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	M68000
Number of Cores/Bus Width	1 Core, 8/16-Bit
Speed	16MHz
Co-Processors/DSP	Communications; RISC CPM
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	132-BPGA Exposed Pad
Supplier Device Package	132-PGA (34.5x34.5)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc68302rc16c

UNITED STATES

ALABAMA, Huntsville	(205) 464-6800
ARIZONA, Tempe	(602) 897-5056
CALIFORNIA, Agoura Hills	(818) 706-1929
CALIFORNIA, Los Angeles	(310) 417-8848
CALIFORNIA, Irvine	(714) 753-7360
CALIFORNIA, Rosevllle	(916) 922-7152
CALIFORNIA, San Diego	(619) 541-2163
CALIFORNIA, Sunnyvale	(408) 749-0510
COLORADO, Colorado Springs	(719) 599-7497
COLORADO, Denver	(303) 337-3434
CONNECTICUT, Wallingford	(203) 949-4100
FLORIDA, Maitland	(407) 628-2636
FLORIDA, Pompano Beach/ Fort Lauderdal	(305) 486-9776
FLORIDA, Clearwater	(813) 538-7750
GEORGIA, Atlanta	(404) 729-7100
IDAHO, Boise	(208) 323-9413
ILLINOIS, Chicago/Hoffman Estates	(708) 490-9500
INDIANA, Fort Wayne	(219) 436-5818
INDIANA, Indianapolis	(317) 571-0400
INDIANA, Kokomo	(317) 457-6634
IOWA, Cedar Rapids	(319) 373-1328
KANSAS, Kansas City/Mission	(913) 451-8555
MARYLAND, Columbia	(410) 381-1570
MASSACHUSETTS, Marborough	(508) 481-8100
MASSACHUSETTS, Woburn	(617) 932-9700
MICHIGAN, Detroit	(313) 347-6800
MINNESOTA, Minnetonka	(612) 932-1500
MISSOURI, St. Louis	(314) 275-7380
NEW JERSEY, Fairfield	(201) 808-2400
NEW YORK, Fairport	(716) 425-4000
NEW YORK, Hauppauge	(516) 361-7000
NEW YORK, Poughkeepsie/Fishkill	(914) 473-8102
NORTH CAROLINA, Raleigh	(919) 870-4355
OHIO, Cleveland	(216) 349-3100
OHIO, Columbus Worthington	(614) 431-8492
OHIO, Dayton	(513) 495-6800
OKLAHOMA, Tulsa	(800) 544-9496
OREGON, Portland	(503) 641-3681
PENNSYLVANIA, Colmar	(215) 997-1020
Philadelphia/Horsham	(215) 957-4100
TENNESSEE, Knoxville	(615) 690-5593
TEXAS, Austin	(512) 873-2000
TEXAS, Houston	(800) 343-2692
TEXAS, Plano	(214) 516-5100
VIRGINIA, Richmond	(804) 285-2100
WASHINGTON, Bellevue	(206) 454-4160
Seattle Access	(206) 622-9960
WISCONSIN, Milwaukee/Brookfield	(414) 792-0122

CANADA

BRITISH COLUMBIA, Vancouver	(604) 293-7605
ONTARIO, Toronto	(416) 497-8181
ONTARIO, Ottawa	(613) 226-3491
QUEBEC, Montreal	(514) 731-6881

INTERNATIONAL

AUSTRALIA, Melbourne	(61-3)887-0711
AUSTRALIA, Sydney	(61(2)906-3855
BRAZIL, Sao Paulo	55(11)815-4200
CHINA, Beijing	86 505-2180
FINLAND, Helsinki	358-0-35161191
Car Phone	358(49)211501
FRANCE, Paris/Vanves	33(1)40 955 900
GERMANY, Langenhagen/ Hanover	49(511)789911
GERMANY, Munich	49 89 92103-0
GERMANY, Nuremberg	49 911 64-3044
GERMANY, Sindelfingen	49 7031 69 910
GERMANY,Wiesbaden	49 611 761921
HONG KONG, Kwai Fong	852-4808333
Tai Po	852-6668333
INDIA, Bangalore	(91-812)627094
ISRAEL, Tel Aviv	972(3)753-8222
ITALY, Milan	39(2)82201
JAPAN, Aizu	81(241)272231
JAPAN, Atsugi	81(0462)23-0761
JAPAN, Kumagaya	81(0485)26-2600
JAPAN, Kyushu	81(092)771-4212
JAPAN, Mito	81(0292)26-2340
JAPAN, Nagoya	81(052)232-1621
JAPAN, Osaka	81(06)305-1801
JAPAN, Sendai	81(22)268-4333
JAPAN, Tachikawa	81(0425)23-6700
JAPAN, Tokyo	81(03)3440-3311
JAPAN, Yokohama	81(045)472-2751
KOREA, Pusan	82(51)4635-035
KOREA, Seoul	82(2)554-5188
MALAYSIA, Penang	60(4)374514
MEXICO, Mexico City	52(5)282-2864
MEXICO, Guadalajara	52(36)21-8977
Marketing	52(36)21-9023
Customer Service	52(36)669-9160
NETHERLANDS, Best	(31)49988 612 11
PUERTO RICO, San Juan	(809)793-2170
SINGAPORE	(65)2945438
SPAIN, Madrid	34(1)457-8204
or	34(1)457-8254
SWEDEN, Solna	46(8)734-8800
SWITZERLAND, Geneva	41(22)7991111
SWITZERLAND, Zurich	41(1)730 4074
TAIWAN, Taipei	886(2)717-7089
THAILAND, Bangkok	(66-2)254-4910
UNITED KINGDOM, Aylesbury	44(296)395-252

FULL LINE REPRESENTATIVES

COLORADO, Grand Junction	
Cheryl Lee Whltely	(303) 243-9658
KANSAS, Wichita	
Melinda Shores/Kelly Greiving	(316) 838 0190
NEVADA, Reno	
Galena Technology Group	(702) 746 0642
NEW MEXICO, Albuquerque	
S&S Technologies, Inc.	(505) 298-7177
UTAH, Salt Lake City	
Utah Component Sales, Inc.	(801) 561-5099
WASHINGTON, Spokane	
Doug Kenley	(509) 924-2322
ARGENTINA, Buenos Aires	
Argonics, S.A.	(541) 343-1787

HYBRID COMPONENTS RESELLERS

Elmo Semiconductor	(818) 768-7400
Minco Technology Labs Inc.	(512) 834-2022
Semi Dice Inc.	(310) 594-4631

NOTE

An interrupt will only be generated if the IDMA bit is set in the IMR.

REQG—Request Generation

The following decode shows the definitions for the REQG bits:

00 = Internal request at limited rate (limited burst bandwidth) set by burst transfer (BT) bits
01 = Internal request at maximum rate (one burst)
10 = External request burst transfer mode ($\overline{\text{DREQ}}$ level sensitive)
11 = External request cycle steal ($\overline{\text{DREQ}}$ edge sensitive)

SAPI—Source Address Pointer (SAP) Increment

0 = SAP is not incremented after each transfer.
1 = SAP is incremented by one or two after each transfer, according to the source size (SSIZE) bits and the starting address.

DAPI—Destination Address Pointer (DAP) Increment

0 = DAP is not incremented after each transfer.
1 = DAP is incremented by one or two after each transfer, according to the destination size (DSIZE) bits and the starting address.

SSIZE—Source Size

The following decode shows the definitions for the SSIZE bits.

00 = Reserved
01 = Byte
10 = Word
11 = Reserved

DSIZE—Destination Size

The following decode shows the definitions for the DSIZE bits.

00 = Reserved
01 = Byte
10 = Word
11 = Reserved

BT—Burst Transfer

The BT bits control the maximum percentage of the M68000 bus that the IDMA can use during each 1024 clock cycle period following the enabling of the IDMA. The IDMA runs for a consecutive number of cycles up to its burst transfer percentage if bus clear ($\overline{\text{BCLR}}$) is not asserted and the BCR is greater than zero. The following decode shows these percentages.

00 = IDMA gets up to 75% of the bus bandwidth.
01 = IDMA gets up to 50% of the bus bandwidth.
10 = IDMA gets up to 25% of the bus bandwidth.
11 = IDMA gets up to 12.5% of the bus bandwidth.

3.1.6 DMA Bus Arbitration

The IDMA controller uses the M68000 bus arbitration protocol to request bus mastership before entering the DMA mode of operation. The six SDMA channels have priority over the IDMA and can transfer data between any two IDMA bus cycles with $\overline{\text{BGACK}}$ remaining continuously low. Once the processor has initialized and started a DMA channel, an operand transfer request is made pending by either an external device or by using an internal request.

When the IDMA channel has an operand transfer request pending and $\overline{\text{BCLR}}$ is not asserted, the IDMA will request bus mastership from the internal bus arbiter using the internal signal IDBR (see Figure 3-12). The arbiter will assert the internal M68000 core bus request ($\overline{\text{CBR}}$) signal and will monitor the core bus grant ($\overline{\text{CBG}}$) and external $\overline{\text{BR}}$ to determine when it may grant the IDMA mastership. The IDMA will monitor the address strobe ($\overline{\text{AS}}$), $\overline{\text{HALT}}$, bus error ($\overline{\text{BERR}}$), and bus grant acknowledge ($\overline{\text{BGACK}}$) signals. These signals must be negated to indicate that the previous bus cycle has completed and the previous bus master has released the bus. When these conditions are met, the IDMA only asserts $\overline{\text{BGACK}}$ to indicate that it has taken control of the bus. When all operand transfers have occurred, the IDMA will release control of the bus by negating $\overline{\text{BGACK}}$.

Internally generated IDMA requests are affected by a mechanism supported to reduce the M68000 core interrupt latency and external bus master arbitration latency (see 3.8.5 Bus Arbitration Logic). The IDMA is forced to relinquish the bus when an external bus master requests the bus ($\overline{\text{BR}}$ is asserted) or when the M68000 core has an unmasked pending interrupt request. In these cases, the on-chip arbiter sends an internal bus-clear signal to the IDMA. In response, any operand transfer in progress will be fully completed (up to four bus cycles depending on the configuration), and bus ownership will be released.

When the IDMA regains the bus, it will continue transferring where it left off. If the core caused the bus to be relinquished, no further IDMA bus cycles will be started until IPA in the SCR is cleared. If the cause was an external request, no further IDMA bus cycles will be started while $\overline{\text{BR}}$ remains asserted. When $\overline{\text{BR}}$ is externally negated, if a transfer request is pending and IPA is cleared, the IDMA will arbitrate for the bus and continue normal operation.

3.1.7 Bus Exceptions

In any computer system, the possibility always exists that an error will occur during a bus cycle due to a hardware failure, random noise, or an improper access. When an asynchronous bus structure, such as that supported by the M68000 is used, it is easy to make provisions allowing a bus master to detect and respond to errors during a bus cycle. The IDMA recognizes the same bus exceptions as the M68000 core: reset, bus error, halt, and retry.

NOTE

These exceptions also apply to the SDMA channels except that the bus error reporting method is different. See 4.5.8.4 Bus Error on SDMA Access for further details.

address match exists within its address space and, therefore, whether to assert the chip-select line.

111 = Not supported; reserved. Chip select will not assert if this value is chosen.
110 = Value may be used.
•
•
•
000 = Value may be used.

After system reset, the FC field in BR3–BR0 defaults to supervisor program space (FC = 110) to select a ROM device containing the reset vector. Because of the priority mechanism and the EN bit, only the $\overline{CS0}$ line is active after a system reset.

NOTE

The FC bits can be masked and ignored by the chip-select logic using CFC in the OR.

Bits 12–2—Base Address

These bits are used to set the starting address of a particular address space. The address compare logic uses only A23–A13 to cause an address match within its block size. The base address should be located on a block boundary. For example, if the block size is 64k bytes, then the base address should be a multiple of 64k.

After system reset, the base address defaults to zero to select a ROM device on which the reset vector resides. All base address values default to zero on system reset, but, because of the priority mechanism, only $\overline{CS0}$ will be active.

NOTE

All address bits can be masked and ignored by the chip-select logic through the base address mask in the OR.

RW—Read/Write

0 = The chip-select line is asserted for read operations only.
1 = The chip-select line is asserted for write operations only.

After system reset, this bit defaults to zero (read-only operation).

NOTE

This bit can be masked and ignored by the read-write compare logic, as determined by MRW in the OR. The line is then asserted for both read and write cycles.

On write protect violation cycles (RW = 0 and MRW = 1), $\overline{BERR}$ will be generated if WPVE is set, and WPV will be set.

If the write protect mechanism is used by an external master, the R/$\overline{W}$ low to $\overline{AS}$ asserted timing should be 16 ns minimum.

questing the M68000 bus with $\overline{BR}$. See 3.8.6 Hardware Watchdog for further details.

3.8.5 Bus Arbitration Logic

Both internal and external bus arbitration are discussed in the following paragraphs.

3.8.5.1 Internal Bus Arbitration

The IMP bus arbiter supports three bus request sources in the following standard priority:

1. External bus master ($\overline{BR}$ pin)
2. SDMA for the SCCs (six channels)
3. IDMA (one channel)

When one of these sources desires the bus, the M68000 core will be forced off through an internal bus request signal ($\overline{CBR}$) from the bus arbiter to the M68000 core (see Figure 3-12). When the arbiter detects the assertion of the M68000 core bus grant ($\overline{CBG}$) signal, it asserts the requester's bus grant signal according to its priority. Thus, as seen externally, the SDMA and IDMA channels do not affect $\overline{BR}$ and $\overline{BG}$, but only $\overline{BGACK}$ (unless disable CPU mode is used).

The MC68302 provides several options for changing the preceding bus master priority list. The options are configured by setting the BCLM bit in the SCR and deciding whether or not the $\overline{BCLR}$ pin is used externally to cause external bus masters to relinquish the bus (see Table 3-10).

Each SCON is a 16-bit, memory-mapped, read-write register. The SCONs are set to $0004 by reset, resulting in the baud rate generator output clock rate being set to the main clock rate divided by 3. The baud rate generator output clock is always available externally, as shown in Table 5-8.

NOTE

The BRG output are 180 degrees out of phase to the TCLK and RCLK signals used by the SCC and output on the RCLK and TCLK pins.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WOMS	EXTC	TCS	RCS	CD10	CD9	CD8	CD7	CD6	CD5	CD4	CD3	CD2	CD1	CD0	DIV4

WOMS—Wired-OR

When WOMS is set, the TXD driver is programmed to function as an open-drain output and may be externally wired together with other TXD pins in an appropriate bus configuration. In this case, an external pullup resistor is required. When WOMS is cleared, the TXD pin operates normally with an active internal pullup.

NOTE

This bit is valid only in NMSI mode.

EXTC—External Clock

The EXTC bit selects whether the baud rate generator input clock source is the internal main clock (EXTC = 0) or external clock (EXTC = 1). If EXTC = 1, the external clock is taken from the TIN1 pin. Note that the single TIN1 pin can be used to supply a clock for all three baud rate generators.

TCS—Transmit Clock Source

The TCS bit selects either the baud rate generator output (TCS = 0) or the TCLK pin (TCS = 1) for the transmitter clock. If TCS = 0, then the baud rate generator output is driven onto the TCLK pin. This bit should be programmed to one if a multiplexed mode is chosen for the SCC.

After system reset, SCC hardware causes TCLK to default to an input and stay an input until a zero is written to TCS. For SCC2 and SCC3, TCLK can be derived directly from the RCLK pin as shown in Table 3-6.

RCS—Receive Clock Source

The RCS bit selects either the baud rate generator output (RCS = 0) or the RCLK pin (RCS = 1) for the receiver clock. If RCS = 0, then the baud rate generator output is driven onto the RCLK pin. This bit should be programmed to one if a multiplexed mode is chosen for the SCC.

After system reset, SCC hardware causes the RCLK to default to an input and stay an input until a zero is written to RCS.

crystal, an externally supplied UART clock on the TCLK or RCLK pins can be as high as 6.67 MHz, giving a maximum baud rate of 417 kbaud.

The baud rate using the baud rate generator is (System Clock or TIN1 clock)/(1 or 4)/(Clock Divider + 1)/16. The baud rate using the baud rate generator with an externally supplied clock to the TCLK or RCLK pins is always (TCLK or RCLK)/16.

Table 4-4 shows examples of typical bit rates of asynchronous communication and how to obtain them with the baud rate generator using an internally supplied clock.

Table 4-4. Typical Bit Rates of Asynchronous Communication

Baud Rates	15.36			16.0			16.667		
	DIV4	DIV	Actual Frequency	DIV4	DIV	Actual Frequency	DIV4	DIV	Actual Frequency
150	1	1599	150	1	1666	149.97	1	1735	150.01
300	1	799	300	1	832	300.12	1	867	300.02
600	0	1599	600	0	1666	599.88	0	1735	600.05
1200	0	799	1200	0	832	1200.48	0	867	1200.1
2400	0	399	2400	0	416	2398.08	0	433	2400.2
4800	0	199	4800	0	207	4807.69	0	216	4800.4
9600	0	99	9600	0	103	9615.34	0	108	9556.76
19200	0	49	19200	0	51	19230.8	0	53	19290.5
38400	0	24	38400	0	25	38461.53	0	26	38581.0

4.5.2.2 Synchronous Baud Rate Generator Examples

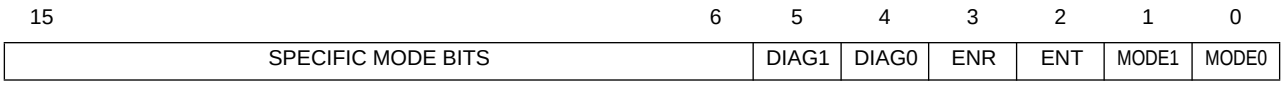
For synchronous communication (HDLC/SDLC, BISYNC, DDCMP, Transparent, and V.110), the internal clock is identical to the baud rate output. To obtain the desired rate, the user selects the appropriate system clock according to the following equation:

Baud rate = (System Clock or TIN1 Clock)/(Clock Divider + 1)/(1 or 4) according to the DIV4 bit

For example, to get the data rate of 64 kbps, the system clock can be 15.36 MHz, DIV4 = 0, and the Clock Divider = 239. Of course, a 64 kbps rate provided externally on the TCLK or RCLK pins could also be used.

4.5.3 SCC Mode Register (SCM)

Each SCC has a mode register. The functions of bits 5–0 are common to each protocol. The function of the specific mode bits varies according to the protocol selected by the MODE1–MODE0 bits. They are described in the relevant sections for each protocol type. Each SCM is a 16-bit, memory-mapped, read-write register. The SCMs are cleared by reset.



DIAG1–DIAG0—Diagnostic Mode

00 = Normal operation ($\overline{\text{CTS}}$, $\overline{\text{CD}}$ lines under automatic control)
In this mode, the CTS and CD lines are monitored by the SCC controller. The SCC controller uses these lines to automatically enable/disable reception and transmission.

ENC—Data Encoding Format

0 = Non-return to zero (NRZ). A one is a high level; a zero is a low level.
1 = Non-return to zero inverted (NRZI). A one is represented by no change in the level; a zero is represented by a change in the level. The receiver decodes NRZI, but a clock must be supplied. The transmitter encodes NRZI. During an idle condition, with the FLG bit cleared, the line will be forced to a high state.

COMMON SCC MODE BITS—See 4.5.3 SCC Mode Register (SCM) for a description of the DIAG1, DIAG0, ENR, ENT, MODE1, and MODE0 bits.

4.5.12.10 HDLC Receive Buffer Descriptor (Rx BD)

The HDLC controller uses the Rx BD to report information about the received data for each buffer. The Rx BD is shown in Figure 4-26.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET + 0	E	X	W	I	L	F	—	—	—	—	LG	NO	AB	CR	OV	CD
OFFSET + 2	DATA LENGTH															
OFFSET + 4	RX BUFFER POINTER (24-bits used, upper 8 bits must be 0)															
OFFSET +6																

Figure 4-26. HDLC Receive Buffer Descriptor

An example of the HDLC receive process is shown in Figure 4-27. This shows the resulting state of the Rx BDs after receipt of a complete frame spanning two receive buffers and a second frame with an unexpected abort sequence. The example assumes that MRBLR = 8 in the SCC parameter RAM.

The first word of the Rx BD contains control and status bits. Bits 15–10 are written by the user before the buffer is linked to the Rx BD table, and bits 5–0 are set by the CP following frame reception. Bit 15 is set by the M68000 core when the buffer is available to the HDLC controller; it is cleared by the HDLC controller when the buffer is full.

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 M68000 core is free to examine or write to any fields of the BD.
1 = The data buffer associated with the BD is empty. This bit signifies that the BD and its associated buffer are available to the HDLC controller. The M68000 core should not write to any fields of this BD after it sets this bit. The empty bit will remain set while the HDLC controller is currently filling the buffer with received data.

X—External Buffer

0 = The buffer associated with this BD is in internal dual-port RAM.
1 = The buffer associated with this BD is in external memory.

R—Ready

0 = This buffer is not currently ready for transmission. The user is free to manipulate this BD (or its associated buffer). The CP clears this bit after the buffer has been fully transmitted or after an error condition has been encountered.
1 = The data buffer has been prepared for transmission by the user (but not yet transmitted). No fields of this BD may be written by the user once this bit is set.

X—External Buffer

0 = The buffer associated with this BD is in internal dual-port RAM.
1 = The buffer associated with this BD is in external memory.

W—Wrap (Final BD in Table)

0 = This is not the last BD in the Tx BD table.
1 = This is the last BD in the Tx BD table. After this buffer has been used, the CP will transmit data from the first BD in the table.

NOTE

The user is required to set the wrap bit in one of the first eight BDs; otherwise, errant behavior may occur.

I—Interrupt

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

L—Last in Message

0 = The last character in the buffer is not the last character in the current block.
1 = The last character in the buffer is the last character in the current block. The transmitter will enter (remain in) normal mode after sending the last character in the buffer and the BCS (if enabled).

TB—Transmit BCS

This bit is valid only when the L bit is set.

0 = Transmit the SYN1–SYN2 sequence or IDLE (according to the SYNF bit in the BISYNC mode register) after the last character in the buffer.
1 = Transmit the BCS sequence after the last character. The BISYNC controller will also reset the BCS generator after transmitting the BCS sequence.

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.

BR—BCS Reset

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

TX—Tx Buffer

A buffer has been transmitted over the DDCMP channel. This bit is set no sooner than at the beginning of the transmission of the second-to-last data (or CRC) bit in the frame, if this buffer was the last in a frame. Otherwise, it is set as the last data byte is written to the transmit FIFO.

RBD—Rx Buffer

A buffer has been received on the DDCMP channel that was not a complete block.

4.5.14.13 DDCMP Mask Register

The SCC mask register (SCCM) is referred to as the DDCMP mask register when the SCC is operating as a DDCMP controller. It is an 8-bit read-write register that has the same bit format as the DDCMP event register. If a bit in the DDCMP mask register is a one, the corresponding interrupt in the event register will be enabled. If the bit is zero, the corresponding interrupt in the event register will be masked. This register is cleared upon reset.

4.5.15 V.110 Controller

V.110 is an ISDN protocol that interfaces non-ISDN terminal equipment into one of the B channels of the 2B + D basic rate ISDN S-interface. V.110 offers the ability to transmit data from non-ISDN terminal equipment over one of the B channels through an ISDN to other non-ISDN terminal equipment. A common application of V.110 rate adaption is providing a connection between RS-232 based terminal equipment over an ISDN.

Since V.110 uses the B channels, a V.110 application requires that an ISDN call first be set up over the D channel. This call setup uses the LAPD protocol, which can be implemented using another SCC in the MC68302 (see 4.5.12 HDLC Controller).

V.110 can adapt slower equipment up to the 64 kbps B channel rate. If the non-ISDN terminal equipment is synchronous, a one- or two-step process is required. If asynchronous equipment is adapted, a three-step process is required. The standard allows for the adaption of asynchronous rates up to 19.2 kbps and synchronous rates up to 56 kbps.

Only the third step (RA2) is required for adaption of synchronous 48- and 56-kbps rates commonly used by synchronous modems. RA1 and RA2 are needed for synchronous rates of 600, 1200, 2400, 4800, 7200, 9600, 12 kbps, 14.4 kbps, and 19.2 kbps. Asynchronous terminals require all three steps because rate-adapting asynchronous terminals require the additional task of compensating for slight underspeed or overspeed of the terminal with respect to the ISDN clock rate. This is why the RA0 function refers to stop-bit manipulation.

Data is transmitted over the ISDN in a defined 80-bit frame format. This frame includes all data from the terminal (including start and stop bits for the asynchronous terminals) as well as various framing and control functions.

Another rate adaption protocol, called V.120, is an alternative protocol to V.110. V.120 is an extension of the LAPD protocol and may be implemented on the MC68302 using an SCC configured in HDLC mode.

If the L bit is set, the frame ends, and the transmission of ones resumes until a new buffer is made ready. $\overline{\text{RTS}}$ is negated during this period. Regardless of whether or not the next buffer is available immediately, the next buffer will not begin transmission until achieving synchronization.

The transmit buffer length and starting address may be even or odd; however, since the transparent transmitter reads a word at a time, better performance can be achieved with an even buffer length and starting address. For example, if a transmit buffer begins on an odd-byte boundary and is 10 bytes in length (the worst case), six word reads will result, even though only 10 bytes will be transmitted.

Any whole number of bytes may be transmitted. If the REVD bit in the transparent mode register is set, each data byte will be reversed in its bit order before transmission.

If the interrupt (I) bit in the Tx BD is set, then the TX bit will be set in the transparent event register following the transmission of the buffer. The TX bit can generate a maskable interrupt.

4.5.16.2 Transparent Channel Buffer Reception Processing

When the M68000 core enables the transparent receiver, it will enter hunt mode. In this mode, it waits to achieve synchronization before receiving data. See 4.5.16.5 Transparent Synchronization for details.

Once data reception begins, the transparent receiver begins moving data from the receive FIFO to the receive buffer, always moving a 16-bit word at a time. After each word is moved to memory, the RCH bit in the transparent event register is set, which can generate a maskable interrupt, if desired. The transparent receiver continues to move data to the receive buffer until the buffer is completely full, as defined by the byte count in MRBLR. The receive buffer length (stored in MRBLR) and starting address must always be even, so the minimum receive buffer length must be 2.

After a buffer is filled, the transparent receiver moves to the next Rx BD in the table and begins moving data to its associated buffer. If the next buffer is not available when needed, a busy condition is signified by the setting of the BSY bit in the transparent event register, which can generate a maskable interrupt.

Received data is always packed into memory a word at a time, regardless of how it is received. For example, in NMSI mode, the first word of data will not be moved to the receive buffer until after the sixteenth receive clock occurs. In PCM highway mode, the same principle applies except that the clocks are only internally active during an SCC time slot. For example, if each SCC time slot is seven bits long, the first word of data will not be moved to the receive buffer until after the second bit of the third time slot, regardless of how much time exists between individual time slots.

Once synchronization is achieved for the receiver, the reception process continues unabated until a busy condition occurs, a $\overline{\text{CD}}$ lost condition occurs, or a receive overrun occurs. The busy condition error should be followed by an ENTER HUNT MODE command to the

4.7.4.1 SMC1 Receive Buffer Descriptor

The CP reports information about the received byte using this (BD).

15	14	13	12	11	10	9	8	7	0
E	L	ER	MS	—		AB	EB	DATA	

E—Empty

0 = This bit is cleared by the CP to indicate that the data byte associated with this BD is now available to the M68000 core.

1 = This bit is set by the M68000 core to indicate that the data byte associated with this BD is empty.

In GCI mode, when the IMP implements the monitor channel protocol, the IMP will wait until this bit is set by the M68000 core before acknowledging the monitor channel data. In other modes (transparent GCI and IDL), additional received data bytes will be discarded until the empty bit is set by the M68000 core.

L—Last (EOM)

This bit is valid only in GCI mode when the IMP implements the monitor channel protocol. This bit is set when the end-of-message (EOM) indication is received on the E bit.

NOTE

When this bit is set, the data byte is not valid.

ER—Error Condition

This bit is valid only in GCI mode when the IMP implements the monitor channel protocol and the L bit is set. This bit is set when an error condition occurs on the monitor channel protocol. A new byte is transmitted before the IMP acknowledges the previous byte.

MS—Data Mismatch

This bit is valid only in GCI mode when the IMP implements the monitor channel protocol. This bit is set when two different consecutive bytes are received and is cleared when the last two consecutive bytes match. The IMP waits for the reception of two identical consecutive bytes before writing new data to the receive BD.

Bits 11–10—Reserved for future use.

AB—Received A Bit

This bit is valid only in GCI mode when the monitor channel is in transparent mode.

EB—Received E Bit

This bit is valid only in GCI mode when the monitor channel is in transparent mode.

Data—Data Field

The data field contains the byte of data received by SMC1.

ters. If DTACK is generated internally, then it is an output. It is an input when the IMP accesses an external device not within the range of the chip-select logic or when programmed to be generated externally.

RMC/IOUT1—Read-Modify-Write Cycle Indication/Interrupt Output 1

This signal functions as RMC in normal operation. RMC is an output signal that is asserted when a read-modify-write cycle is executed. It indicates that the cycle is indivisible.

When the M68000 core is disabled, this pin operates as IOUT1. IOUT2–IOUT0 provide the interrupt request output signals from the IMP interrupt controller to an external CPU when the M68000 core is disabled.

IAC—Internal Access

This output indicates that the current bus cycle accesses an on-chip location. This includes the on-chip 4K byte block of internal RAM and registers (both real and reserved locations), and the system configuration registers ($0F0–$0FF). The above-mentioned bus cycle may originate from the M68000 core, the IDMA, or an external bus master. Note that, if the SDMA accesses the internal dual-port RAM, it does so without arbitration on the M68000 bus; therefore, the IAC pin is not asserted in this case. The timing of IAC is identical to that of the CS3–CS0 pins.

IAC can be used to disable an external address/data buffer when the on-chip dual-port RAM and registers are accessed, thus preventing bus contention. Such a buffer is optional and is only required in larger systems. An external address/data buffer with its output enable (E) and direction control (dir) may be placed between the two bus segments as shown in Figure 5-7. The IAC signal saves the propagation delay and logic required to OR all the various system chip-select lines together to determine when to enable the external buffers.

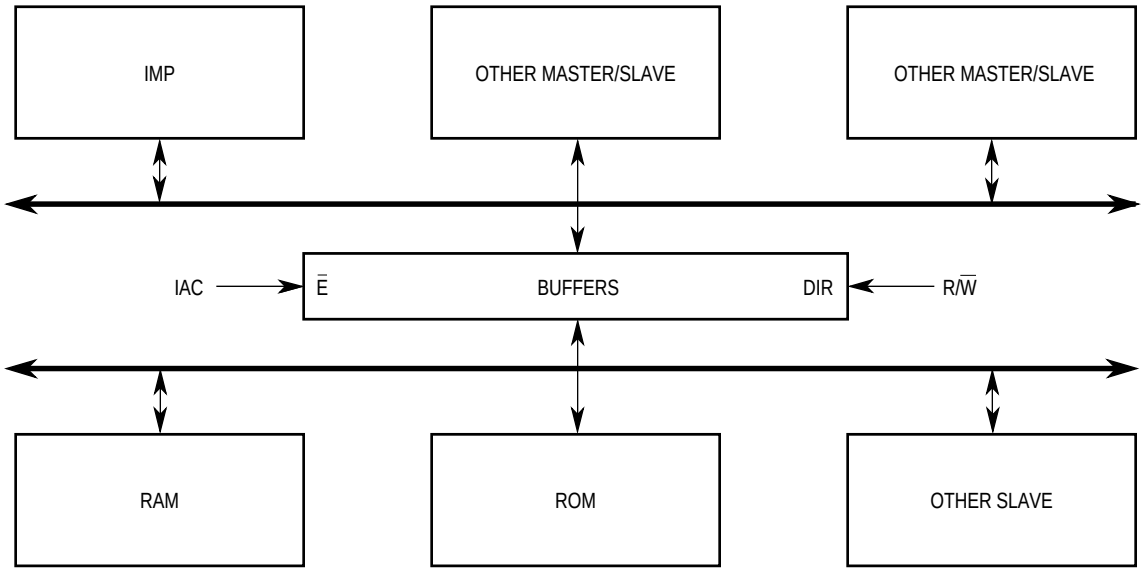


Figure 5-7. External Address/Data Buffer

6.18 AC ELECTRICAL SPECIFICATIONS—SERIAL COMMUNICATIONS PORT (see Figure 6-19).

		16.67 MHz		20 MHz		25 MHz		
Num.	Characteristic	Min	Max	Min	Max	Min	Max	Unit
250	SPCLK Clock Output Period	4	64	4	64	4	64	clks
251	SPCLK Clock Output Rise/Fall Time	0	15	0	10	0	8	ns
252	Delay from SPCLK to Transmit (see Note 1)	0	40	0	30	0	24	ns
253	SCP Receive Setup Time (see Note 1)	40	—	30	—	24	—	ns
254	SCP Receive Hold Time (see Note 1)	10	—	8	—	7	—	ns

NOTES:
1. This also applies when SPCLK is inverted by CI in the SPMODE register.
2. The enable signals for the slaves may be implemented by the parallel I/O pins.

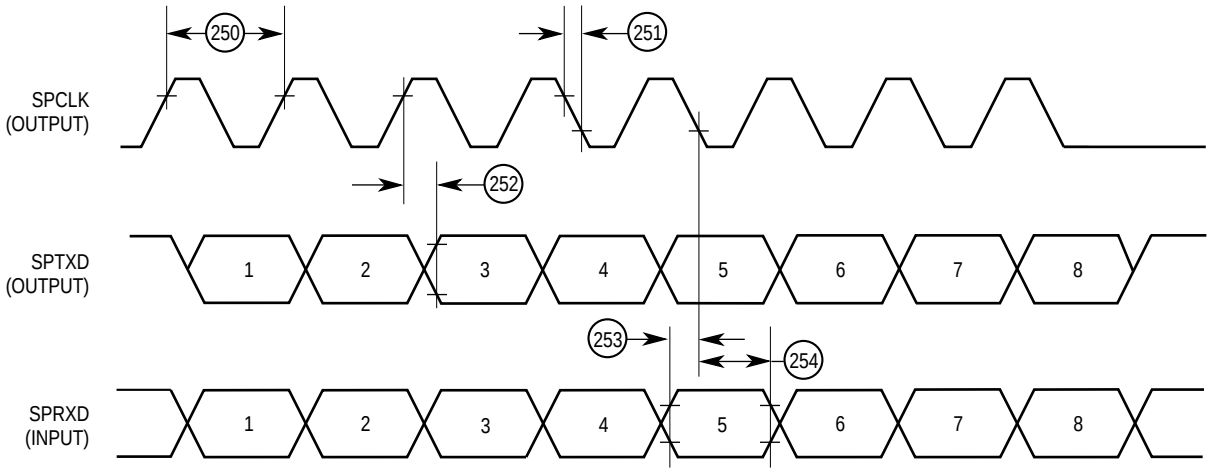


Figure 6-19. Serial Communication Port Timing Diagram

initialization corresponds to the recommended order described in 4.5.7 SCC Initialization. The second part is a set of loops waiting for data to arrive to be retransmitted out of the SCC3. The third part is the SCC3 receive interrupt handler. Transmit interrupts are masked in this example.

D.4.2 Organization of Buffers

In the MC68302, there is no such thing as an receive register (Rx) or transmit register (Tx). Rather, a flexible structure called a buffer descriptor (BD) is used. In this example, two Rx BDs and two Tx BDs are used. Each BD is set up to point to a one-byte location for data. Thus, the receiver and transmitter are double-buffered. The number of Rx or Tx BDs can be changed simply by changing the number of BDs initialized in the code (and two other lines documented in the code). However, using at least two BDs has advantages as noted in the following paragraphs.

The structure of the buffers is shown in Figure D-6.

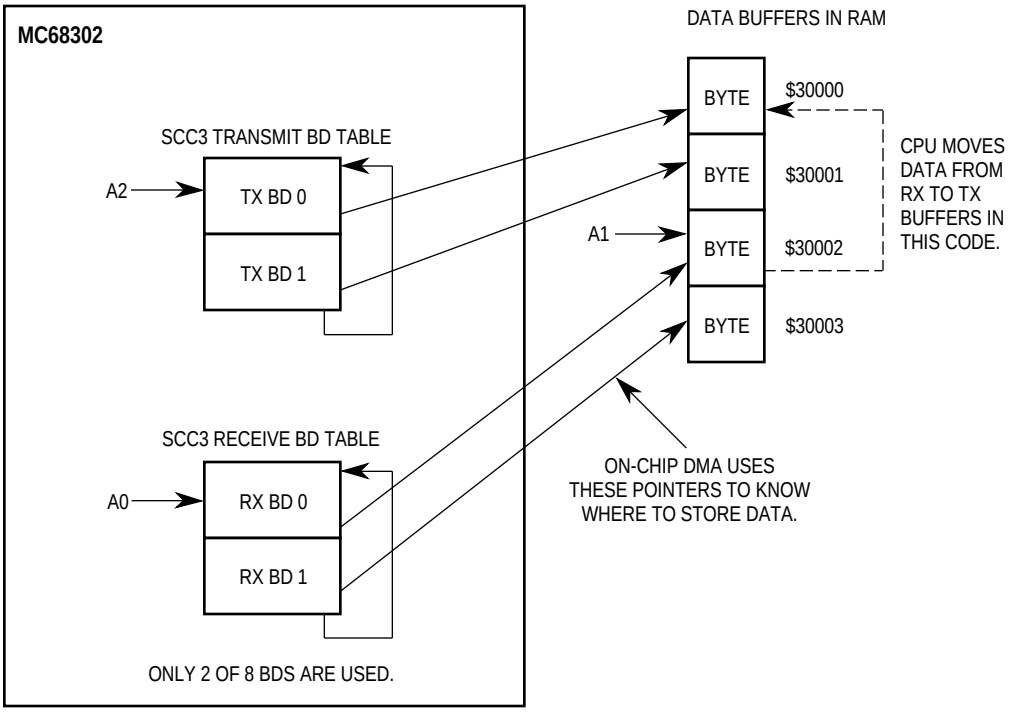


Figure D-6. Transmit and Receive BD Tables and Buffers

To make the application more general, the data buffers were located in external RAM; however, internal RAM could have been used. Each BD points to just one byte in memory as shown. Note that the data buffers do not need to be consecutive as shown.

From one to eight BDs may be used for both the transmit and receive operation. Use of eight BDs for the transmit side of SCC3 requires that the SCP and SMCs not be used. As data rates increase substantially beyond the 9600 baud of this example, the use of more BDs and more data bytes per BD becomes justified. Why were two Rx BDs and two Tx BDs chosen rather than just one each?

Since the L bit is set, once the frame ends, the synchronization process must occur once again. Also, to force resynchronizations instead of waiting for the transmission to finish, a STOP TRANSMIT command can be given, followed by a RESTART TRANSMIT command; however, the STOP TRANSMIT command will abort the current buffer.

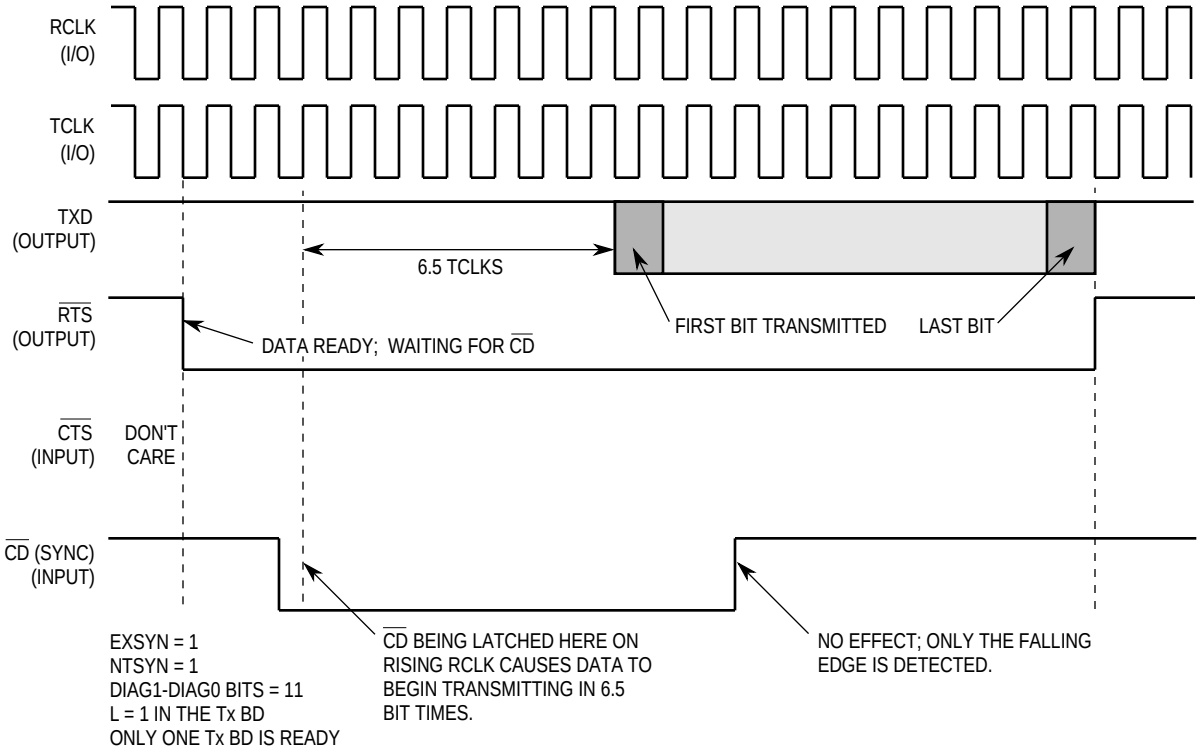


Figure D-26. Using $\overline{CD}$ (Sync) In the NMSI Transmit Case

Figure D-27 shows how $\overline{CD}$ (sync) is used in the NMSI receive case. Setting the EXSYN bit causes $\overline{CD}$ (sync) to control the reception of data. $\overline{CD}$ (sync) should be latched low on the rising clock (RCLK) of the second *bit of the frame.* (Latching $\overline{CD}$ (sync) during the 2nd bit of the frame allows external BISYNC sync detection logic, which also uses EXSYN, extra time to present the external sync to the SCC.) Once synchronization is achieved, it will never be lost unless an ENTER HUNT MODE command is given, a receive overrun occurs, or the receiver is disabled and re-enabled (ENR bit is cleared, ENTER HUNT MODE command is issued, and ENR is set). Once synchronization is lost, a new frame can be resynchronized using $\overline{CD}$ (sync).

Notice that we have been discussing the receive and transmit cases separately. The receive and transmit halves of the SCC really are separate and distinct; however, in transparent mode, the receive and transmit halves of the SCC share the $\overline{CD}$ (sync) pin, which is not true in normal NMSI.

In the Tx BD, the last (L) bit should be set, and the TB, BR, TD, and TR bits should be cleared. In each Rx BD, the CR bit, which indicates a bad BCS, should be ignored.

If all the frames are of a fixed length, you do not need to use ETX. Instead, disable the whole control character table, and set the MRBLR to the frame length. If MRBLR = 2, for example, then you can send and receive the following frame types:

syn-syn-Data-Data-syn-syn-syn-syn-Data-Data-syn-syn

where syn is a one-byte sync character that cannot be sent as data.

To be able to send the sync character within the data stream requires full BISYNC capabilities in a mode called BISYNC transparent, which is not discussed in this subsection.

D.8.6.2 TRANSYNC MODE. In the normal transparent mode examples discussed previously, both the NTSYN and the EXSYN bits were set in the SCM register. Also, different ways of using BISYNC mode have been described in which both the NTSYN and EXSYN bits are cleared. However, what happens if you set NTSYN and clear EXSYN? The answer is a combination of transparent and BISYNC modes that is referred to here as TRANSYNC.

On the transmission side, normal transparent operation takes place with no sync characters transmitted. On the receive side, however, reception will not be synchronized until the pattern in the DSR is matched on the line. In other words, the $\overline{CD}$ (sync) function is eliminated on transmit and is replaced with the DSR matching function on receive. Recall that $\overline{CD}$ and $\overline{CTS}$ can still control transmission and reception in TRANSYNC mode if the DIAG1-DIAG0 bits are set for normal mode and not software operation mode.

NOTE

When NTSYN is cleared and EXSYN is set, the result is normal BISYNC mode except that the external synchronization function, $\overline{CD}$ (sync), is required for proper reception. Syncs are transmitted in this mode, but are not required on receive. This is the opposite of TRANSYNC mode.

D.8.7 Gating Clocks in NMSI Mode

If the behavior of CTS and CD (sync) are not what is needed for an application, there is always the possibility of gating clocks to the SCC. The term “gating clocks” usually means providing clocks to an SCC only while it is in the act of transmitting or receiving, but at no other time. Gating clocks is a requirement in some multidrop applications and can be useful for many special applications. Gating clocks is only possible if the clocks are inputs to the SCC since the internal SCC baud rate generators do not support gating clocks.

The gating of clocks can provide extra control over the transmission and reception of data, albeit with extra logic external to the MC68302. The SCCs are designed with static logic; thus, the clock signal may be held in a constant high/low state for any period of time. Whenever clocks are provided externally (and especially when they are gated), care should be taken to avoid glitches, excessive ringing, and very long rise/fall times in a very noisy environment. If the minimum clock high/low time is violated, erratic operation can result, which can cause an SCC to immediately transition to an error state such as underrun or overrun.

SDC2—Serial Data Strobe Control 2

0 = SDS2 signal is asserted during the B2 channel.
1 = SDS1 signal is asserted during the B2 channel.

SDC1—Serial Data Strobe Control 1

0 = SDS1 signal is asserted during the B1 channel.
1 = SDS2 signal is asserted during the B1 channel.

B2RB, B2RA—B2 Channel Route in IDL/GCI Mode or CH-3 Route in PCM Mode

00 = Channel not supported.
01 = Route channel to SCC1.
10 = Route channel to SCC2 (a MSC2 is cleared).
11 = Route channel to SCC3 (if MSC3 is cleared).

B1 RB, B1 RA—B1 Channel Route in IDL/GCI Mode or CH-2 Route in PCM Mode

00 = Channel not supported.
01 = Route channel to SCC1.
10 = Route channel to SCC2 (a MSC2 is cleared).
11 = Route channel to SCC3 (if MSC3 is cleared).

DRB, DRA—D Channel Route in IDL/GCI Mode or CH-1 Route in PCM Mode

00 = Channel not supported.
01 = Route channel to SCC1.
10 = Route channel to SCC2 (a MSC2 is cleared).
11 = Route channel to SCC3 (if MSC3 is cleared).

MSC3—SCC3 Connection

0 = SCC3 is connected to the multiplexed serial interface.
1 = SCC3 is not connected to the multiplexed serial interface.

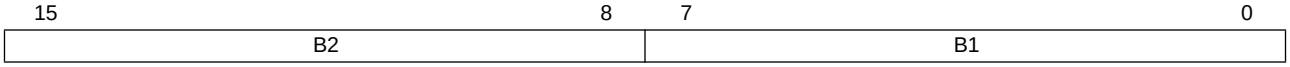
MSC2—SCC2 Connection

0 = SCC2 is connected to the multiplexed serial interface.
1 = SCC2 is not connected to the multiplexed serial interface.

MS1, MS0—Mode Supported

00 = NMSI mode.
01 = PCM mode.
10 = IDL mode.
11 = GCI interface.

E.1.1.1.3 Serial Interface Mask Register (SIMASK). This 16-bit register is located at offset $8B2. The SIMASK register is used to configure which bits on the B1 and B2 channels are used in the GCI and IDL modes. Bit 0 of SIMASK is the first bit transmitted and received on B1.



E.1.1.4.1 Receive BD Control/Status Word. To initialize the buffer, the user should write bits 15-12 and clear bits 11-10 and 5-0. The IMP clears bit 15 when the buffer is closed and sets bits 5-0 depending on which error occurred.

E—Empty

0 = This data buffer is full or has been closed due to an error condition.
1 = This data buffer is empty; must be set by the user to enable reception into this buffer.

X—External Buffer

0 = The data buffer associated with this BD is in internal dual-port RAM.
1 = The data buffer associated with this BD is in external memory.

W—Wrap (final BD in table)

0 = This is not the last BD in the receive BD table.
1 = This is the last BD in the receive BD table.

I—Interrupt (The RXF bit in the event register is set when a complete frame is received, independent of the I bit.)

0 = The RXB bit in the event register is not set when this buffer is closed.
1 = The RXB bit (or RXF bit, if this is the last buffer in a frame) in the event register is set when this buffer is closed.

L—Last in Frame

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

F—First in Frame

0 = This buffer is not the first buffer in a frame.
1 = This buffer is the first buffer in a frame.

Bits 9–6—Reserved for future use

LG—Rx Frame Length Violation

0 = No frame length violation occurred.
1 = A frame length violation was detected. Up to the number of bytes specified in the maximum frame length will be written to the buffer (or buffers, if multiple buffers per frame).

NO—Rx Nonoctet Aligned Frame

0 = An octet aligned frame was received.
1 = A nonoctet aligned frame was received.

AB—Rx Abort Sequence

0 = No abort was received.
1 = A minimum of seven ones was received during frame reception.

Table E-1. (a)Transparent Programming Model
Receive and Transmit Buffer Descriptors for SCCx

Initialized by User	Offset Hex	Name	Initialized by User	Offset Hex	Name
Yes	00	Rx BD 0 Control/Status	Yes	40	Tx BD 0 Control/Status
	02	Rx BD 0 Data Count	Yes	42	Tx BD 0 Data Count
Yes	04	Rx BD 0 Data Pointer (High Word)	Yes	44	Tx BD 0 Data Pointer (High Word)
Yes	06	Rx BD 0 Data Pointer (Low Word)	Yes	46	Tx BD 0 Data Pointer (Low Word)
Yes	08	Rx BD 1 Control/Status	Yes	48	Tx BD 1 Control/Status
	0A	Rx BD 1 Data Count	Yes	4A	Tx BD 1 Data Count
Yes	0C	Rx BD 1 Data Pointer (High Word)	Yes	4C	Tx BD 1 Data Pointer (High Word)
Yes	0E	Rx BD 1 Data Pointer (Low Word)	Yes	4E	Tx BD 1 Data Pointer (Low Word)
Yes	10	Rx BD 2 Control/Status	Yes	50	Tx BD 2 Control/Status
	12	Rx BD 2 Data Count	Yes	52	Tx BD 2 Data Count
Yes	14	Rx BD 2 Data Pointer (High Word)	Yes	54	Tx BD 2 Data Pointer (High Word)
Yes	16	Rx BD 2 Data Pointer (Low Word)	Yes	56	Tx BD 2 Data Pointer (Low Word)
Yes	18	Rx BD 3 Control/Status	Yes	58	Tx BD 3 Control/Status
	1A	Rx BD 3 Data Count	Yes	5A	Tx BD 3 Data Count
Yes	1C	Rx BD 3 Data Pointer (High Word)	Yes	5C	Tx BD 3 Data Pointer (High Word)
Yes	1E	Rx BD 3 Data Pointer (Low Word)	Yes	5E	Tx BD 3 Data Pointer (Low Word)
Yes	20	Rx BD 4 Control/Status	Yes	60	Tx BD 4 Control/Status
	22	Rx BD 4 Data Count	Yes	62	Tx BD 4 Data Count
Yes	24	Rx BD 4 Data Pointer (High Word)	Yes	64	Tx BD 4 Data Pointer (High Word)
Yes	26	Rx BD 4 Data Pointer (Low Word)	Yes	66	Tx BD 4 Data Pointer (Low Word)
Yes	28	Rx BD 5 Control/Status	Yes	68	Tx BD 5 Control/Status
	2A	Rx BD 5 Data Count	Yes	6A	Tx BD 5 Data Count
Yes	2C	Rx BD 5 Data Pointer (High Word)	Yes	6C	Tx BD 5 Data Pointer (High Word)
Yes	2E	Rx BD 5 Data Pointer (Low Word)	Yes	6E	Tx BD 5 Data Pointer (Low Word)
Yes	30	Rx BD 6 Control/Status	Yes	70	Tx BD 6 Control/Status
	32	Rx BD 6 Data Count	Yes	72	Tx BD 6 Data Count
Yes	34	Rx BD 6 Data Pointer (High Word)	Yes	74	Tx BD 6 Data Pointer (High Word)
Yes	36	Rx BD 6 Data Pointer (Low Word)	Yes	76	Tx BD 6 Data Pointer (Low Word)
Yes	38	Rx BD 7 Control/Status	Yes	78	Tx BD 7 Control/Status
	3A	Rx BD 7 Data Count	Yes	7A	Tx BD 7 Data Count
Yes	3C	Rx BD 7 Data Pointer (High Word)	Yes	7C	Tx BD 7 Data Pointer (High Word)
Yes	3E	Rx BD 7 Data Pointer (Low Word)	Yes	7E	Tx BD 7 Data Pointer (Low Word)

NOTE: The offset is from the MC68302 base address + ($400 for SCC1, $500 for SCC2, or $600 for SCC3)

.

Table E-1 (b). Transparent Programming Model (Continued)
General Parameter and Transparent Protocol-Specific RAM for SCCx

Initialized by User	Offset Hex	Name		Initialized by User	Offset Hex	Name
Yes	80	RFCR	TFCR		A2	Reserved
Yes	82	MRBLR			A4	Reserved
	84	Rx Internal State			A6	Reserved
	86	Reserved	Rx Internal Buffer No.		A8	Reserved
	88 8A	Rx Internal Data Pointer (High Word) Rx Internal Data Pointer (Low Word)			AA	Reserved
	8C	Rx Internal Byte Count			AC	Reserved
	8E	Rx Temp			AE	Reserved
	90	Tx Internal State			B0	Reserved
	92	Reserved	Tx Internal Buffer No.		B2	Reserved
	94 96	Tx Internal Data Pointer (High Word) TX Internal Data Pointer (Low Word)			B4	Reserved
	98	Tx Internal Byte Count			B6	Reserved
	9A	Tx Temp			B8	Reserved
	9C	Reserved			BA	Reserved
	9E	Reserved			BC	Reserved
	A0	Reserved			BE	Reserved

NOTE: The offset is from the MC68302 base address + ($400 for SCC1, $500 for SCC2, or $600 for SCC3).