SECTION 2
STD BUS Practice

The industry-based STD Manufacturers Group has resolved the following recommended practices for the design of STD BUS cards. The STD Practice is supplemental to the STD specification and is to be applied at the discretion of the user. The current STD Practice relates to:

- Compatiblity Designation Practice
- Bus Timing Practice
- Interrupt Priority Practice
- Bus Priority Practice
- Memory Expansion Practice
- Backplane Practice
- Open-Collector/Drain Bus-Signal Practice
- CMOS Load Capacitance Practice
- CMOS User Edge Input Practice
- CMOS Pull-Up Resistor Practice

Compatibility Designation Practice

STD BUS cards that use peripheral chips usually depend on specific timing signals from the processor. This dependency prevents peripheral cards from being used interchangeably with cards from other families.

The STD Practice for designating compatibility is to label cards that are processor-timing-dependent, with reference to the CPU device: STD-Z80/CMOS Z80, STD-8085/80C85A, STD-6800, etc.

Bus Timing Practice

Card designers require bus timing definitions to insure compatibility. The recommended STD Practice for cards that source the bus control signals is for each card to specify the waveforms and timing information. Bus timing is further defined in the family timing specifications.

- STD-65/68XX
- STD-8085/80C85A
- STD-Z80/CMOS Z80
- STD-8088/80C88
- STD-NSC800

Interrupt Priority Practice

The STD BUS provides signal lines for interrupt requests. In systems with only a single interrupting device, these lines are sufficient to allow direct implementation. In systems with multiple interrupting devices, a priority scheme is necessary. This practice explains a serial priority scheme and defines a request signal for the user interface which allows implementation of a parallel priority scheme for interrupts.

Serial Priority for Interrupts. The STD BUS includes a priority-chain bus signal for serial priority schemes. Serial priority, using PCI and PCO signals, requires that each peripheral needing priority must have logic on the card to service the request, as shown in figure 2-1. This scheme is practical with peripheral devices designed to service a serial priority chain such as the Z80 family of devices.

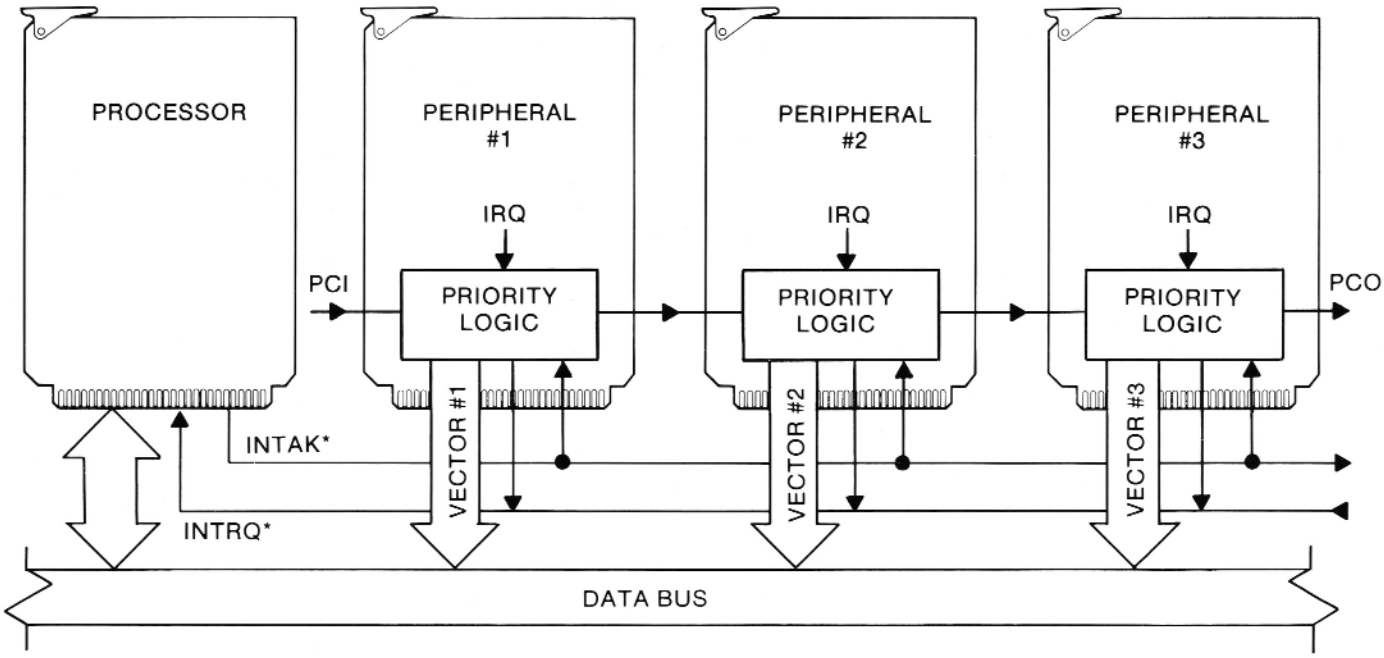

Figure 2-1. Serial Scheme for Interrupt Priority

Parallel Priority for Interrupts. A parallel priority scheme for interrupts can be implemented on the STD BUS, so that the priority logic rides on a separate card and not on each peripheral card. The parallel priority card is a modular function that can be tailored to individual processor requirements. This scheme allows peripheral cards to be processor-independent. It requires that the individual requests be made from the user edge of the card, as shown in figure 2-2. The parallel priority encoder could be included on the processor card.

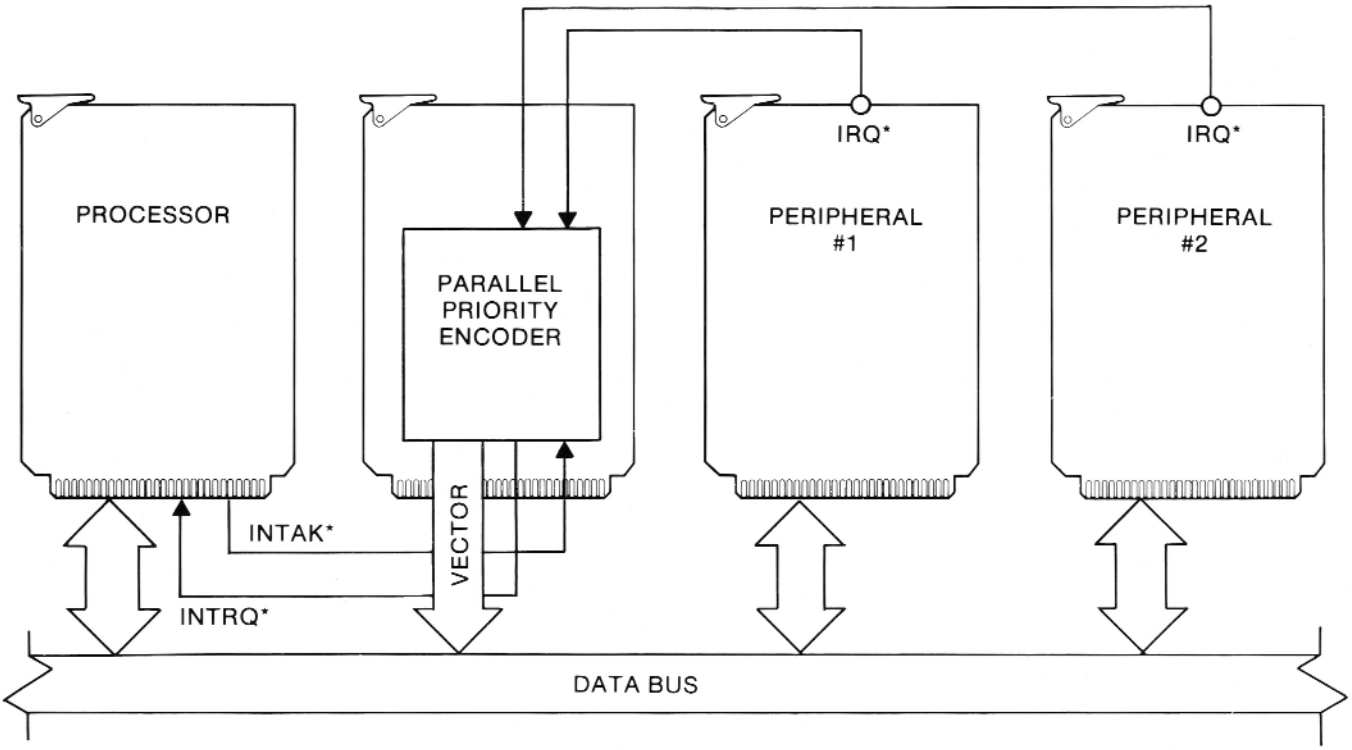

Figure 2-2. Parallel Scheme for Interrupt Priority

User Interface Signal for Parallel Priority. This practice defines a signal for the user interface which allows implementation of the parallel priority scheme as shown in figure 2-2.

IRQ* — Internal Interrupt Request. This signal indicates that the card is requesting an interrupt. IRQ* is passed separately by each card to a priority resolver card via the user interface. The priority resolver shall control INTRQ* and respond to INTAK*.

Interrupt — Parallel Priority Interface Practice. Cards designed to work with the parallel priority scheme require circuit connections at the user interface edge of the card. The STD Practice for compatibility is:

Mechanical:

- Two connections for each request channel: a request signal and a ground signal.
- Connector layout for priority encoder cards is as shown in figure 2-3. The ground pins are the top row and the request pins are the bottom row.
- Connector pins are 0.025-in. square posts or equivalent.
- Requesting cards provide the cable and mating connector. This practice does not define the method for fastening the cable to the requesting card. The cable end at the requesting card may be permanently wired or connected via any desired connector means.
- Mating cables can be twisted pairs or flat cable.
- Mating connectors are two pins per channel, in any size from single channel (2-pin) to multiple channel.

Electrical:

- Low-level active signal.
- LSTTL or high speed CMOS logic levels.
- Drive sink capability of 16 mA at 0.4V minimum for TTL or 6 mA at 0.37V minimum for CMOS.
- Open-collector/drain driver with 10K pull-up minimum.
- Load on encoder input: 4 LSTTL or CMOS loads in parallel with 4.7K (TTL) or 10K (CMOS) pull-up resistor.

Pull-up resistors on the encoder card inputs are recommended to disable the request if no connection is made. Open-collector/drain drivers on requesting cards are recommended, to allow wire-ORing of multiple requests on a single channel. This scheme is useful to low-level requests and requires the processor to poll to identify the requester.

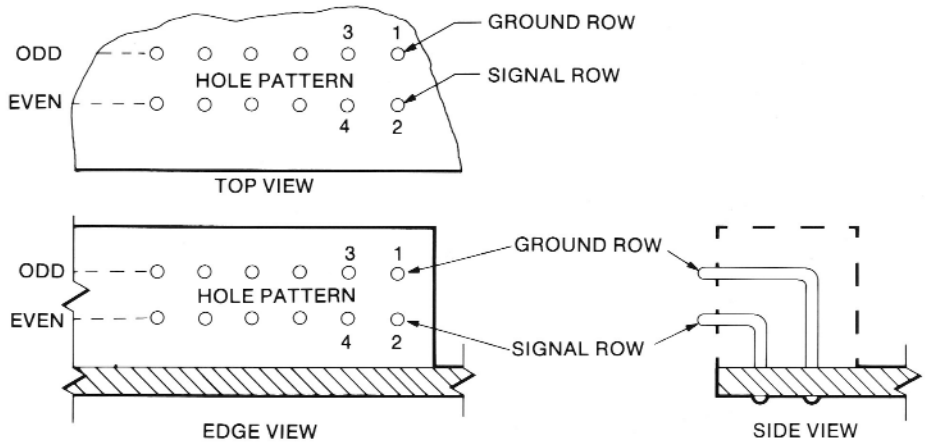

Figure 2-3. Recommended Connector Arrangement for Priority Encoder Cards

Bus-Priority Practice

The STD BUS provides signal lines for servicing bus requests. In systems with only a single alternate controller, these lines are sufficient to allow direct implementation. In systems with multiple bus controllers, a priority scheme is necessary. This practice defines three signals for the user interface which allow implementation of either a serial or parallel scheme for bus priority.

Serial Priority Bus Control. The STD BUS includes a priority-chain for serial priority schemes, however, this chain is generally used for interrupt priority. This practice defines a separate alternate chain for BUS-priority via the user interface as shown in figure 2-4. Serial priority requires that each peripheral needing priority must have logic on the card. This scheme is practical with devices designed to service a serial priority chain such as the Z80 family of devices.

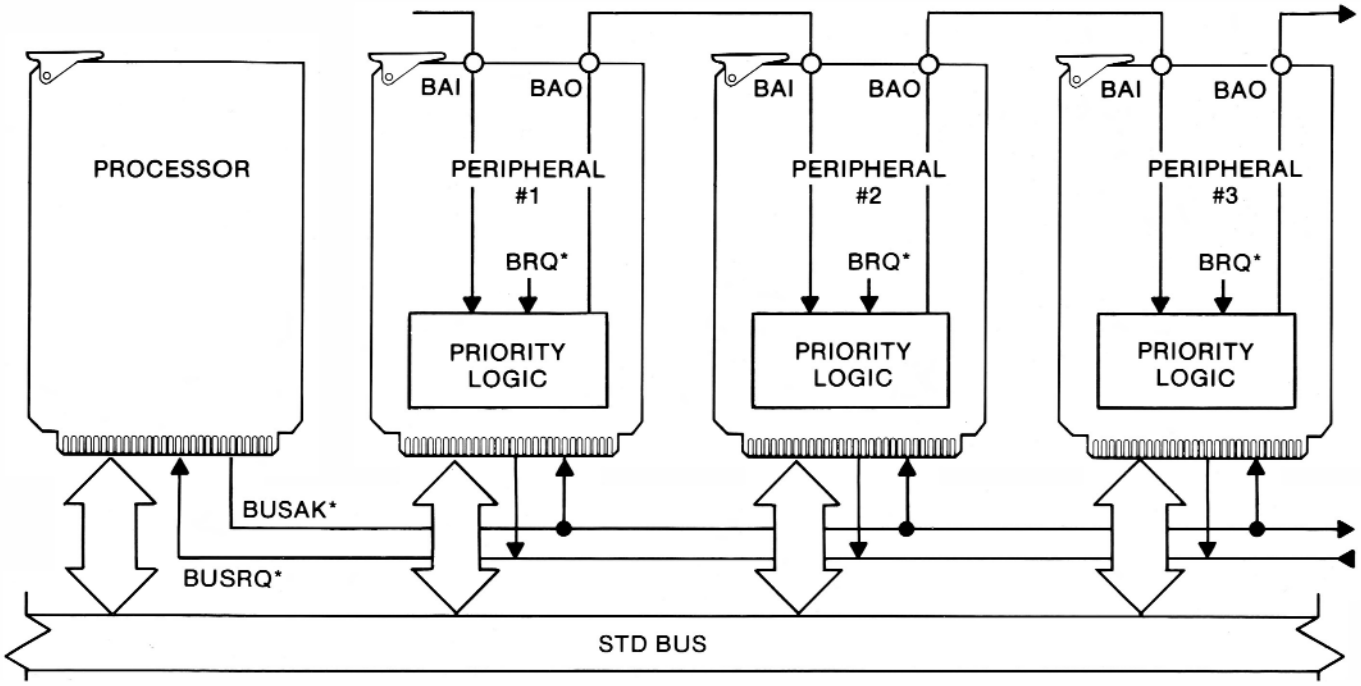

Figure 2-4. Serial Scheme for Bus Priority

User Interface Signals for Serial Priority Bus Control. This practice defines three signals for the user interface which allow implementation of the serial priority scheme shown in figure 2-4. The three signals are:

- BRQ* - Internal Bus Request
- BAI - Bus Acknowledge In
- BAO - Bus Acknowledge Out

BRQ* — Internal Bus Request (active low). This signal indicates that the card is requesting bus control. BRQ* drives BUSRQ* on the bus. BRQ* shall not originate if BUSAK* is low. When BRQ* and BUSAK* are low and BAI is high, bus control is given to this card.

BAI — Bus Acknowledge In (active high, 1K ohm (TTL) or 10K ohm (CMOS) pull-up). In serial priority if BAI is high and BRQ* and BUSAK* are low, bus control is given to this card.

BAO — Bus Acknowledge Out (active high). In serial priority BAO is high during a BUSAK* cycle if BRQ* is high. BAO is connected to BAI of the next lowest priority controller.

Serial Priority Signal Sequence. The signal sequence for serial priority is shown in figure 2-5.

- BRQ* shall not occur if BUSAK* is low.
- BUSRQ* occurs in response to BRQ*.
- BUSAK* occurs in response to BUSRQ* (see STD specification).

- BAI occurs in response to BUSAK*.
- BAO occurs in response to BAI and BUSAK*.

Serial Priority, Interface Practice. Cards designed to work with multiple bus requests using serial priority require circuit connections at the user interface edge of the controller cards. The STD Practice for compatibility is defined for each controller.

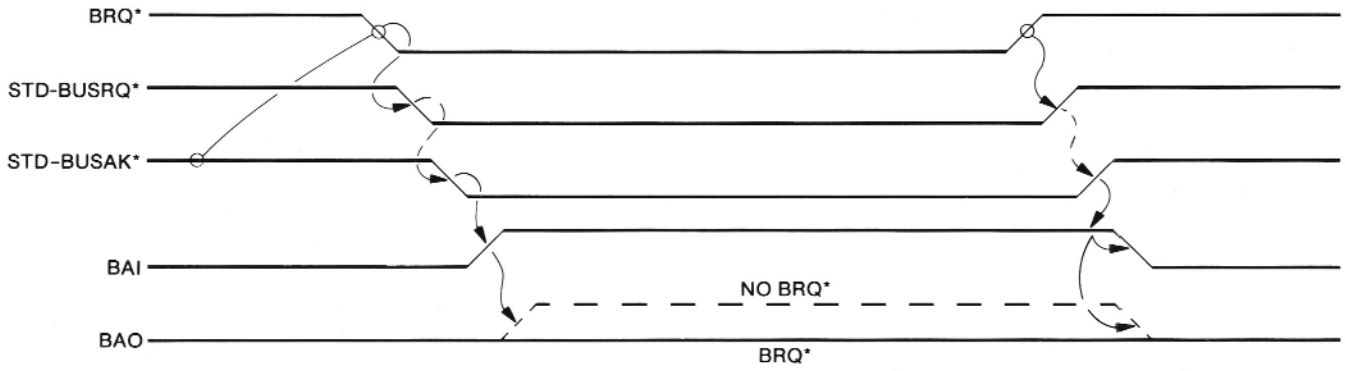

Figure 2-5. Serial Priority Signal Sequence

Mechanical:

- Connector layout for controller cards is as shown in figure 2-6. The ground pins are the top row and the signal pins are the bottom row.
- Connector with a minimum of six pins: three signal lines and three ground lines.
- Connector pins are 0.025-in. square posts or equivalent.
- Mating cables are twisted pairs.

Electrical:

- LSTTL or high speed CMOS logic levels.
- Connector pin assignments:

Pins 1, 3, 5	Ground
Pin 2	BAI
Pin 4	BAO
Pin 6	BRQ*

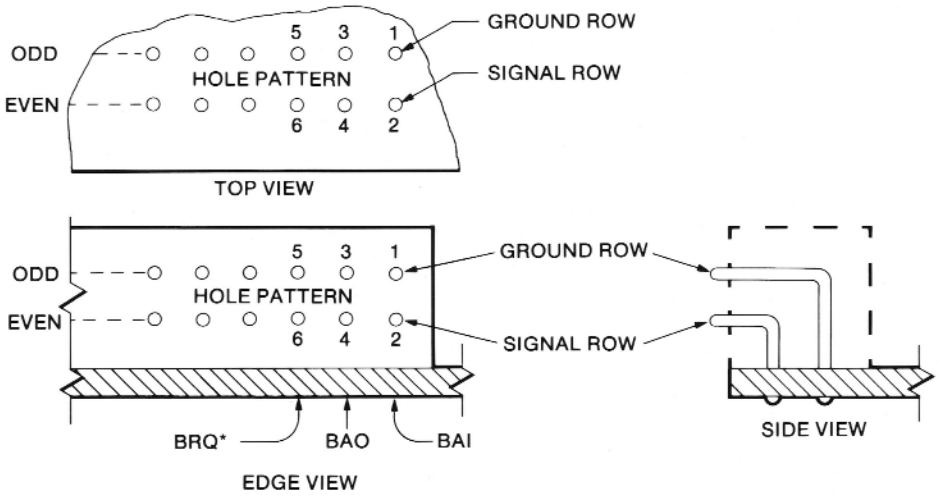

Figure 2-6. Recommended Connector Arrangement for Bus Controller Cards Using Serial Priority

Parallel Priority Bus Control. A parallel scheme for bus priority can be implemented on the STD BUS, so that the priority logic rides on a separate card and not on each peripheral card. The parallel priority card is a modular function that can be tailored to individual processor requirements. This scheme allows peripheral cards to be processor-independent. It requires that individual requests and acknowledges be made from the user edge of the card, as shown in figure 2-7. The parallel priority encoder could be included on the processor card.

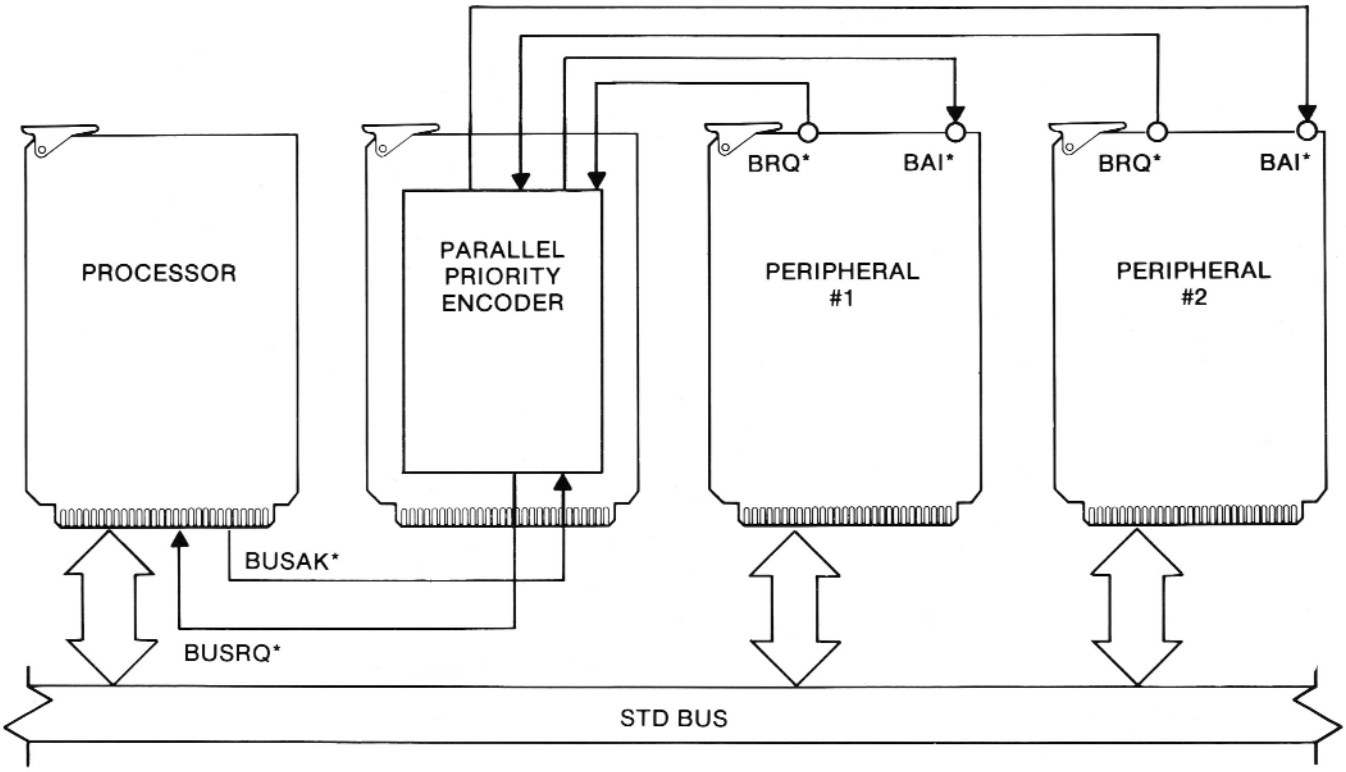

Figure 2-7. Parallel Scheme for Bus Priority

User Interface Signals for Parallel Priority Bus Control. This practice defines two signals for the user interface which allow implementation of the parallel priority scheme shown in figure 2-7. The two signals are:

- BRQ* — Internal Bus Request
- BAI* — Bus Acknowledge In

BRQ* — Internal Bus Request (active low). This signal indicates that the card is requesting bus control. BRQ* is passed separately by each controller card to a priority resolver card via the user interface. The priority resolver shall control BUSRQ* and respond to BUSAK* on the STD BUS.

BAI* — Bus Acknowledge In (active low, 1K ohm (TTL) or 10K ohm (CMOS) pull-up). BAI* is passed separately from the priority resolver card to each controller card via the user interface. Bus control is given to the controller card that senses BAI* low.

Parallel Priority, Interface Practice. Cards designed to work with multiple bus requests using parallel priority require circuit connectons at the user interface edge of the card. The STD practice for compatibility is:

Mechanical:

- Connector layout for priority resolved cards is as shown in figure 2-8. The ground pins are the top row and the signal pins are the bottom row.
- Four connections for each request channel: two signal lines and two ground lines.
- Connector pins are 0.025-in. square posts or equivalent.
- Mating female connectors are four pins per channel, in any size from single channel (4-pin) to multiple channel.
- Mating cables can be twisted pairs or flat cable.

Electrical:

- LSTTL or high speed CMOS logic levels.
- Connector pin assignments:

Pins	1,3	5,7	9,11	—,—	—,—	Two ground pins for each channel.
Pin	2	6	10	—	—	One BAI* signal for each channel.
Pin	4	8	12	—	—	One BRQ* signal for each channel.

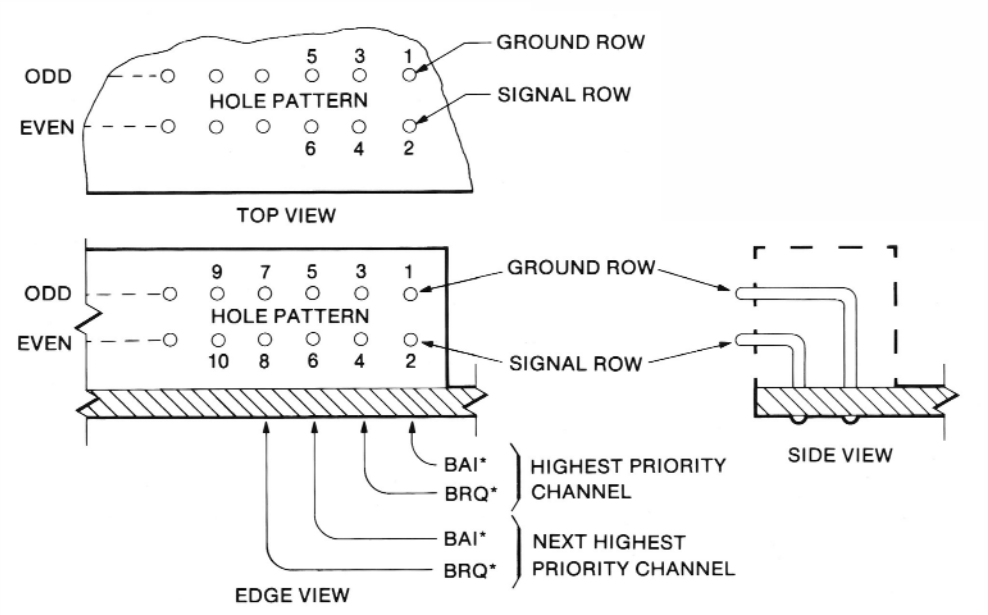

Figure 2-8. Recommended Connector Arrangement for Priority Resolver Cards for Parallel Priority Bus Control

Memory Expansion Practice

The STD BUS supports a primary memory space of 64K. Expansion of memory to 128K is supported by the MEMEX line on the bus. This practice discusses using the MEMEX line for expansion and suggests other methods.

MEMEX Memory Bank Selection. The MEMEX line is one of the signals for controlling fundamental memory operations. MEMEX must be included in the memory selection decoders for the STD compatible memory cards. When MEMEX is low the primary system memory is enabled. MEMEX may be used to enable an alternate 64K memory bank if the memory cards can be strapped for either high or low level enable by the MEMEX signal.

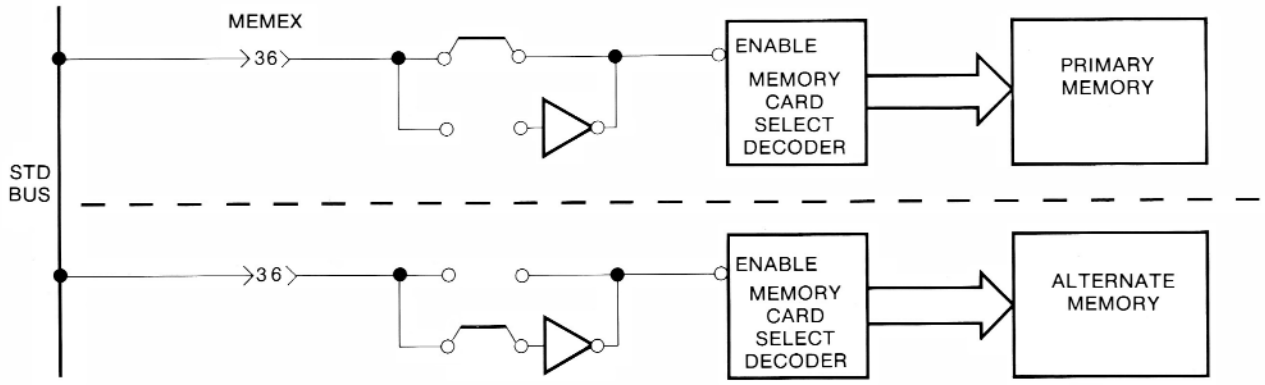

Figure 2-9. MEMEX Memory Bank Selection

Output Port Memory Bank Selection. Multiple banks of 64K memory may be selected by using an output port to enable individual banks. Various schemes are possible. The use of output port decoding to expand memory requires additional support logic on the memory cards.

On-Board Port Memory Selection. The on-board port scheme of memory expansion suggests using port address FF to decode and latch a strappable memory enable signal on each memory card. This scheme is represented in figure 2-10. Port address FF is decoded to enable latching the card select from the data bus. The latch outputs are jumper selectable to allow bank assignment for individual cards. System reset forces selection of memory bank 1 for compatibility with existing software.

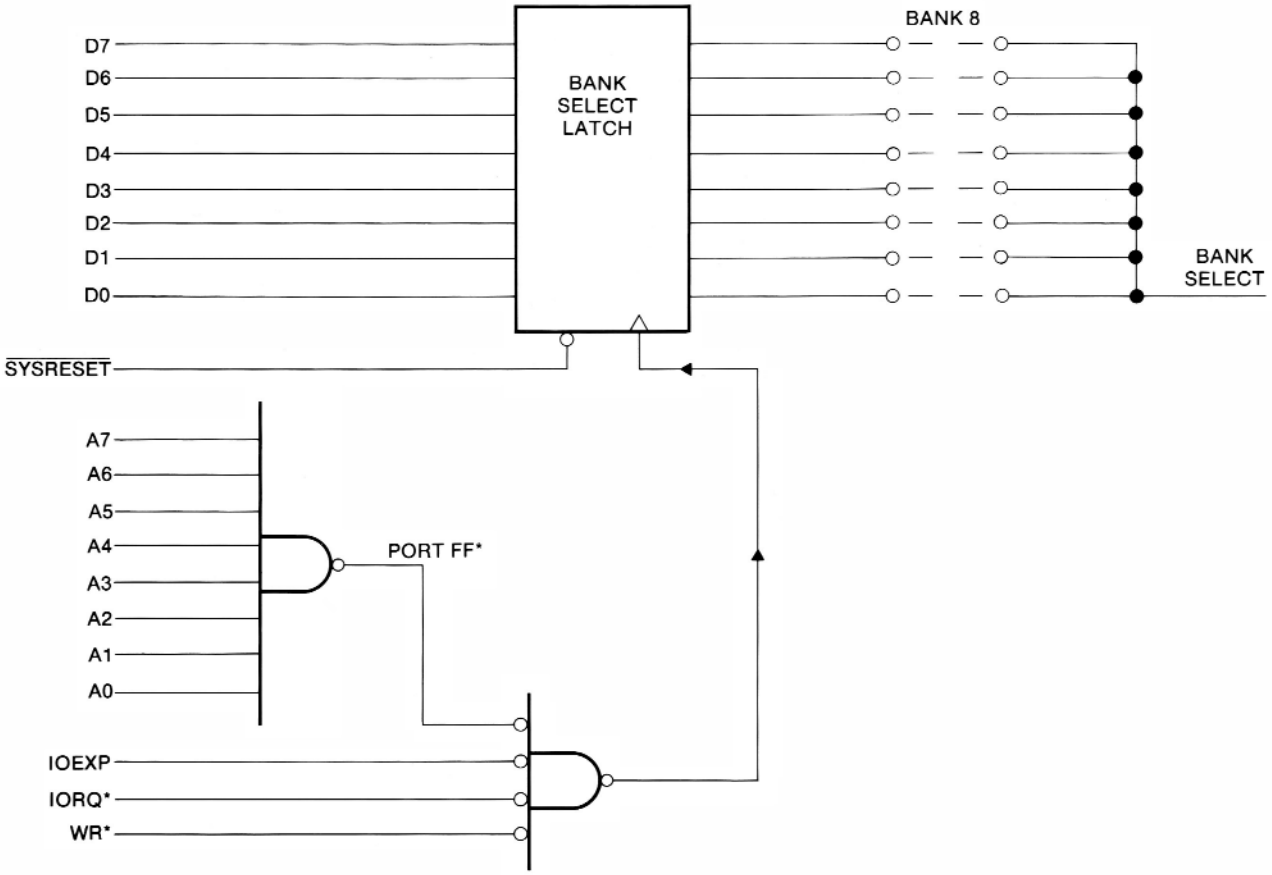

Figure 2-10. On-Board Port Memory Selection

Off-Board Port Memory Selection. The off-board port scheme of memory expansion suggests using a standard output port and wiring to special memory cards via the user interface. The memory cards require a user interface connection into the memory selection decoder on the card as shown in figure 2-11.

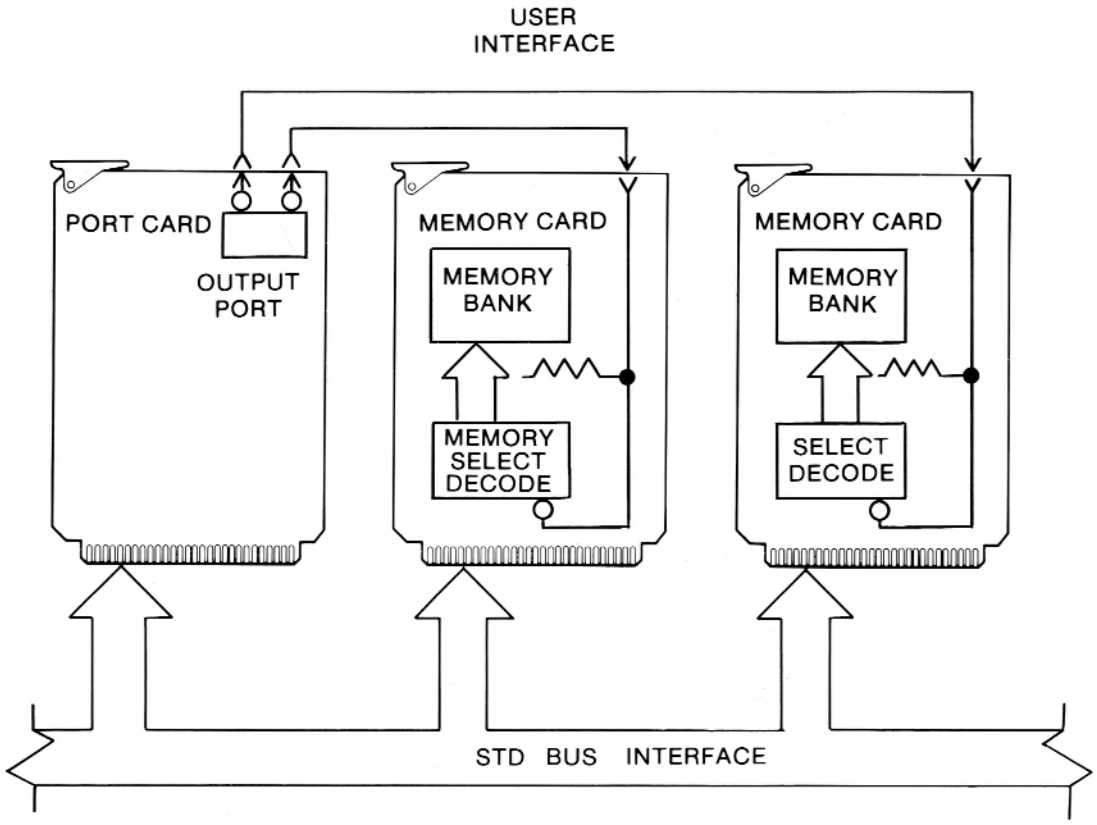

Figure 2-11. Off-Board Port Memory Selection

Multiplexed Memory Addressing. Processors on the STD BUS with more than 16 address lines can be accommodated by multiplexing up to 8 additional address lines on the data bus. This scheme assumes the processor timing can accommodate multiplexed addressing.

Multiplexed addressing requires special memory cards capable of demultiplexing and decoding the upper address lines.

The STD BUS pins are defined as shown in figure 2-12 for multiplexed data bus operation.

BUS PIN	SIGNAL NAME	SIGNAL DESCRIPTION
8	D7/A23	High order data/adr segment.
10	D6/A22	
12	D5/A21	
14	D4/A20	
7	D3/A19	
9	D2/A18	
11	D1/A17	
13	D0/A16	Low order data/adr segment.

Figure 2-12. Multiplexed Data Bus Pin Definitions

Backplane Practice

Extending the STD BUS Off the Backplane. The STD BUS backplane is sensitive to layout, length and loading and is subject to speed limitations. The bus is confined to the backplane so that its characteristics can be predicted and controlled. The backplanes are engineered to be reliable with properly buffered STD cards. Because of the critical nature of the BUS, extension of the bus off the backplane is not recommended.

Functions can be extended away from the backplane when properly buffered.

Backplane Bus Design Considerations. The physical nature of the bus with its ordered layout of parallel signal traces causes the bus to act as a controlled transmission line. Transmission lines of this nature have the undesirable signal properties of crosstalk and line reflections. The ordered bus layout allows these properties to be predicted and controlled.

Crosstalk on a printed circuit bus backplane cannot be eliminated, however some element of control is possible with proper layout considerations. Uniform track spacing and the use of ground planes or ground track shielding are some techniques that should be considered. A uniform distribution of signal track to ground capacitance will provide some rise time control which will act to reduce crosstalk.

Backplane Terminations. The reflection characteristics of the bus signal lines can and should be controlled by proper termination. Ideal termination requires terminating each signal in its theoretical characteristic impedance. The backplane bus impedance by itself is predictable, however as cards are introduced into various slots along the bus the impedance can only be estimated.

There are various methods of providing physical termination. An example using passive AC termination for TTL is shown in figure 2-13. This circuit does not affect the DC drive and loading of the bus signal; however, high frequency ringing is effectively terminated to the characteristic impendance of the bus line.

Termination networks can be optimally located if the exact bus loading configuration is known and fixed. Since loading changes with the number and location of cards on the bus optimal termination location is impossible. Considering only first order effects, termination networks can be located either on the backplane motherboard or on a separate terminator card.

Signal Pull-Ups. Bus driving devices for LSTTL do not pull the bus signals to the full 5 volt high level. Full 5 volt logic swings can be achieved by the addition of pull-up resistors to the bus signals. Pull-ups may be located on the backplane motherboard or on a separate card.

A 1K ohm resistor is recommended for the LSTTL STD BUS as shown in figure 2-13.

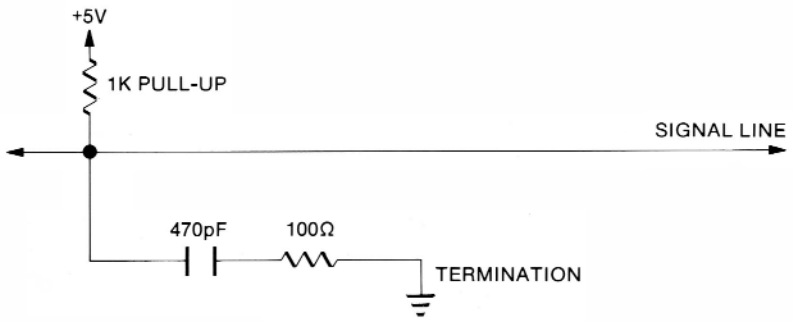

Figure 2-13. Passive AC Termination and Signal Pull-Up

Typical practice in a CMOS STD BUS system is to use the bus without termination. Any termination that is used should not exceed the rated load capacitance. The power drawn by termination should also be allowed for, since some termination schemes can draw a significant portion of system power. The need for termination is related to bus length and processor speed, as well as other factors.

Open-Collector/Drain Bus-Signal Practice

Bus control inputs to the processor card are often wire-OR connected, which requires open-collector/drain drivers. It is recommended, as STD Practice, that the following signals be open-collector/drain on any source card and pulled up on any destination card:

- BUSRQ* — Pin 42
- INTRQ* — Pin 44
- WAITRQ* — Pin 45
- NMIRQ* — Pin 46
- SYSRESET* — Pin 47
- PBRESET* — Pin 48

Also, it is recommended that these lines be specified as follows:

- Low-Level active signal.
- LSTTL or high speed CMOS logic levels.
- Driver sink capability of 16 mA at 0.4V for TTL or 6 mA at 0.37V for CMOS.
- Open-collector/drain driver with 10K pull-up.
- Destination load pull-up of 4.7K (TTL) or 10K (CMOS).

CMOS Load Capacitance Practice

The maximum load capacitance seen by any card should not exceed 150 pF. This load capacitance includes bus capacitance and input capacitance of all other cards in the system. Input capacitance for each card should not exceed 10 pF

CMOS User Edge Input Practice

Typical practice is to have no protection circuitry at the user edge of the STD card. However, if voltage transients are expected on the inputs, protective circuitry can be used to prevent latch-up and possible chip damage. Typically, the protective circuit is a current-limiting series resistor, or voltage-clamping diodes.

CMOS Pull-Up Resistor Practice

Since CMOS gates draw the most power when inputs are floating, pull-up resistors are sometimes required. It is recomended that these pull-ups be on the interior of the STD card whenever possible. Floating STD BUS lines are the responsibility of the current bus master. In the event of the transfer of bus control, the requester is responsible for floating STD BUS lines.