

USING A/D CONVERTERS WITH THE IOP-16

The IOP-16 is a DMA interface between the Main Data Memory of FPS 38-bit array processors and an external port. The basic form of the IOP-16 is a DMA controller which has no processor or decision making capability, and must be controlled by the systems to which it attaches. Occupying a single board, the IOP-16 has a set of registers to control its operation, and provides source and destination addresses and DMA word count. The IOP's external connection provides parallel address and data paths, and asynchronous handshaking control.

The IOP also is available as:

- ✓ IOP-38 General purpose 38-bit parallel interface (occupying 2 boards)
- ✓ IOP-UNI Unibus® interface with cable (occupying 1 board)

The options provide a low-cost interface to external equipment while minimizing the interface design effort. A common problem in scientific applications is the connection of Analog to Digital converters for data collection. The IOP-16 is the best choice because it includes signal line termination; and few A/D converters need more than 16-bits.

An IOP-16 can be connected to almost any A/D converter with a simple interface consisting of three parts:

1. Data buffers to provide proper open-collector drivers on the data lines (low true).
2. Slave-sync synchronizer to provide proper start-up synchronization between the converter's end-of-convert signal and the sync signal it sends to the IOP.
3. An address latch to allow the IOP to select one of many channels on an input multiplexer.

These requirements can be simplified further by assuming certain functionality available in the A/D conversion system. If the data and address lines are low true, then no buffers are required. If not, the channel selection may be inverted, and the data will appear opposite polarity.

For A/D conversion rates less than 300 kHz, a configuration similar to that in Figure 1 may be used. This configuration shows no external latch for channel selection.

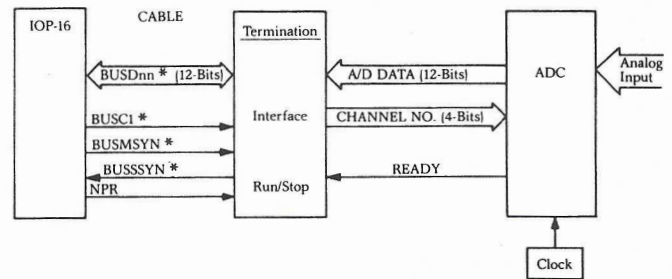


Figure 1. Block diagram of a typical interface between an IOP and a 12-bit A/D converter with an external sample rate clock.

An address latch may be required if A/D converters above 300 kHz are used, because the address may not be stable over the entire sample time. The IOP will have valid address and latch data at the edge of EOC. Figure 2 shows the logical connection of the A/D with the channel latch.

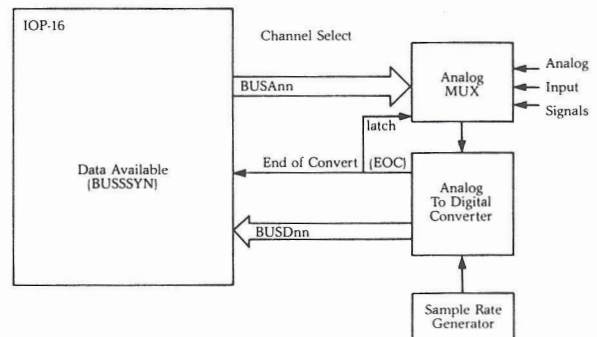


Figure 2. "Simple" A/D interface makes use of arbitration on IOP to synchronize data at external sample rate. The converter's control signals and logic levels are chosen to match those of the IOP.

A typical system implementation, using a 250 kHz, 16-channel board level module, is shown in Figure 3. Sample clock, power supply, and cable assemblies are not shown but are required. Although this is not off-the-shelf because of packaging considerations, it remains a simple solution to the problem of building a data acquisition system.

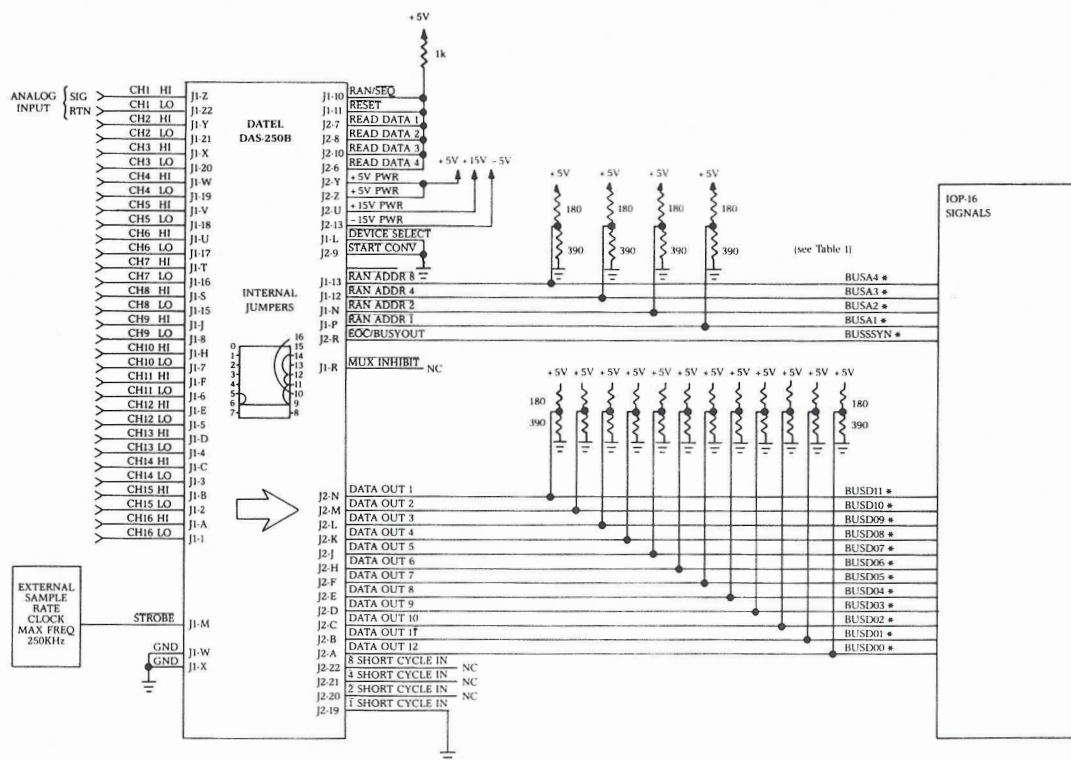


Figure 3. TYPICAL CONNECTION METHOD. Example shows correspondence of DAS250B signals to IOP-16 signals. The MUX channel is specified for every conversion via the address bus of the IOP. Data will be inverted as shown. (See Figure 3 notes.)

Figure 3 Notes

1. The DAS-250 has four READ DATA lines, which are constantly enabled to allow data to be read by the IOP in parallel.
2. The DAS-250 DEVICE SELECT line is grounded for constant selection.
3. The DAS-250 CONVERT line is grounded for this application.
4. The DAS-250 is equipped with a multiplexer capable of selecting any of 16 analog lines. The four DAS-250 RAN ADDR lines, which are connected to the multiplexer, are connected to the four least significant IOP address lines; the IOP thereby controls the address for selection of the correct input data.
5. The DAS-250 End of Conversion (EOC/BUSY OUT) line is connected directly to the IOP slave synch line (BUSSSYN *).
6. The 12 DAS-250 DATA OUT lines are connected to the IOP data bus. The remaining data lines may be tied to the signbit for 2's complement applications.
7. The DAS-250 1 SHORT CYCLE IN line is grounded to inhibit resetting the channel address to channel one after each conversion.
8. The signal line STROBE (J1-M, Figure 3) is used to control an A/D conversion. STROBE connects to a continuous clock signal with a maximum frequency of 250 kHz. With the duty cycle approximately 50 percent, STROBE goes low every 4 microseconds minimum and typically remains low for 2 microseconds. STROBE goes low, and A/D conversion begins. Two to three microseconds after STROBE goes low, the signal EOC/BUSY OUT goes low; this indicates that the conversion is completed and the data is available.
9. Logic level compatibility will vary with applications. The RAN inputs are low true and will match the low true address lines of the IOP. The data, however, will be inverted unless inverting buffers are used (not shown in Figure 3).

DAS-250 LINE	PIN #	IOP LINE
RAN ADDR 8	J1-13	BUSA4 *
RAN ADDR 4	J1-12	BUSA3 *
RAN ADDR 2	J1-N	BUSA2 *
RAN ADDR 1	J1-P	BUSA1 *
EOC/BUSY OUT	J2-R	BUSSSYN *
DATA OUT 1 (MSB)	J2-N	BUSD11 *
DATA OUT 2	J2-M	BUSD10 *
DATA OUT 3	J2-L	BUSD09 *
DATA OUT 4	J2-K	BUSD08 *
DATA OUT 5	J2-J	BUSD07 *
DATA OUT 6	J2-H	BUSD06 *
DATA OUT 7	J2-F	BUSD05 *
DATA OUT 8	J2-E	BUSD04 *
DATA OUT 9	J2-D	BUSD03 *
DATA OUT 10	J2-C	BUSD02 *
DATA OUT 11	J2-B	BUSD01 *
DATA OUT 12 (LSB)	J2-A	BUSD00 *

TABLE I
DAS-250/IOP Connections

The notes shown here are intended as a general introduction to the problem of building A/D converter interfaces with the IOP-16. Many A/D converter ICs, modules, and subsystems are available, and may be connected using these general principles. Information regarding the specific module mentioned above is available by contacting DATEL-INTERSIL, Inc., 11 Cabot Boulevard, Mansfield, Massachusetts 02048.

For additional information, contact the author, Ian Curington, FPS Product Specialist, (800) 547-1445, extension 783.

NEW 38-BIT RELEASES

Updated software, identified as Release D.3 or D.4, recently has been released for several hosts recently. The enhancements in these releases are described below.

D.3 Release Enhancements

- Symbolic MD Addressing on the Host—This feature eliminates the need to refer to blocks of data in MD memory by absolute addresses.
- Full 38-bit Triples—The triple data type has been extended from 32 to 38 bits, allowing the users to specify all the bits in an MD word.
- Updated Environment for GPIOP Software—The users can now use GPIOP software with APLOAD/LOD100 and AP-FORTRAN; this software formerly could be used only with APLINK/LNK100.

- 8K AP PS Support—All the software now provides support for 8K AP PS available for AP120B/AP180V/AP190L.
- Large HSR Support—APLINK/LNK100 has been modified to use a disk file for the HSRs, thereby allowing the HSR size to be limited only by the size of AP PS.
- Installation of GPIOP and AP-DISK Software—The GPIOP and AP-DISK software installation has been integrated into the AP software installation, thereby providing a consistent and uniform installation mechanism for all FPS AP-38 software.
- Free Format Input—APSIM/SIM100 and APLINK/LNK100 have been changed to enable the user to type commands in free format similar to APDEBUG/DBG100.

D.4 Release Enhancements

All of the D.3 enhancements are included, plus the following additions:

- APLOAD/LOD100 Table Sizes—The Default table sizes are now host dependent. The tables are larger for those hosts that typically have more memory space available. APLOAD/LOD100 can handle more than 128 external symbols on machines with adequate memory space. Typical table sizes are:

	Large Host	Small Host
External Symbols	200	70
Subroutine Names	200	100
Common Block Names	200	70

- APLOAD/LOD100 Bug Fixes—The /SIZE option on the OUTPUT command now works.
- IOCAL Table Sizes—IOCAL has been modified to handle larger program sizes. The Default table sizes are now host dependent. The tables are larger for those hosts that typically have more memory space available. Typical table sizes are:

	Large Host	Small Host
Max IOCAL Program Size (AP words)	8192	512
Number of Symbols	700	200

- IOCAL Usage with APLOAD/LOD100—The object output is now compatible with APLOAD/LOD100. \$SUB in IOCAL produces both PS and MD external symbols. The PS size is zero. This feature allows extraction of IOCAL code from Libraries without requiring the FORCE statement in APLOAD/LOD100. Also, IOCAL accepts a \$EXT pseudo-op. The named symbol must be an MD symbol such as IOCAL code or MD Common. This allows an IOCAL program to call other IOCAL programs that reside in a library file.

IOP-UNI APPLICATIONS

Abstract

The IOP-UNI is one of several I/O products which provides an interface mechanism between FPS array processors (AP) and a variety of customer-supplied external devices. This article explains the operation and architecture of the IOP-UNI, highlighting its unique features and common uses. Differences between the IOP-UNI and the very similar IOP-16 are outlined, as well as applications in which the IOP-UNI is used to solve system integration problems.

Description of IOP-UNI

38-bit FPS array processors, such as the AP120B, FPS100, AP180V, and AP190L, allow optional I/O devices to be configured for secondary data paths in addition to the host interface. The IOP-UNI provides a simple, 16-bit path to or from Main Data memory (MD) in the AP. A cable is provided to connect the single board IOP-UNI to an external Unibus® system. In the proper environment, the IOP-UNI operates as a peripheral in a PDP-11 or similar Unibus system. The IOP-UNI product includes the control board (currently FPS type number 437), the IOP-UNI cable assembly, IOPLIB utility software library, and an IOP users guide.

IOP-UNI Unibus Characteristics

In order to ensure compatibility certain guidelines must be followed when using the IOP-UNI with other Unibus devices. The register access and DMA read or write operations of the device all occur within the Unibus memory-mapped address space. The IOP-UNI requests use of the bus and performs DMAs similar to other Unibus devices, such as a DR11-B. An IOP-UNI does not provide a complete Unibus environment. For example, it does not generate or arbitrate interrupts on the Unibus, nor does it provide sources for all of the bus arbitration signals (functions normally provided by a Unibus processor). The IOP-UNI does impose one logical load on Unibus signals, which should be terminated by an external source.

A special cable is provided for Unibus connections. The IOP-UNI cable is a "Y" configuration; each end is a Unibus paddle card (see Figure 1). All Unibus signals are carried through this cable, so that it can be used in place of a jumper cable to connect two adjacent Unibus backplane sections. At the center of this cable, three 50-pin insulation displacement ribbon cable connectors are placed to provide connection to the IOP interconnect on the AP chassis. From the AP's I/O interconnect bracket, signals are wired to the IOP backplane slot.

Only those signals needed for the control board operation are wired at this point; hence not all Unibus signals are present on the IOP-UNI control board. Optional termination resistors may be installed for any of the signals available on the IOP. For this reason, an external Unibus terminator is recommended to ensure that all signals are terminated properly.

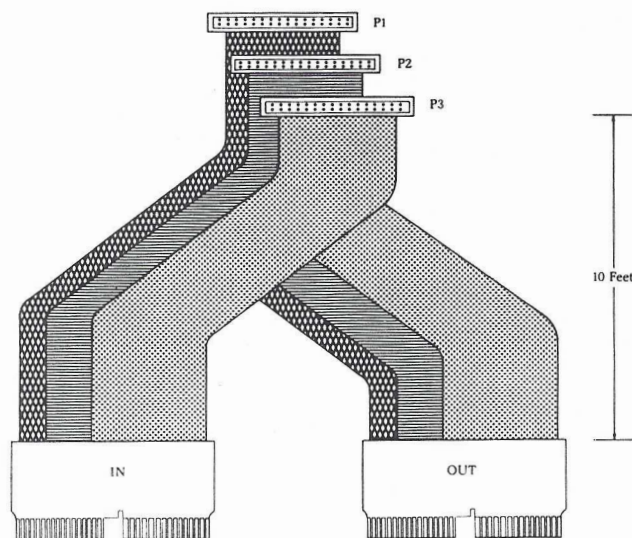


Figure 1.
IOP-UNI cable is designed for loop-through connections.

IOP-UNI vs. the IOP-16

Because both the IOP-UNI and the IOP-16 are 16-bit wide DMA devices, they use the same control board in the AP. Strap options and termination options form most of the functional differences. The IOP-16 does not include the Unibus cable assembly, but instead is supplied with three standard 50-pin ribbon cables, with identical connectors on each end. The IOP-16 has the Unibus arbitration response circuitry disabled, and is populated with termination resistors for signals brought to the chassis interconnect. The IOP-16 can be used in simple dedicated device connections where it is at the electrical end of the cable system. With board strap changes, either the IOP-UNI or the IOP-16 can be tailored to meet specific interface requirements. The IOP User's Guide includes a chapter on optional strapping, and shows the position of the straps for both products. Nothing in the AP chassis is specific to one or the other, so given an available IOP slot, either model can be added later or changed.

Unibus Configurations with the IOP-UNI

The IOP-UNI can be used in many configurations of Unibus equipment, meeting varying system requirements. In some cases it is used as a DMA path to a dedicated peripheral device. In others it is a foreign device in a Unibus computer system. If connected to a PDP-11 computer (via Unibus), several cabling options are available. Figure 2A shows the simplest configuration, in which the IOP-UNI is tied into the host computer system. The PDP-11 processor is at the top end of the Unibus, with the IOP-UNI cable bringing the Unibus out from the last slot in the host. From the IOP, the cable connects to the AP host interface where a Unibus terminator is plugged into the loop-through position. In this mode the AP is the last device on the bus, and the IOP-UNI is next to last. If the IOP-UNI requires higher priority than the AP or some other device, the configuration shown in Figure 2B is a possible connection. The Unibus is brought out from the middle of the host backplane, returned to the next host backplane section, and then brought out again at the end for the AP. This method extends the Unibus at least 30 feet (20 feet for the IOP-UNI cable, and 10 feet for the AP), and must be evaluated carefully to avoid bus problems.

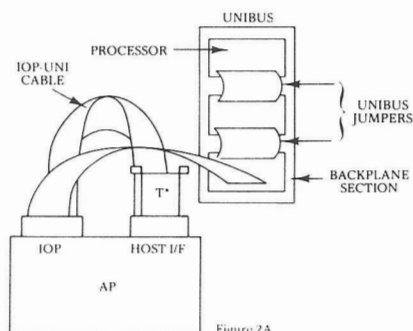


Figure 2A

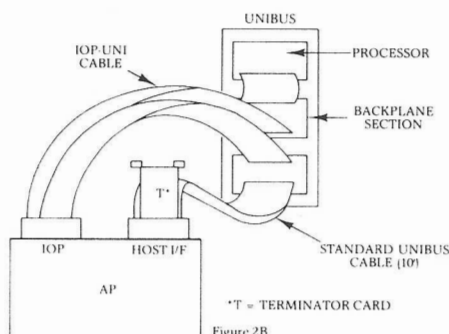


Figure 2B

Figure 2A, 2B.
Methods of Connecting an IOP-UNI to the HOST UNIBUS

Having an IOP-UNI tied into the same Unibus as the host interface can offer useful advantages in real-time simulation. A block of PDP-11 memory can be allocated as a message passing or simulation variable table. The AP can direct single word reads or writes to or from this memory under AP application program control. The overhead to start the IOP-UNI is small, about 5 microseconds assuming the IOP-UNI is idle and the bus is available. Imbedding this type of control structure in the AP application code (such as VFC or APFTN38) can speed up host/AP communication considerably, when the alternative is using APEX and the host operating system services.

In some applications two separate computer systems are involved. For example one PDP-11 could be used as a "front end" data acquisition system, with the AP performing analysis, and a second PDP-11 acting as both post processor and AP host (see Figure 3). The advantages of this configuration is pipelined operation of several functions, as well as the ability of the IOP-UNI to grab data directly from the front-end memory without operating system intervention.

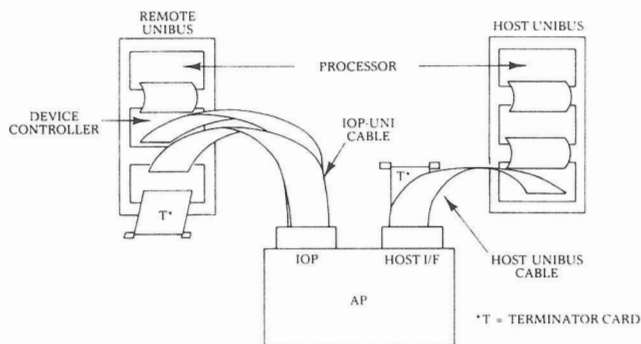


Figure 3.
Remote UNIBUS System Connections.

Figure 3 shows a remote Unibus system interfaced to the AP via the IOP-UNI, and illustrates several important functions of the PDP-11 processor. The remote processor provides a source for bus arbitration signals, fields interrupts, and provides device initialization. All of these functions cannot easily be performed by an IOP-UNI. At the opposite end of the Unibus, a terminator card provides signal termination.

In applications in which an intelligent device is interfaced to the IOP-UNI, the application has the choice of several control methods. Since the IOP-UNI's control registers are memory mapped externally, the remote PDP-11 can specify and control its operation invisibly to the AP. This mode allows for data-driven timing, in which the remote system provides the data to the AP.

However, if the AP controls the IOP-UNI, the remote system has only to allocate a buffer area of its memory for message passing, while the AP invisible directs access to the buffer. No register access arbitration is available; therefore, only one or the other should have control.

Dedicated Device Operation

The system shown in Figure 3 can be modified to form direct communication between the IOP-UNI and a peripheral controller. If the peripheral controller in the remote system does not need all the processor-generated arbitration signals, and can operate without generating interrupts, the processor can be removed. This case typically occurs only when one device is connected to the IOP-UNI.

A feature of the IOP-UNI is its ability to hold the Unibus address constant throughout a DMA, while pointing to a data buffer in the peripheral controller. If the controller can supply data in this way, a Unibus memory buffer can be avoided.

Some controllers are designed to communicate via DMA into Unibus memory. These may require the use of an intermediate memory buffer unless they also are able to hold a fixed address during a DMA. The destination of the DMA can be the data buffer register of the IOP-UNI (which is mapped into the I/O page) and the IOP-UNI will automatically advance the internal MD address pointer and perform a write. The IOP-UNI registers must be set up before such a transfer takes place, either by the AP or by the remote processor.

Example Applications

Dedicated devices constitute the majority of the IOP-UNI applications. The configuration shown in Figure 2A typically is used by the FPS Test Department during the testing of IOP-UNIs and IOP-UNI cable assemblies before shipment. For control of a Unibus display system such as a DeAnza 8500 Image Processing system, the IOP-UNI can be connected into a Unibus expansion chassis without the use of a PDP-11 system. The Unibus interface of the DeAnza unit is register-command driven, so the IOP-UNI can deposit commands and read status from the interface. The IOP-UNI must always have access to the bus and operate in programmed I/O mode.

Building a high performance data acquisition system can take the form of using a Q-bus chassis, Q-bus compatible A/D converters, and using a commercial adapter to convert from Unibus to Q-bus. Figure 4 shows the major system components of this design. With local memory on the Q-bus and an inexpensive

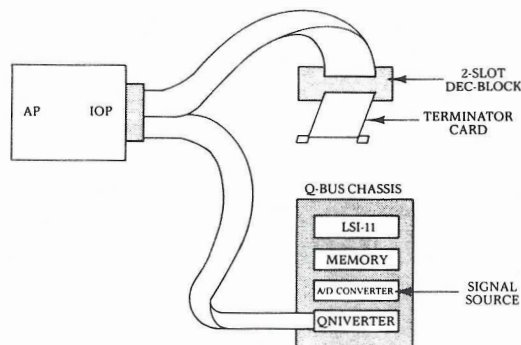


Figure 4.
Using IOP-UNI Interface to Data Acquisition System.

LSI-11 processor, testing of the A/D system can be accomplished without writing specialized array processor drivers.

*Unibus® is a registered trademark of Digital Equipment Corporation.

References

1. "PDP-11 Bus Handbook," Digital Equipment Corporation, 1979, Maynard, Massachusetts.
2. "IOP-16/UNI/38 User's Guide, Release D.3" Floating Point Systems, Inc. (860-7310-004D), 1982, Beaverton, Oregon.

For additional information, contact the author, Ian Curington, FPS Product Specialist, (800) 547-1447, extension 783.