
A buffer has been transmitted over the UART channel. This bit is set once the transmit data of the last character in the buffer was written to the transmit FIFO. The user must wait two character times to be sure that the data was completely sent over the transmit pin.

RX—Rx Buffer

A buffer has been received and its associated Rx BD is now closed. This bit is set no sooner than the middle of the last stop bit of the last character that was written to the receive buffer.



NOTES:
1. The first RX event assumes receive buffers are six bytes each.
2. The second RX event position is programmable based on the max_IDL value.
3. The BRK event occurs after the first break character is received.



NOTE: The TX event assumes all seven characters were put into a single buffer, and the TX event occurred when the seventh character was written to the SMC transmit FIFO.

Figure 7-77. SMC UART Interrupts Example

For each DDx bit, the definition is as follows:

0 = General-purpose I/O. The peripheral functions of the pin are not used.
1 = Dedicated peripheral function. The pin is used by the internal module. The on-chip peripheral function to which it is dedicated may be determined by other bits such as these in the PBDIR.

7.14.8 Port B Example

PB0 can be configured as a general-purpose I/O pin or as an open-drain pin. It may also be the receiver reject pin for the SCC1 Ethernet CAM interface ($\overline{\text{RRJCT1}}$) or the SPI select input ($\overline{\text{SPISEL}}$). If the PB0 pin is not configured to connect to the $\overline{\text{RRJCT}}$ signal or the $\overline{\text{SPISEL}}$ signal, then the SCC and/or SPI receives V_{DD} on that signal.

NOTE

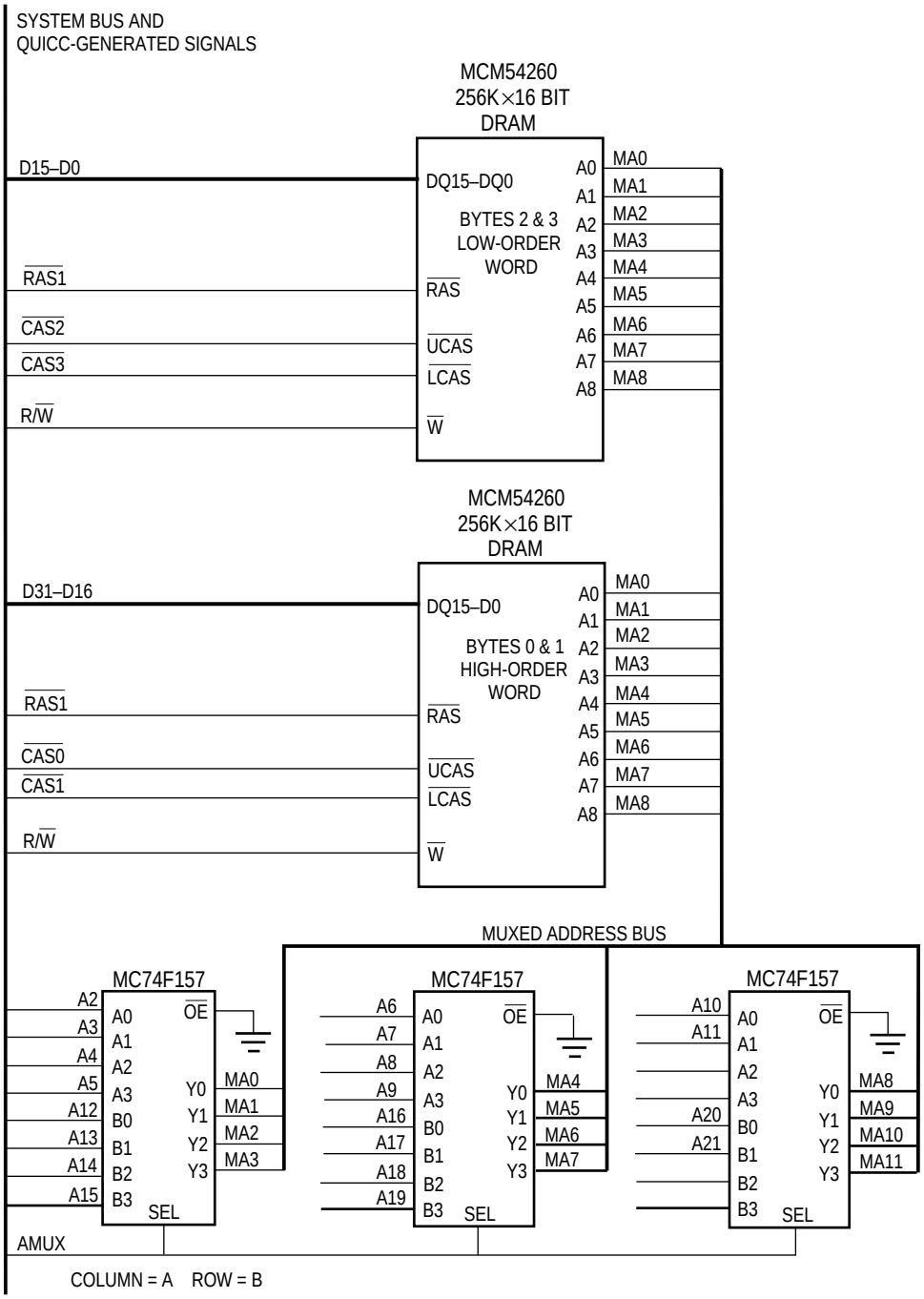
In the description of the PIP, the PB0 pin, as well as other port B pins, can also be used as PIP functions. However, the PIP does not affect the operation of port B unless it is enabled. Therefore, the PIP description does not need to be studied by users of port B unless the PIP will be used in the application.

7.14.9 Port C Pin Functions

Port C consists of 12 general-purpose I/O pins with interrupt capability on each pin. Refer to Table 7-21 for the description of all port C pin options.

Table 7-21. Port CPin Assignment

Signal	PCPAR = 0		PCPAR = 1		Input to On-Chip Peripherals
	PCDIR = 1 or PCSO = 0	PCDIR = 0 and PCSO = 1	PCDIR = 0	PCDIR = 1	
PC0	Port C0	—	RTS1	L1ST1	—
PC1	Port C1	—	RTS2	L1ST2	—
PC2	Port C2	—	$\overline{\text{RTS3}}$/L1RQB	L1ST3	—
PC3	Port C3	—	$\overline{\text{RTS4}}$/L1RQA	L1ST4	—
PC4	Port C4	CTS1	—		GND
PC5	Port C5	CD1	TGATE1		GND
PC6	Port C6	CTS2	—		GND
PC7	Port C7	CD2	TGATE2		GND
PC8	Port C8	CTS3	L1TSYNCB	SDACK2	$\overline{\text{CTS3}}$ and/or L1TSYNCB = GND
PC9	Port C9	CD3	L1RSYNCB		GND
PC10	Port C10	CTS4	L1TSYNCA	SDACK1	$\overline{\text{CTS4}}$ and/or L1TSYNCA = GND
PC11	Port C11	CD4	L1RSYNCA		GND



NOTE: MA11–MA9 not used.

Figure 9-33. 1-Mbyte DRAM Bank—32 Bits Wide

9.8.3.2 CONFIGURING THE MEMORY CONTROLLER. The following register initializations are for the memory controller.

The global memory register (GMR) should be configured as follows:

The RFCNT bits may be set as desired. At 25 MHz, an RFCNT value of 24 (decimal) gives

one refresh every 15.6 µs.

RFEN should be set.

RCYC depends on the DRAM speed. At 25 MHz (an 80-ns DRAM SIMM), RCYC should be 00.

PGS2–PGS0 should be set to 011 for the 1M × 32 DRAM SIMM.

DPS should be set to 00 (32-bit DRAM port size).

WBT40 does not apply to this application.

WBTQ depends on timing; it should be set for 80-ns MCM32100 SIMMs.

DWQ should be set if page mode enabled (PGME = 1).

DW40 does not apply to this application.

EMWS is not used in synchronous mode. (SYNC = 1)

SYNC should be set for a synchronous operation of the memory controller.

OPAR does not apply to this application.

PBEE does not apply to this application.

TSS40 does not apply to this application.

NCS should normally be cleared.

GAMX should be cleared for an external master system.

The memory controller status register (MSTAT) is used for reporting write protect and parity errors and does not require initialization.

The eight base registers (BRs), one for each memory bank, should be configured as follows:

The BA27–BA11 bits may be set as desired. Different memory arrays should not overlap. BA31–BA28 should be cleared since the byte write lines are used with an external master in the system.

For simplicity, FC3–FC0 can be cleared.

TRLXQ depends on timing of memory/peripheral.

BACK40 does not apply to this application.

CSNT40 does not apply to this application.

CSNTQ should normally be cleared.

PAREN should be cleared since parity is not used in this application.

WP should be set for EPROM and flash EPROM; otherwise, it should be cleared.

V should be set if the memory bank is used.

The eight option registers (ORs), one for each memory bank, should be configured as follows:

The TCYC bits should be set to determine the number of wait states required.

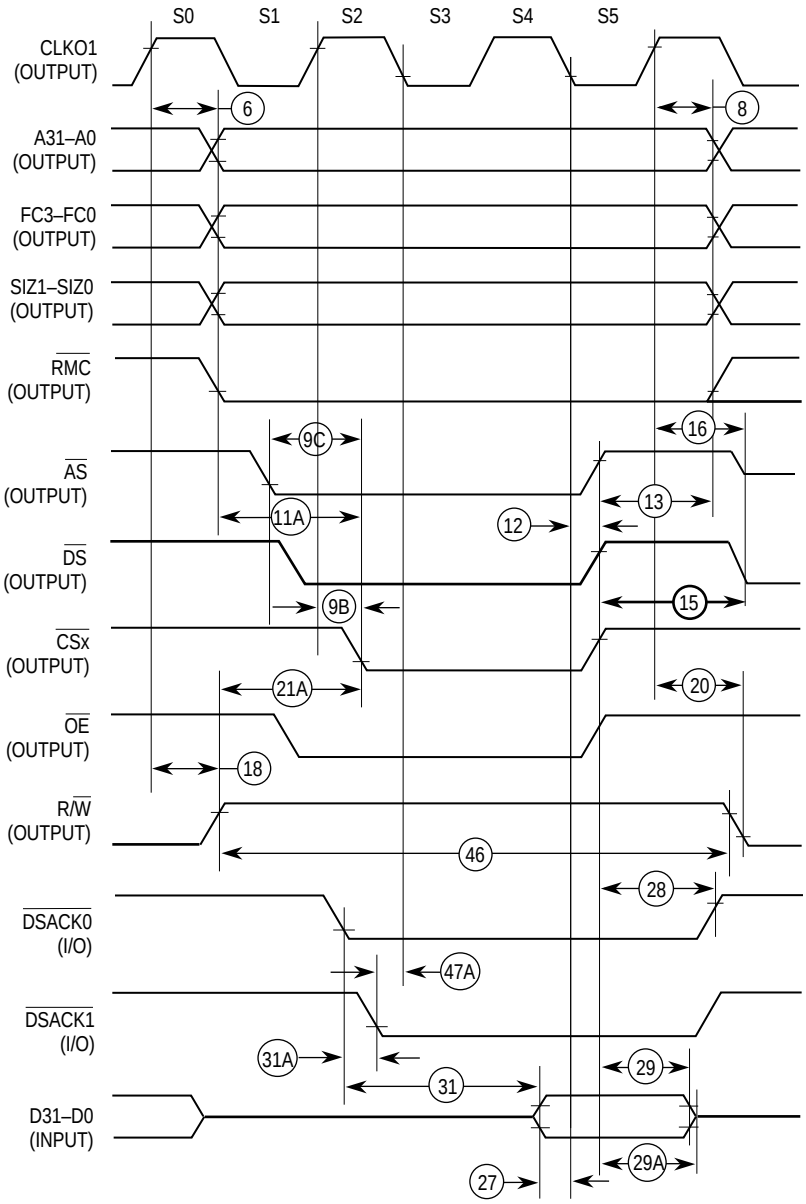


Figure 10-6. SRAM: Read Cycle (TRLX = 1)

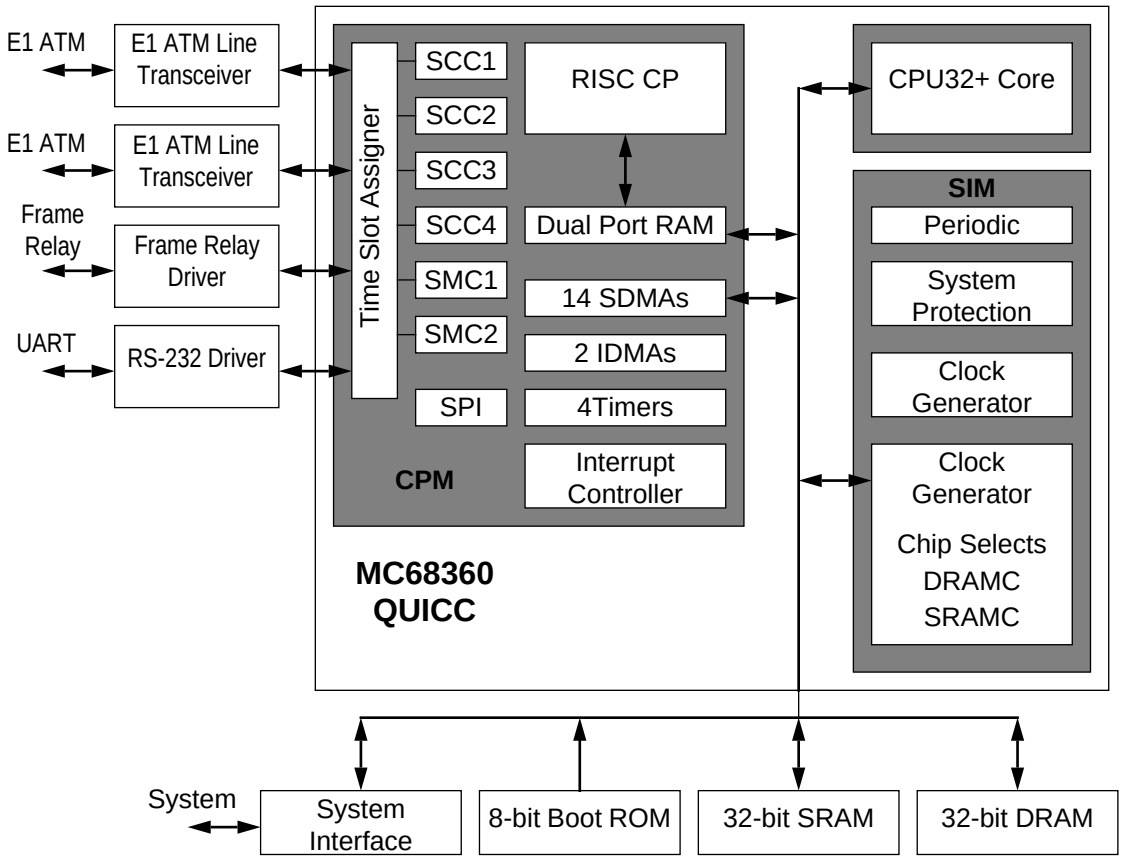


Figure C-2. ATM/Frame Relay Interwork System

- 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.