Application Note

The Zilog® Datacom Family with the 80186 CPU

AN009703-0508

Abstract

Zilog's customers need a way to evaluate its serial communications controllers with a central CPU. This application note describes how Zilog's Datacom family interfaces and communicates with Intel's 80186 CPU on this evaluation board. The evaluation board helps you to evaluate Zilog's data communication controllers in an Intel® environment.

The most advanced and complex component of the serial family is the Integrated Universal Serial Controller (IUSC). This application note highlights how the IUSC adapts to the 80186 CPU with minimum difficulty and maximum bus and functional flexibility.

General Description

The evaluation board includes the following hardware:

- Intel 80186 Integrated 16-Bit Microprocessor.
- Zilog Z16C32 Integrated Universal Serial Controller (IUSC™).
- Zilog Z16C33 Monochannel Universal Serial Controller (MUSC™).
- Zilog Z16C35 Integrated Serial Communications Controller (ISCC™).
- Zilog Z85230 Enhanced Serial Communications Controller (ESCC™) or SCC.
- Two 28-pin EPROM sockets, suitable for 2764's through 27512's.
- Six 32-pin (or 28-pin) SRAM sockets, suitable for 32K x 8 or 128K x 8 devices.
- Four Altera EPLD circuits comprising the glue logic. See Figure 1 through Figure 4 on page 21.
- RS-232 and RS-422 line drivers and receivers.
- Pin headers for configuring and interconnecting the above hardware to serial applications.

Note: *All signals with an overline are active Low. For example, in R/$\overline{W}$ (WRITE is active Low); in $\overline{R}$/W (Read is active Low).*

Table 1 lists the conventional descriptions for the power connections.

Table 1. Power Connections

Connection	Circuit	Device
Power	V_{CC}	V_{DD}
Ground	GND	V_{SS}

Processor

The 80186 CPU can operate at speeds up to 16 MHz. To use the CPU clock for accurate serial bit clocking, a 9.8304 MHz CPU clock can be used. The crystal connected to the processor is 2X the operating frequency.

The processor's 1 MB address space is well-filled, if the maximum RAM complement is installed. Of the integrated Chip Select outputs provided by the 80186, the $\overline{UCS}$ is used for the EPROMs and the $\overline{PCS6}$-$\overline{PCS0}$ outputs are used for the datacom controllers. A hardware address decoder is used for the SRAMs instead of the 80186's $\overline{LCS}$ and $\overline{MCS3}$-$\overline{MCS0}$ outputs because the RAMs must be accessible to the on-chip DMA function of the ISCC and IUSC as well as the 80186 CPU.

The 80186 CPU does not decodes addresses from external bus masters. Both 8-bit and 16-bit accesses are provided for RAM. The EPROMs are only accessible to the 80186 CPU.

The 80186's mid-range memory chip select feature (specifically, the $\overline{\text{MCS2}}$ output) provides software a way to hardware Reset the ISCC, IUSC, and (M)USC. This allows your program to operate as if it were in a target system starting from Reset, including the initial write to the Bus Configuration Register (BCR).

The 80186's two integrated DMA channels can be used for any of the four or six serial data streams in the B side of the (E)SCC and the (M)USC.

The DMA EPLD derives requests for the 80186's two DMA channels from six inputs, two each for (E)SCC channel B and the one or two channels in the (M)USC.

It asserts DREQ0 and DREQ1 (High) if any of the inputs for that channel is Low, and the 80186 is not performing an Interrupt Acknowledge cycle.

Jumper blocks J22, J23, J24, and J29 control the assignment of the 80186's internal DMA controllers, including provision for a clipped Tx request which is needed if a standard SCC is installed in place of the ESCC. Table 2 lists the 80186 DMA Jumper Connections and various possibilities.

If more than one channel among the ESCC B and (M)USC are enabled for one of the 80186's internal DMA channels, software must ensure that only one of the enabled devices makes requests during a given block transfer. This can be done by leaving an entire Receiver or Transmitter idle or disabled, or by programming the device so that the DMA request is not output on the pin.

Table 2. 80186 DMA Jumper Connections

DMA Channel	Function	Install This Jumper
0	(E)SCC B Rx	J23-1 to J23-2
0	MUSC Rx or USC A Rx	J22-1 to J22-2
0	MUSC Tx or USC A Tx	J22-4 to J22-2
0	USC B Rx	J29-1 to J29-2
0	USC B Tx	J29-4 to J29-2
1	ESSCC B Tx	J24-1 to J24-3
1	(E)SCC B Tx w/early release	J24-1 to J24-2
1	MUSC Rx or USC A Rx	J22-1 to J22-3
1	MUSC Tx or USC A Tx	J22-4 to J22-1
1	USC B Rx	J29-1 to J29-3
1	USC B Tx	J29-4 to J29-3

The ISCC and IUSC handle their own DMA transfers through the 80186's HOLD/HLDA facility.

Note: *Either a Z16C33 MUSC or a Z16C30 USC can be installed in socket U5. If this is the case, references to (M)USC in the following discussion may mean the USC in its entirety or just channel A. Which one should be clear from the context.*

The inputs and outputs associated with the processor's integrated counter/timer facility are brought to the pin header labelled J26 so that they can be used in applications as listed in Table 3.

Table 3. Counter/Timer Signal Locations

J26 Pin	Signal
1	Timer In 1
2	Timer Out 1
3	Timer In 0
4	Timer Out 0
5	N/C
6	Ground

The 80186's integrated interrupt controller is bypassed in favor of the Zilog® interrupt daisy-chain structure.

Push button switches are provided for Reset and Non-Maskable Interrupt (NMI). A method to generate an NMI, in response to a start bit received from the user's PC or terminal, is also provided. The first transmitted Start bit on the RS-232. Console connector J1, after a Reset, also produces an NMI. This feature can be used to find which serial controller channel is connected to the Console connector.

Address Map

EPROM is located at the highest addresses, and its size is programmable in the 80186 CPU for the $\overline{\text{UCS}}$ output. The addresses of the Datacom controllers are programmed in the 80186 for the $\overline{\text{PCS6}}$-$\overline{\text{PCS0}}$ outputs, as a block of 128x7=896 Bytes starting at a 1 KB boundary. The block can be in I/O space or in a part of memory space that is not used for SRAM or EPROM. The starting 1 KB boundary is called Physical Block Address (PBA) in the following sections. RAM extends upwards from address 0.

The address map using 128K x 8 SRAMS and 64K x 8 EPROMS are listed in Table 4.

Table 4. Suggested Address Map

Function	Address
RAM	00000-BFFFF
(E)SCC	D8000, 2, 4, 6, or D8000-D803E (even address only)
ISCC	D8080-D80FE (even address only)
(M)USC	D8100-D81FF
IUSC	D8200-D837F
ISCC-IUSC-(M)USC Reset	DB000-DB7FF (if enabled)
27512 EPROM	E0000-FFFFF

EPROM

Two 28-pin EPROM sockets are provided. Both sockets must be populated to handle the 80186's 16-bit instruction fetches. Jumper header J18 allows the sockets to be compatible with 2764s, 27128s, 27256s, or 27512s. Jumper J18 is jumpered at the factory to match the EPROMs provided. For 27512s only, jumper J18-J2 to J18-J3 and leave J18-J1 open. For 2764s, 27128s, or

27256s, jumper J18-2 to J18-1 and leave J18-3 open.

Note: *J18 connects Pin 1 of both sockets to either A16 or V_{CC}. For 2764s, 27128s, and 27256s, Pin 1 is V_{PP} which may require a high voltage and/or draw more current than a normal logic input.*

For 2764s and 27128s, a similar jumper might be provided in some designs for pin 27 ($\overline{PGM}$). As long as the address for $\overline{UCS}$ is programmed as described in the next paragraph, A15 (which is connected to pin 27) is High whenever $\overline{UCS}$ is Low, so that 2764s and 27128s operate correctly.

The first code executed after Reset must program the 80186s Chip Select Control Registers to set up the address ranges for which outputs like $\overline{UCS}$ and $\overline{PCS6}$-$\overline{PCS0}$ are asserted. In particular, the UMCS register (address A0H within the 80186's Peripheral Control Block) must be programmed to correspond to the size of the EPROMs used as listed in Table 5.

Table 5. EPROM Address Ranges

UMCS Value	EPROM	Address Range
2764	FC3C	FC000-FFFFF
27128	F83C	F8000-FFFFF
27256	F03C	F0000-FFFFF
27512	E03C	E0000-FFFFF

The three LSBs of the above UMCS values are all 100, which signifies no external Ready/WAIT is used and no wait states are required. If the EPROMS are not fast enough for no-wait-state operation, making the three LSBs 101, 110, or 111 extends EPROM cycles by 1, 2, or 3 wait states, respectively.

RAM

Six 32-pin sockets are provided. These sockets must be populated in pairs, starting with the lower-numbered sockets to allow for 16-bit accesses. V_{CC} is provided at both Pin 32 and Pin 30 so that 28-pin 32K x 8 SRAMs can be installed in Pins 3-30 of the sockets. Jumper block J19 allows decoding of the Chip Select signals from A17-A16 for 32K x 8 SRAMs or from A19-A18 for 128K x 8 SRAMS.

Table 6 lists the 6 standard memory populations.

Table 6. Standard Memory Populations

One pair of 32K x 8 devices	64 KB at 00000-0FFFF
Two pair of 32K x 8 devices	128 KB at 00000-1FFFF
Three pair of 32K x 8 devices	192 KB at 00000-2FFFF
One pair of 128K x 8 devices	256 KB at 00000-3FFFF
Two pair of 128K x 8 devices	512 KB at 00000-7FFFF
Three pair of 128K x 8 devices	768 KB at 00000-BFFFF

J19 is factory set according to the size of the SRAMs provided.

For 32K x 8 SRAMs, jumpers are installed between J19-J2 and J19-J3, and between J19-J5 and J19-J6, with J19-J1 and J19-J4 left open.

For 128K x 8 SRAMs, jumpers are installed between J19-J1 and J19-J2, and between J19-J4 and J19-J5, with J19-J3 and J19-J6 left open.

32K x 8 SRAMs have cyclic/redundant addressing starting at 40000, 80000, and C0000. The only configuration in which this causes problems is with three pairs of 32K x 8 SRAMs and 275122

EPROMs. In this case, there is a conflict in the range E0000-EFFFF.

This conflict can be avoided by any of the following methods:

- Using two pairs of 32K x 8 SRAMs
- Using one pair of 128K x 8 SRAMs
- Using 27256 EPROMs, or
- Using 27512 EPROMs but programming the size of $\overline{\text{UCS}}$ like they are 27256s

Since the $\overline{\text{LCS}}$ output of the 80186 is not used, the LCMS register in the 80186 is not written with any value.

Programming the Peripheral Chip Selects

The 80186 allows the $\overline{\text{PCS6}}$-$\overline{\text{PCS0}}$ pins, which in this case select the various Datacom controllers, to be asserted for a selected 896-Byte block of addresses. The block may reside in either memory or I/O space depending on the values programmed into the PACS and MPCS registers, locations A4H and A8H of the 80186's Peripheral Control Block, respectively.

The choice of address space depends on the needs of the customer's application and the configuration of software supplied with the board as listed in Table 7. The 81 in the MS Byte of the MPCS values (see Table 7) makes each MCS3-MCS0 pin correspond to a 2 KB block of addresses in memory space. The actual active pin addresses are determined by the value written into the MMCS register, location A6H of the 80186 Peripheral Control Block.

The three LSBs of the PACs value specify the Ready/WAIT handling for the $\overline{\text{PCS3}}$-$\overline{\text{PCS0}}$ lines which select the (E)SCC, ISCC, and (M)USC.

The three LSBs of the MPCS value specify the Ready/WAIT handling for the $\overline{\text{PCS4}}$-$\overline{\text{PCS6}}$ lines, which select the IUSC. Both fields are shown here with the LSB's 000, signifying that the 80186 should honor a WAIT on the external Ready/WAIT signal, but that it should not provide any minimum wait.

Programming Mid-Range Memory to Reset ISCC, IUSC, and (M)USC

A Reset puts the ISCC, IUSC, and (M)USC in a state in which the first write to each device implicitly goes to a Bus Configuration Register (BCR) that controls the device's basic bus operation. The BCR is not accessible thereafter.

This board can serve as a complete development environment for your software. It includes a means where software (that is, the debug monitor) can assert the $\overline{\text{RESET}}$ input of these three devices. Specifically, assertion of the $\overline{\text{MCS2}}$ output of the 80186 such a Reset.

Table 7. Three Standard Alternatives for Serial Controller Addressing

Basic Requirement	Base Address (BPA)	PACS Value	MPCS Value
I/O Space	8000	0838	81B8
Memory Space, 32K x 8 SRAMs Used	38000	3838	81F8
Memory Space, 128K x 8 SRAMs Used	D8000	D838	81F8

Table 8 lists suggested MMCS values as a function of the RAM chip size, and the corresponding range of addresses for which any read or write access causes the three controllers to be reset.

Table 8. Address Ranges for Reset

RAM Size	MMCS Value	Address Range for which ISCC, IUSC, and (M)USC are Reset
32K x 8	3BFF	3B000-3B7FF
128K x 8	DBFF	DB000-DB7FF

The three LSBs of the above MMCS values are 111 so that the longest possible Reset pulse is generated when any of the locations in the indicated range are accessed.

Note: *If this feature is not required, it can be disabled by not programming the MMCS register.*

Interrupt Daisy Chain (Priority) Order

Jumper block J25 selects whether the (E)SCC device is at the start or the end of the interrupt daisy chain.

Table 9. Priority Order

To make the interrupt priority be:	Jumper J25 as follows
(E)SCC highest, IUSC, ISCC, (M)USC lowest	J25-J2 to J25-J3, J25-J4 to J25-J5 (J25-J1, J25-JX open)
IUSC highest, ISCC, MUSC, (E)SCC lowest	J25-J1 to J25-J2, J25-J3 to J25-J4 (J25-J5, J25X open)
IUSC highest, ISCC, USC, (E)SCC lowest	J25X to J25-2, J25-J3 to J25-J4 (J25-J1, J25-J5 open)

This variability is provided in part because early versions of the Z85230 ESCC had trouble passing an interrupt acknowledge down the daisy chain if it occurred to a lower-priority device's request just as the ESCC was starting to make its own request. Current 85230's do not have this problem.

(E)SCC

Socket U2 can be configured for either ESCC or SCC, and for versions that use either multiplexed address and data. Jumper blocks J20 and J21 select certain signals accordingly. For a part with multiplexed addresses and data (80x30), Jumper J20-J1 to J20-J2 leaving J20-J3 open and jumper J21-J1 to J21-J2, J21-J4 to J21-J5 leaving J21-J3 and J21-J6 open. With such a part, software can directly address the (E)SCC's registers and writing register addresses to Write Register 0 (WR0) is not required.

For a part having a non-multiplexed bus (85x30), jumper J20-J2 to J20-J3, J21-J2 to J21-J3, J21-J5 to J21-J6 leaving J20-J1, J21-J1, and J21-J4 open. In this case, software must handle the (E)SCC by writing register addresses into its WR0 to access any register other than WR0, RR0, or the data registers. Channel A and Channel B can be handled on a polled or interrupt-driven basis. Channel A of the (E)SCC connects the user's PC or terminal for use with the Debug Monitor included in this evaluation kit.

Channel B (but not A) can be handled on a DMA basis using the 80186's internal DMA channels, or on a polled or interrupt driven basis.

Jumper block J23 allows Channel B's $\overline{W}/\overline{REQB}$ output to be used for either a Wait function or a Receive DMA Request function. To use the out for Wait, jumper J23-J2 to J23-J3 and leave J23-J1 open. The Wait function only if the software has to delay completion of a Read from the (E)SCC Receive Data Register until data is available, and/or has to delay completion of a Write to the Transmit Data Register until the previously written character is transferred to the Transmit Shift Register. These modes are alternatives for checking the corresponding status flags and can be used to achieve operating speeds higher than those possible with traditional polling, although not as fast as the speeds possible with a DMA approach. To use the $\overline{W}/\overline{REQB}$ output as a Receive DMA Request, jumper J23-J1 to J23-J2 and leave J23-J3 open.

Jumper block J24 determine's how Channel B's $\overline{DTR}/\overline{REQB}$ output is used. To use this output for the Data Terminal Ready function, jumper J24-J3 to J24-J4 and leave J24-J1 and J24-J2 open. To use this output directly as a Transmit DMA Request (using the (E)SCC's early release capability), jumper J24-J1 to J24-J3 and leave J24-J2 and J24-J4 open. To drive the Transmit DMA Request with a clipped version of the signal that is forced High earlier than a standard SCC drives it High, jumper J24-J1 to J24-J2 and leave J24-J3 and J24-J4 open.

The `SCC EPLSD` handles the (E)SCC's signaling requirements. The EPLD configures the (E)SCC socket's Pins 35 and 36 for either a multiplexed or non-multiplexed part, based on whether J20 is jumpered to connect the 80186 ALE signal to one of the input pins. If the device detects high-going pulses on this input, it drives the corresponding low-going Address Strobe pulses onto (E)SCC Pin 36.

If the SCC EPLD's Pin 9 stays at ground, the part drives Read strobes onto Pin 36 and drives delayed Write strobes onto Pin 35, for a non-multiplexed 85x30 device.

While the ESCC's relaxed timing capability allows the 80186's $\overline{WR}$ output to be connected directly to the $\overline{WR}$ input of a non-multiplexed ESCC, the SCC EPLD delays start of the SCC's write cycle until write data is valid, even though this is not necessary for an ESCC.

The ESCC EPLD also generates the clipped DMA-request mentioned in connection with J24, and logically ORs Reset onto Pins 35 and 36. The device also tracks the two IACK cycles provided by the 80186 for each Interrupt Acknowledge cycle. For a multiplexed address/data port, it drives the address strobe only on the first cycle and provides the $\overline{RD}$ or $\overline{DS}$ pulse needed by the (E)SCC only on the second cycle. The `DMA EPLD` provides the INTACK signal needed by the (E)SCC.

The (E)SCC is only accessible at even addresses. For a non-multiplexed part (85x30), the four register locations (see Table 10) are repeated throughout the even addresses from (PBA) the (PBA) +126.

Table 10. Register Locations

(PBA), (PBA)+8....(PBA)+120	Channel B Command/ Status Register
(PBA)+2, +10.....(PBA)+122	Channel B Data Register
(PBA)+4, +12.....(PBA)+124	Channel A Command/ Status Register
(PBA)+6, +14.....(PBA)+126	Channel A Data Register

For a multiplexed part (80x30), the Select Shift Left command (`D1-D0=11`) must be written to Channel B's WR0 before any other registers are accessed.

Then, the basic register map occurs twice in the even addresses from (PBA) through (PBA)+126 as listed in Table 11.

Table 11. Register Map

(PBA), (PBA)+2....(PBA)+30	Channel B Registers 0-15
(PBA)+32, +34.....(PBA)+62	Channel A Registers 0-15
(PBA)+64, +66.....(PBA)+94	Channel B Registers 0-15
(PBA)+96, +98.....(PBA)+126	Channel A Registers 0-15

The redundant addressing of the (E)SCC is used to control a feature which can be used by software to allow you to interrupt software execution from a keyboard. If the (E)SCC is read at an address with `A6-A5=11` (for a multiplexed part, this means in the higher-addressed A Channel) a mode is set in which a Low on the console Received Data line (Start bit on pin 3 of the J1 connector) causes a Non-Maskable Interrupt on the 80186. The mode is cleared by Reset or when the (E)SCC is read at an address with A6-A5=10 (on a multiplexed part, in the higher addressees B Channel). The NMI handler should do the latter to prevent subsequent data bits on the Received Data line from causing further NMIs.

ISCC

Since 80186 processor provides multiplexed addresses and data, the ISCC is configured to use the addresses on the AD lines. Software can address the various ISCC registers directly, and need not be concerned with writing register addressees into the indirect address fields of the ISCC's WR0 and CCAR.

As the ISCC includes four DMA channels, its Channel A and B Transmitters and Receivers can be handled on a polled, interrupt-driven, and/or DMA basis, in any combination.

Since the ISCC can only be programmed as an 8-bit device on the AD7-AD0 lines, it occupies only the even addressed bytes within its address range, [(PBA)+128 through (PBA)+190]. Details of this transaction are as follows:

- The High induced by a pull-up resistor on the ISCC's A/B input selects the WAIT protocol on the $\overline{\text{WAIT}}/\overline{\text{RDY}}$ pin, which corresponds to how the 80186 functions, in subsequent register accesses, the AB selection is taken from A5 of the multiplexed address.
- A Low on the ISCC's SCC/$\overline{\text{DMA}}$ input, which is connected to A6, is required by the internal logic of the ISCC. This is why the BCR write is restricted to the first half of the ISCC's address range.
- As with all transactions between 80186 and ISCC, the address must be even because the ISCC only accepts slave-mode data on the AD7-AD0 pins.
- The MSB of the data (D7) is 1 to enable the Byte Swap feature, so that when the ISCC's DMA controller is reading transmit data from RAM, it takes alternate Bytes from AD7-AD0 and AD15-AD8.
- D6 of the data is 1 so that when the ISCC's DMA controller is reading transmit data from RAM, it takes even-addressed Bytes from D7-D0 and odd-addressed bytes from D15-D8 (same function as the 80186).
- D2-D1 of the data are 11 to select double-pulsed mode for the ISCC's $\overline{\text{INTACK}}$ input. This is how the 80186 functions.
- D0 of the data is 0 to select Shift Left Address mode so that the ISCC subsequently takes register addressing from the AD5-AD1 lines rather than from AD4-AD0. This is because the 80186 is a 16-bit processor that locates even-addressed Bytes on AD7-AD0 and odd-addressed bytes on AD15-AD8, but the ISCC only accepts slave-mode writes on the AD7-AD0 pins.

- The ISCC's internal logic detecting activity on its $\overline{AS}$ pin, which is inverted from the 80186 ALE signal, automatically conditions it for a multiplexed Address/Data bus. Given that the BCR is written as previously described, the ISCC's slave mode map is as listed in Table 12.

Table 12. ISCC's Slave Mode Map

(PBA)+128, 130....(PBA)+190	DMA Controller Registers
(PBA)+192, 194....(PBA)+222	ISCC Serial Channel B Registers 0-15
(PBA)+224, 226....(PBA)+254	ISCC Serial Channel A Register 0-15

(M)USC

Since the 80186 processor provides multiplexed addresses and data, the (M)USC is configured to use the addresses on the AD lines. The software does not need to write register addresses into the indirect address field of the (M)USC CCAR.

The (M)USC's Transmitter and Receiver can be handled on a polled or interrupt-driven basis. In addition, and two of the Receivers and Transmitters in the (M)USC and Channel B of the (E)SCC can be handled on a DMA basis, using the 80186's integrated controllers.

Jumper block J22 connects the (M)USC's $\overline{RxREQ}$ and $\overline{TxREQ}$ outputs to the DMA EPLD that makes the DMA Requests to the 80186.As shipped from the factory, jumpers are installed between J22-J3 and J22-J4.In this configuration, the (M)USC's $\overline{RxREQ}$ drives the 80186 DREQ0 and (M)USC $\overline{TxREQ}$ drives the 80186 DREQ1. To reverse this assignment, jumper J22-J1 to J22-J3 andJ22-J2 to J22-J4. To disconnect the (M)USC from one or both of the 80186's DMA channels, remove one or both jumpers (put them in a safe place in the event they need to be reinstalled).

Jumper block J29 provides the same connection-variability for the $\overline{RxREQ}$ and $\overline{TxREQ}$ outputs of Channel B of a USC.

Since the 80186's channels are not capable of fly-by operation, the (M)USC's $\overline{RxACK}$ and $\overline{TxACK}$ pins have no dedicated function. They can be used for Request to Send and Data Terminal Ready. The two signals are lightly pulled up since they are not driven after Reset.

The (M)USC can be programmed using 16-bit data on the AD15-AD0 lines or 8-bit data on AD15-AD8 and AD7-AD0. It makes the distinction between 8-bit and 16-bit operations as part of its address map rather than through a control input. The PS pin of an MUSC, or the A/$\overline{B}$ pin of a USC, is connected to a latched version of the 80186 A7. The D/$\overline{C}$ pin of the (M)USC is grounded. The overall address range of the (M)USC is 256 Bytes, between (PBA)+256 and (PBA)+511.

The first write to this address range, after a Reset, implicitly writes the (M)USC Bus Configuration Register (BCR). To match the rest of the board's hardware, the first write must be 16-bit write, storing the hex value 0007 at any address in the second half of the (M)USC's range [any address in (PBA)+384 through 510, that is, in the A channel of a USC]. Details of this transaction are as follows:

- The High on the PS or A/$\overline{B}$ input, which is connected to A7, selects the WAIT protocol on the $\overline{WAIT}$/$\overline{RDY}$ pin, corresponding to how the 80186 works.
- The MSB of the data (D15) is 0 because a separate non-multiplexed address is not wired to pins AD13:8 of the (M)USC.
- Bits 14-3 are required to be all zeros by the (M)USC internal logic.
- D2 of the data is 1 to tell the (M)USC that the data bus is 16 bits wide.

- D1 of the data is 1 to select double-pulsed mode for the (M)USC's $\overline{\text{INTACK}}$ input. This is how the 80186 CPU functions.
- D0 of the data is 1 to select Shift Right Address mode so that the (M)USC subsequently takes register addressing from the AD6-AD0 lines rather than from AD7-AD1.
- The fact that the (M)USC's internal logic sees activity on its $\overline{\text{AS}}$ pin, which is inverted from the 80186' ALE signal, automatically conditions it for a multiplexed Addresses/Data bus.

Given that the BCR is written as described above, the (M)USC address map is as listed in Table 13.

Table 13. (M)USC Address Map

Starting Address	Ending Address	Registers Accessed
(PBA)+256	(PBA)+319	16-bit access to (M)USC registers or USC Channel B Registers
(PBA)+320	(PBA)+383	8-bit access to (M)USC registers or USC Channel B Registers
(PBA)+384	(PBA)+447	16-bit access to (M)USC registers or USC Channel A Registers
(PBA)+448	(PBA)+511	8-bit access to (M)USC registers or USC Channel A Registers

Note: To maximize compatibility, program the (M)USC using the second half of this range (PBA)+384 through (PBA)+511.

While the ESCC and ISCC drive their Baud Rate Generators from their PCLK inputs, the (M)USC has no such input. The 80186 clock output SYSCLK is brought to pins 7 of J9, J10, and J12, at which point it can be jumpered to pin 9 or 8 so that it is routed to the $\overline{\text{TxC}}$ or $\overline{\text{RxC}}$ pin of the device.

IUSC

Since the 80186 processor provides multiplexed addresses and data on the AD lines, the IUSC is configured to use these addresses. Software does not need to write register addresses into the indirect address fields of the IUSC's CCAR and DCAR.

The IUSC's two DMA channels allow its Receiver and Transmitter to be handled on a polled, interrupt-driven, or DMA basis, in any combination.

The IUSC can be programmed using 16-bit data on the AD15-AD0 lines or 8-bit data on AD15-AD8 and AD7-AD0. The distinction between 8-bit and 16-bit operations is made as part of the address map rather than via a control input.

The D/$\overline{\text{C}}$ pin of the IUSC is driven from A7 during slave cycles and the S/$\overline{\text{D}}$ pin is driven from A8.The overall address range of the IUSC is 384 bytes from (PBA)+512 through (PBA)+895.

The first write to this address range, after a Reset, implicitly writes the IUSC's Bus Configuration Register (BCR). To match with the rest of the board's hardware, this first write is a 16-bit write, storing the recommended hex value `00F7` at any word address in the range (PBA)+768 through (PBA)+830.

Details of this transaction are as follows:

- The High on the IUSC's S/$\overline{\text{D}}$ input, which is connected to A8, selects the WAIT protocol on the $\overline{\text{WAIT}}$/$\overline{\text{RDY}}$ pin, which is how the 80186 works.
- It may not be required for this initial write, but it is good programming form to have A6 set to zero, since this is a word write. This and the first bullet determine the recommended address range.

- The MSB of the data (D15) is 0 because a separate non-multiplexed address is not wired to Pins AD13:8 of the IUSC.
- Bits 14-8 are more or less required to be all 0's by the IUSC's internal logic.
- D7-D6 are 11 to allow the DMA controllers to do either 16-bit transfers, or alternating byte transfers on AD7-AD0 for even-addressed bytes and on AD15-AD8 for odd-addressed bytes. This is compatible with 80186 byte ordering.
- D5-D4 of the data are 11 to select double-pulsed mode for the IUSC's $\overline{\text{INTACK}}$ input.
- D3 of the data is 0 to select open-drain mode on the IUSC's $\overline{\text{BUSREQ}}$ pin. The board's control logic also drives this signal Low when the ISCC asserts its Bus Request output.
- D2 of the data is 1 to tell the IUSC that the data bit is 16 bits wide.
- D1 of the data is 1 to select open-drain mode on the IUSC's $\overline{\text{INT}}$ pin which is OR-tied with the interrupt request from the (E)SCC.
- D0 of the data is 1 to select Shift Right Address Mode, so that the IUSC subsequently takes register addressing from the AD6-AD0 lines rather than from AD7-AD1.
- The fact that the IUSC's internal logic sees activity on its $\overline{\text{AS}}$ pin, which is inverted from the 80186' ALE signal, automatically conditions it for a multiplexed Address/Data bus.

Given that the BCR is written as described above, the IUSC slave mode address map is as listed in Table 14.

Table 14. IUSC Slave Mode Address Map

Starting Address	Ending Address	Registers Accessed
(PBA)+512	(PBA)+575	16-bit access to IUSC Transmit DMA Registers
(PBA)+576	(PBA)+639	8-bit access to IUSC Transmit DMA Registers
(PBA)+640	(PBA)+703	16-bit access to IUSC Receive DMA Registers
(PBA)+704	(PBA)+767	8-bit access to IUSC Receive DMA Registers
(PBA)+768	(PBA)+831	16-bit access to IUSC Serial Controller Registers
(PBA)+832	(PBA)+895	16-bit access to IUSC Serial Controller Registers

While the ESCC and ISCC can drive their Baud Rate Generators from their PCLK inputs, the IUSC cannot perform the same from its CLK input. The 80186 clock output SYSCLK is brought to Pins 7 of J9, J10, and J12 at which point it can be jumpered to pin 9 or 8 so that it is routed to the $\overline{\text{TxC}}$ or $\overline{\text{RxC}}$ pin of the device.

Since the IUSC contains its own DMA channels, its $\overline{\text{RxREQ}}$ and $\overline{\text{TxREQ}}$ have no dedicated function. They can be used for Request to Send and Data Terminal Ready. The two signals are lightly pulled up to allow for the fact that they are not driven after Reset.

Serial Interfacing

The serial I/O pins of the four serial controllers are connected to the six connector blocks labelled J5 through J10. In addition, the port pins of the IUSC are connected to the J11 connector block and the port pins of an MUSC or the B channel of a USC are connected to J12. The connector blocks can be interconnected for communication between on-board serial controllers, or they can be connected to the user's custom communications hardware on another board. As a third option, they can be connected to three on-board serial interfaces via the connector blocks labelled J13 through J15.

Two of the on-board serial interfaces use EIA-RS-232 signal levels and pin arrangement. 25-pin D connectors J1A or J2A are configured as DTE, while J1B and J2B are configured as DCE. These serial interfaces are used by connecting one of the J5-J10 to J13 or J14, respectively. J1B is typically used for connection to the user's PC or terminal.

The third on-board serial interface uses EIA-422 signal levels on connector J3A, J3B, or J4 and is used by connecting one of J5-J10 to J15. The 25-pin D connector J3A uses the DTE pin arrangement put forth in the EIA-530 standard. J3B is a DCE version of EIA-530, while the 8-pin circular DIN connector, J4, is compatible with the Apple Macintosh Plus and later Macintoshes, and thus with AppleTalk/LocalTalk equipment.

Table 15 and Table 16 summarize the serial interface connectors.

Table 15. Controller Port Connectors

Serial Controller Channel w/on/off Board Hardware	Connect to this (these) Connector Blocks
(E)SCC Channel A	J5
(E)SCC Channel B	J6
ISCC Channel A	J7
ISCC Channel B	J8
IUSC	J9 (J11 for Port pins)
(M)USC	J10 (J12 for MUSC Port pins or USC Channel B

Table 16. On-Board Line Driver/Receiver Connectors

Using a Serial Chip Controller w/following on-chip Serial Interface	Connect Connectors in Table 6 to:
J1A or J1B EIA-RS-232 Console	J13
J2A or J2B EIA-RS-232	J14
RS-422 Differential: J3A or J3B EIA-530 or J4 Circular-8 (DIN)	J15

The pin-out of the J5-J10 connectors is consistent, but may not be identical because of differences among the various serial controllers. These differences are listed in Table 17.

Table 17. Pin Assignments of Standard Controller Connectors

Pins	J5 (E)SCC A pin	J6 (E)SCC B pin	J7, J8 ISCC pin	J9 IUSC pin	J10 MUSC or USC A pin	J12 USC B pin
1	TxD	TxD	TxD	TxD	TxD	TxD
2	RxD	RxD	RxD	RxD	RxD	RxD
3	$\overline{\text{RTS}}$	$\overline{\text{RTS}}$	$\overline{\text{RTS}}$	(N/C)	$\overline{\text{RxACK}}$	$\overline{\text{RxACK}}$
4	$\overline{\text{CTS}}$	$\overline{\text{CTS}}$	$\overline{\text{CTS}}$	$\overline{\text{CTS}}$	$\overline{\text{CTS}}$	$\overline{\text{CTS}}$
5	$\overline{\text{DTR}}$	$\overline{\text{DTR}}$ or N/C (Note)	$\overline{\text{DTR}}$	(N/C)	$\overline{\text{TxACK}}$	$\overline{\text{TxACK}}$
6	$\overline{\text{DCD}}$	$\overline{\text{DCD}}$	$\overline{\text{DCD}}$	$\overline{\text{DCD}}$	$\overline{\text{DCD}}$	$\overline{\text{DCD}}$
7	$\overline{\text{SYNC}}$	$\overline{\text{SYNC}}$	$\overline{\text{SYNC}}$	(SYSCLK)	(SYSCLK)	(SYSCLK)
8	$\overline{\text{RTxC}}$	$\overline{\text{RTxC}}$	$\overline{\text{RTxC}}$	$\overline{\text{RxC}}$	$\overline{\text{RxC}}$	$\overline{\text{RxC}}$
9	$\overline{\text{TRxC}}$	$\overline{\text{TRxC}}$	$\overline{\text{TRxC}}$	$\overline{\text{TxC}}$	$\overline{\text{TxC}}$	$\overline{\text{TxC}}$
10	GND	GND	GND	GND	GND	GND
11	NA	NA	NA	$\overline{\text{TxREQ}}$	$\overline{\text{TxREQ}}$	$\overline{\text{TxREQ}}$
12	NA	NA	NA	$\overline{\text{RxREQ}}$	$\overline{\text{RxREQ}}$	$\overline{\text{RxREQ}}$

Note: Controlled by the J24 jumper block. Must be N/C id (E)SCC Channel B transmitter is to be handled by an 80186 DMA channel.

The ground pins are included as signal references with off-board hardware. When interconnecting between two connectors among J5-J10, DO NOT jumper corresponding pins straight across, as this connects outputs to outputs and inputs to inputs. Connect at least pin to the other pin 2, and enough opposing inputs and outputs as needed to make the communications protocol meaningful.

The pinout of the 12-pin J13-J15 connectors is similar to that of J5-J10, but more extensive. To allow for the DCE connectors that were added in revision 'B' of the board, J13 and J14 are 16-pin headers and J15 is a 14-pin header. See Table 18.

Table 18. Pin Assignments of Line Driver/Receiver Connectors

Pins	J13-J14 DTE Signal	J13-J14 DCE Signal	J15 DTE Signal	J15 DCE Signal	Direction/Where Used
1	TxD	RxD	TxD	RxD	Output to J1-J4
2	RxD	TxD	RxD	TxD	Input from J1-J4
3	$\overline{\text{RTS}}$	$\overline{\text{CTS}}$	$\overline{\text{RTS}}$	$\overline{\text{CTS}}$	Output to J1-3
4	$\overline{\text{CTS}}$	$\overline{\text{RTS}}$	$\overline{\text{CTS}}$	$\overline{\text{RTS}}$	Input from J1-J4
5	$\overline{\text{DTR}}$	$\overline{\text{DSR}}$	$\overline{\text{DTR}}$	$\overline{\text{DSR}}$	Output to J1-J4
6	$\overline{\text{DSR}}$	$\overline{\text{DTR}}$	$\overline{\text{DSR}}$	$\overline{\text{DTR}}$	Input from J1-J4

Table 18. Pin Assignments of Line Driver/Receiver Connectors (Continued)

Pins	J13-J14 DTE Signal	J13-J14 DCE Signal	J15 DTE Signal	J15 DCE Signal	Direction/Where Used
7		$\overline{DCD}$		$\overline{DCD}$	Output to J1B, J2B, J3B
8	$\overline{DCD}$		$\overline{DCD}$		Input from J1A, J2A, J3A, J4
10	GND	GND	GND	GND	
11		$\overline{RxC}$		$\overline{RxC}$	Output to J1B, J2B, J3B
12	$\overline{RxC}$		$\overline{RxC}$		Input from J1A, J2A, J3A
13	$\overline{TxCO}$	$\overline{TxCI}$	$\overline{TxCO}$	$\overline{TxCI}$	Output to J1-3
14	$\overline{TxCI}$	$\overline{TxCO}$	$\overline{TxCI}$	$\overline{TxCO}$	Input from J1-3 (NOTE)
15		$\overline{RI}$			Output to J1B, J2B
16	$\overline{RI}$				Input from J1A, J2A

Note: Various conventions are used to combine synchronous clock inputs and modem control inputson Apple Macintosh connectors similar to J4, as described in the later section.

Comparison of the two previous tables leads to following conclusions:

- Pins 1-5 can always be jumpered straight across from a J5-J10 connector block to a J13-J15 connector block.
- In a synchronous environment, the Transmit clock can be either driven or received and the Receive clock can be received from the DTE connector or sent on the DCE connector.

The 10-pin J11 and J12 jumper blocks provide connections to the Port pins of the IUSC and (M)USC, respectively. As with J5-J10, these connections may be to the customer's off-board custom circuits and/or to certain pins in the J13-J15 blocks. The following pin assignment is determined so that if a 2-channel USC is plugged into the (M)USC socket, J12 has the same pin-out for the USC's B channel as do J5-J10 for other channels (see Table 19).

Table 19. Pin Assignments of Controller Port Connectors

Pin No	J11:IUSC Signal	J12: (M)USC Signal
1	Port1(Clock 1 In)	Port1
2	Port4 (Xmit TSA Gate Out)	Port4 (Xmit TSA Gate Out)
3	N/C	N/C
4	Port0 (Clock 0 In)	Port0
5	N/C	N/C
6	Port3 (Rcv TSA Gate Out)	Port3 (Rcv TSA Gate Out)
7	N/C	(SYSCLK)

Table 19. Pin Assignments of Controller Port Connectors (Continued)

Pin No	J11:IUSC Signal	J12: (M)USC Signal
8	Port5 (Rcv Sync Out)	Port5 (Rcv Sync Out)
9	Port2	Port2
10	GND	GND
11	PORT6 (Rcv Sync IN)	PORT6 (Rcv Sync IN)
12	Port7 (Xmit Complete Out)	Port7 (Xmit Complete Out)

Finally, an unpopulated 4-pin oscillator socket is included on the board with its output connected to a single jumper/wire-wrap pin. This socket can be populated with a user-supplied oscillator and connected to various clock pins among J5-J15.

Sensing Which Serial Controller Channel is Connected to the Console

To use the software provided with this evaluation board, one of the serial controller channels must be connected to a PC or a dumb terminal via the J1 and J13 connectors. Some versions of this software may restrict the choice to (E)SCC Channel A or (M)USC, depending on your application needs. There is nothing in the hardware that limits the choice of which serial channel is used for the console. However, on the J1-J4 (J13-J15) side, there are two things that are special about the J1-J13 section as compared to the others. One is the provision for a Non-Maskable Interrupt in response to a received Start bit, as described in (E)SCC addressing on page 6 .

Software can use the other special feature of the J1/J13 section, after a Reset, to sense which serial channel is connected to the Console port. A Reset signal (from power-on or Reset button but not from the Reset-The-ISCC, address decode (and others) described earlier) places the "NMI" EPLD in a special mode where the first Start bit on the Consoles Transmit Data lead causes an NMI. This feature can be used in a start-up procedure like the following example, to determine which serial controller channel is used for the Console.

For each serial controller channel that the software can use for the Console:

1. Initialize the channel.
2. Send NULL character to the channel.
3. Wait for a short time to see if an NMI occurs. If an NMI occurs., the current channel is the Console otherwise go on to the next serial channel and try again.

If none of the allowed serial channels produce an NMI, you may not have properly jumpered any J5-J10 connector block to the J13 block.

Basic software should use the serial controller channel for the Console in a very basic, polled way. Because of this and similarities between the (E)SCC and the ISCC, and between the (M)USC and the IUSC, note that software allows the Console to be connected to either the (E)SCC Channel A or to the (M)USC. It includes most of the code necessary to use any of the six serial controller channels for the Console.

Notes on J4/Macintosh/AppleTalk/LocalTalk

The J4 connector is similar to the connectors offered on various Macintosh systems. The ESCC and ISCC are well adapted for use with this port, and development of USC family capability for AppleTalk/LocalTalk.

The J3 and J4 connectors cannot be used simultaneously. The J16 jumper block controls whether the RS-422 driver for Transmit Data is turned ON and OFF under control of the associated Request to Send signal, as on the Mac, or is ON full time, which is more suitable for the use of J3. To put the TxD driver under control of RTS, jumper J16-J1 to J16-J2 and leave J16-J3 open. For full time drive on TxD (and also the J3 RTS pins), jumperJ16-J2 to J16-J3 and leave J16-J1 open.

The J17 jumper block controls whether the reception of Data Carrier Detect and Clear to Send is differential (on J3) or unbalance, as on J4. To use differential signalling from J3, remove all jumpers from J17.

On the initial Macintosh and subsequent ones as well, Apple did the unbalanced signaling backward from the standard RS-423 and RS-232 polarity for the CTS lead (also called HSK and HSKI). If you are developing code for Macintosh hardware, you can preserve Mac compatibility by jumpering J17-J3 to J17-J5 and J17-J4 to J17-J6. This grounds the CTS- lead and connects the CTS+ lead to J4-J2. It also (assuming a standard source at the other end) inverts CTS to the opposite sense from that expected by the serial controller for functions such as auto-enabling. To make the CTS input of the serial controller have its normal (low-true) sense, jumper J17-J3 to J17-J4, and J17-J5 to J17-J6. This grounds the CTS+ lead and connects the CTS- lead to J4-J2.

The DTR (HSKO) is provided in the Apple systems from Mac Plus onward and has standard RS-423 (and RS-232) polarity.

The DCD input on J4-J7 is provided in Apple systems from the Mac II and SE onward, and also has standard polarity on Apple hardware. Jumper J17-J1 to J17-J2 to ground "+" input of the receiver. The "-" lead is connected to J4-J7.

With jumpers installed to make DCD and CTS unbalanced, J4 can also be used for an additional RS-232 serial link. Connect a "Mac to Hayes modem" cable to J4, and optionally a null modem interconnect module to the other end. The cable internally grounds the RxD+ and TxD+ leads so that RxD- and TxD- function like RS-232 signals.

Macintosh systems also include provisions for synchronous clock inputs. It is unknown if these features are used by any applications or attached hardware. On all known Macs, the SCC's TRxC pin is driven from the same signal as CTS. To be compatible with this feature, connect J15-J4 to pins 4 and 9 of the selected connector among J5-J10.

On the Mac SE, Mac II, and later models, a multiplexing scheme is provided on SCC channel A's RTxC pin to drive from either the same signal as DCD, or from an on-board 3.672 MHz clock. Channel B always had the 3.672 MHz clock. The former capability can be provided by connecting J15-J6 to pins 6 and 8 of the selected connector among J5-J10. The latter capability can be only approximated using the 80186 clock with different baud rate divisors, or by using another oscillator. The board includes an unpopulated 4-pin oscillator socket that can be useful in this regard.

Jumper Summary

Table 20 includes only those connector blocks intended to be populated by 2-pin option jumpers. J1-J15 and J26 are actual connectors meant for use with cables, jumper wires, or wire wrapped connectors.

Table 20. Two-Pin Option Jumpers

Jumpers	Installed	Open
J9-J7 thru -9	7 to 8: 80186 SYSCLK is IUSC $\overline{RxC}$ 7 to 9: 80186 SYSCLK is IUSC $\overline{TxC}$	8: Something else on $\overline{RxC}$, or N/C 9: Something else on $\overline{TxC}$, or N/C
J10-J7 thru -9	7 to 8: 80186 SYSCLK is MUSC (USC A) $\overline{RxC}$ 7 to 9: 80186 SYSCLK is MUSC (USC A) $\overline{TxC}$	8: Something else on $\overline{RxC}$, or N/C 9: Something else on $\overline{TxC}$, or N/C
J12-J7 thru -9	7 to 8: 80186 SYSCLK is USC B $\overline{RxC}$ 7 to 9: 80186 SYSCLK is USC B $\overline{TxC}$	8: Something else on $\overline{RxC}$, or N/C 9: Something else on $\overline{TxC}$, or N/C
J16-J1 thru -3	1 to 2: J3, J4 TxD driven when RTS 2 to 3: J3, J4 TxD driven full-time	Must install one or the other
J17-J1 thru -2 J17-J3 thru -6	Unbalanced DCD- on J3 or J4 3 to 5 and 4 to 6: CTS+ on J4-J2 3 to 4 and 5 to 6: CTS- on J3 or J4	Differential DCD+,. DCD- on J3 Differential CTS+,. CTS- on J3
J18-J1 thru -3	1 to 2: 2764, 27128, 27256 EPROMs 2 to 3: 27512 EPROMs	Must install one or the other
J19-J1 thru -6	1 to 2 and 4 to 5: 128K x 8 SRAMs 2 to 3 and 5 to 6: 32K x 8 SRAMs	Must install one or the other
J20-J1 thru -3	1 to 2: U2 contains 80C30 or 80230 2 to 3: U2 contains 85C30 or 85230	Must install one or the other
J21-J1 thru -6	1 to 2 and 4 to 5: U2 contains 80C30 or 80230 2 to 3 and 5 to 6: U2 contains 85C30 or 85230	Must install one or the other
J22-J1 thru -4	1 to 2: MUSC (USC A) RxREQ on DMA 0 1 to 3: MUSC (USC A) RxREQ on DMA 1 2 to 4: MUSC (USC A) TxREQ on DMA 0 3 to 4: MUSC (USC A) TxREQ on DMA 1	1: MUSC (USC A) Rx no DMA 4: MUSC (USC A) Tx no DMA
J23-J1 thru -3	1 to 2: (E)SCC B RxRQ on DMA 0 2 to 3: (E)SCC B Wait function	(E)SCC B neither RxDMA nor Wait
J24-J1 thru -4	1 to 2: clipped SCC B TxREQ on DMA1 1 to 3: direct ESCC B TxREQ on DMA1 3 to 4: $\overline{DTR}$ output from ESCC B	(E)SCC B neither TxDMA nor $\overline{DTR}$
J25-J1 thru -5 and J25X	1 to 2 and 3 to 4: (E)SCC last on IACK chain MUSC second to last J25X to 2 and 3 to 4: (E)SCC last, USC second to last 2 to 3 and 4 to 5: (E)SCC first on IACK chain	Must be one of these three ways

Table 20. Two-Pin Option Jumpers (Continued)

Jumpers	Installed	Open
J28-J1 thru -6	1 to 2: 80186 SYSCLK is (E)SCC PCLK 3 to 4: 80186 SYSCLK is ISCC PCLK 5 to 6: 80186 SYSCLK is IUSC PCLK	Connect some other clock to 2, 4, or 6
J29-J1 thru -4	1 to 2: USC B RxREQ on DMA0 1 to 3: USC B RxREQ on DMA1 2 to 4: USC B TxREQ on DMA0 3 to 4: USC B TxREQ on DMA1	1: USC B Rx no DMA 4: USC B Tx no DMA

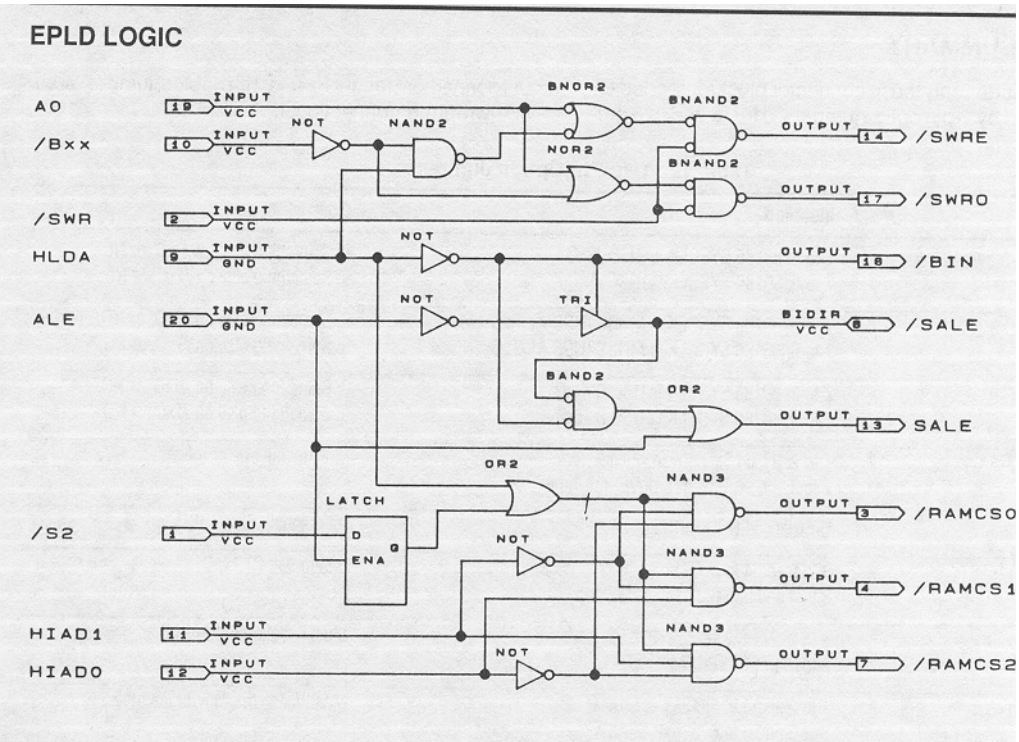

Figure 1. Control EPLD for 186 Board

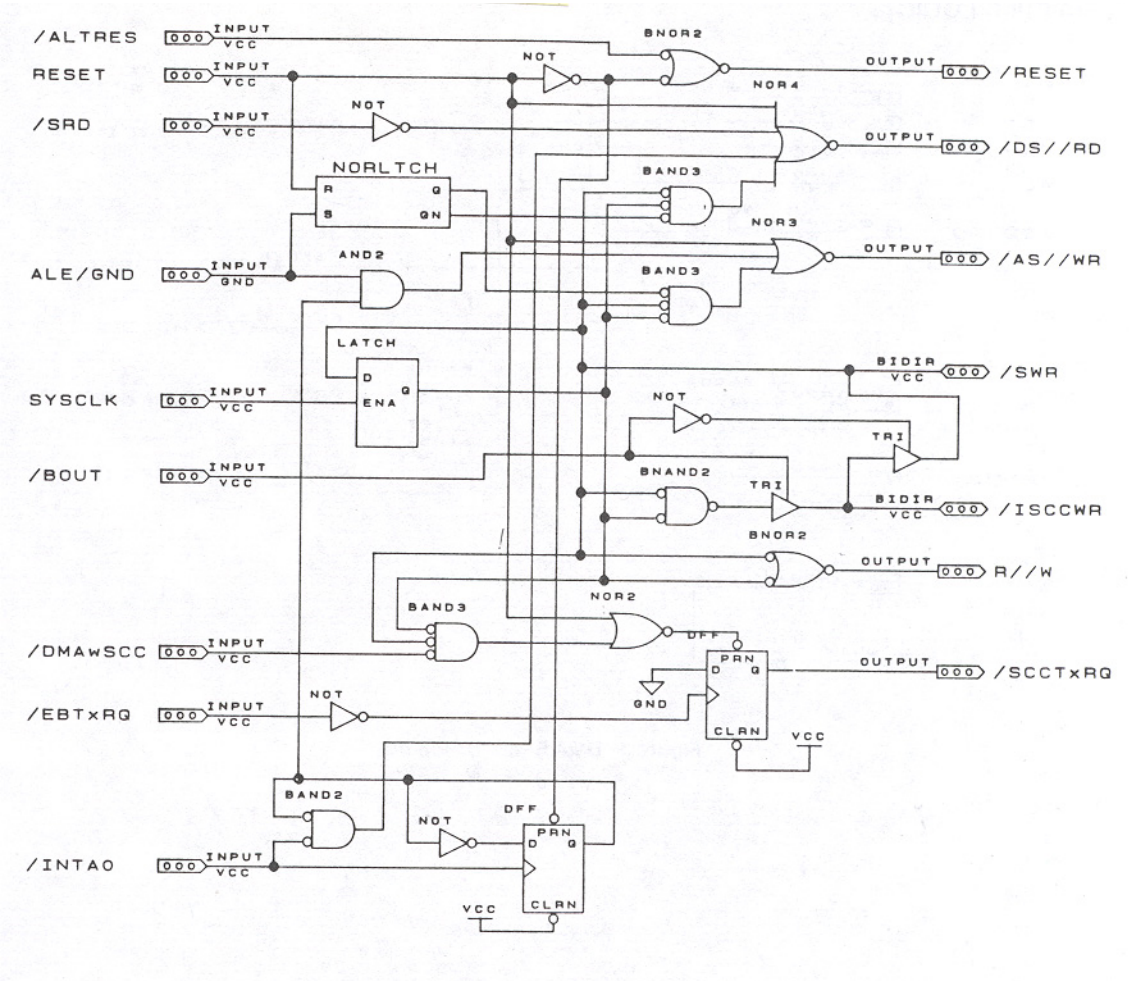

Figure 2. SCC EPLD for 186 Board

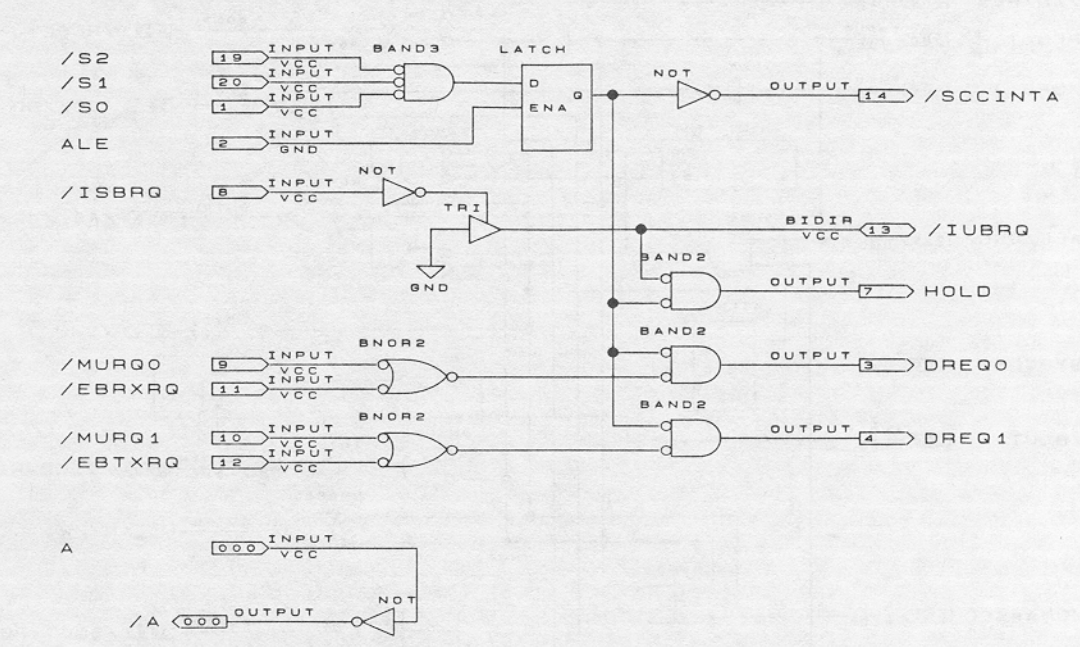

Figure 3. DMA EPLD for 186 Board

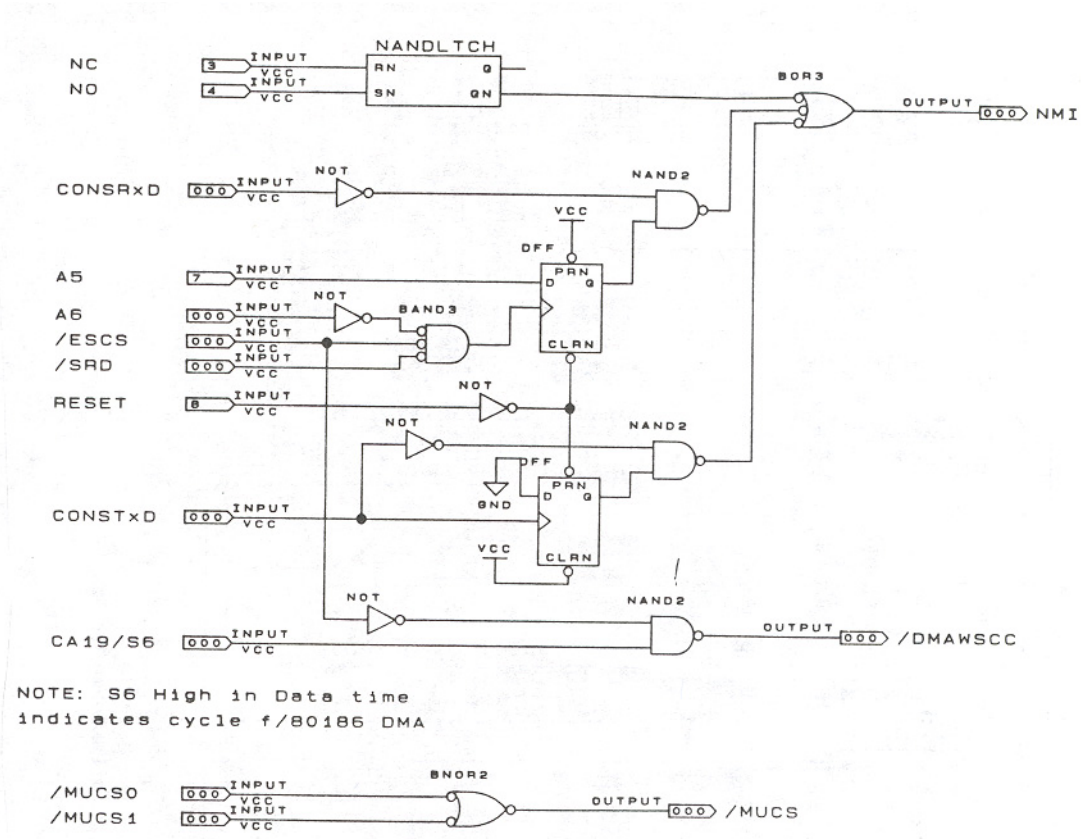

Figure 4. NMI Field for 186 Board

Warning: DO NOT USE IN LIFE SUPPORT

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer

 Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.