An I/O Subnet for the APS Control System – The BITBUS Universal Gateway*

G. J. Nawrocki, N. D. Arnold, M. G. Hoffberg, J. R. Winans, S. J. Benes
Argonne National Laboratory
Advanced Photon Source
9700 South Cass Avenue
Argonne, Illinois 60439

Abstract

The Advanced Photon Source (APS) control system is based on a distributed topology of microprocessor–based Input/Output Controllers (IOCs). Since the cost effectiveness of placing an IOC near every point where an interface to the control system is required may be prohibitive, I/O subnets implemented via message passing network protocols are utilized. For greatest flexibility, such a subnet must support connections to equipment via discrete I/O points, connections to standard interfaces such as GPIB and RS232, and be a practical network for custom–designed interfaces to intelligent equipment. This paper describes the BITBUS Universal Gateway (BUG), a device which supports the different interfaces mentioned above with a connection to a single BITBUS distributed subnet. The BUG utilizes an interchangeable set of circuit boards, which allow for a commonality among interface points, and the ability to use commercially–available modules for I/O. This approach also circumvents several limitations of GPIB and RS232, which restrict their use in industrial, electrically harsh environments, via an implementation of the BITBUS protocol over optical fibers.

I. SUBNETS IN THE APS CONTROL SYSTEM

The APS control system provides for VME–based Input/Output Controllers (IOCs) to be distributed throughout the facility and interconnected via Ethernet to one another and also to UNIX–based Operator Interface (OPI) consoles. Although this distributed architecture allows for intelligent processors near the major subsystems, I/O subnets are frequently required to interface directly to the equipment and communicate I/O information to the nearest IOC. [1]

II. CURRENT INTERFACE PROBLEMS

Currently GPIB (IEEE–488) and RS232 interfaces are being used to interface instruments to the APS control system. These interfaces have severe limitations when used in an industrial, non–office type environment for computer control systems. GPIB offers no error detection mechanisms, no ground isolation, and severe distance limitations. Although both fiber optic and twisted pair extenders are available for GPIB they are cost inefficient. RS232 offers no ground isolation, severe distance limitations, and exists with a single node master/slave topology. Again, extenders and multidrop RS232 network solutions are available, however, their repeated use can be expensive.

*Work supported by U.S. Department of Energy, Office of Basic Energy Sciences under Contract No. W–31–109–ENG– 38.

Basic binary and analog I/O have obvious distance and noise immunity problems when interfaced to a control system. Currently the Allen–Bradley 1771 series I/O modules provide remote interfacing for these types of signals in the APS control system. Although effective, it is best used for a multitude of signals as its use becomes cost prohibitive when used with just a few raw binary or analog signal points. An additional complication with the Allen–Bradley solution is that the network and I/O chassis are proprietary designs of Allen–Bradley. Modules cannot be customized to meet the unique requirements of the APS control system.

III. THE BITBUS SUBNET

An ideal subnet for the APS control system should provide a distributive, homogeneous solution to differing control interface topologies. The design should be based upon a non–proprietary commonly accepted network which provides the ability to communicate control signals to multiple nodes, over distances up to several hundred meters, and in electrically harsh environments. The subnet must provide a "gateway" for GPIB, RS232, and raw binary and analog I/O signals. BITBUS† was selected as the subnet of choice for this type of interface to the APS control system. [1]

IV. THE BITBUS UNIVERSAL GATEWAY

To implement BITBUS as a subnet in the APS control system, a BITBUS slave node was developed to meet the previously defined criteria. This instrument has taken the form of the BITBUS Universal Gateway, or BUG.

A. Hardware Description

The BUG acts as a slave node on the BITBUS subnet and provides a communication link between BITBUS protocol and other computer–signal interfaces such as GPIB, RS232, and discrete binary and analog I/O (see Figure 1). The BUG is housed in a commercially available plastic enclosure measuring 2.25H x 5.08W x 5.25L. Located on this small enclosure are connections to the BITBUS subnet, I/O points, and power supply. There are red and green CPU status LEDs, and a series of eight diagnostic LEDs which can be written with a byte of data for additional status indication. Power is generally supplied by a small "calculator type" wall plug adapter or brick type power supply. The entire part cost for each BUG node is under $400.

B. The BITBUS Interface Board

The BITBUS interface board is the BUG's link to the BITBUS subnet. Communication over the BITBUS subnet in the APS control system is self clocked at a speed of 375Kb/s. Origi-

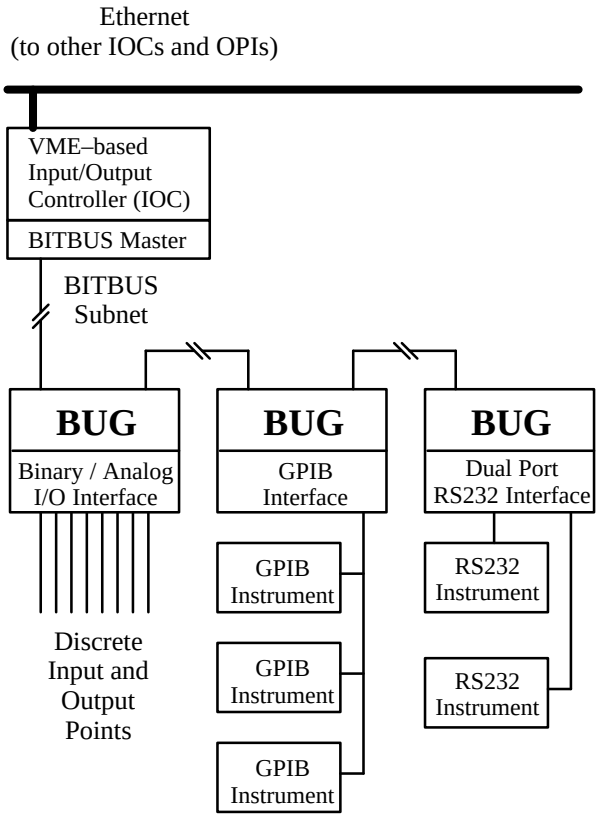

Figure 1. BITBUS Subnet at the Advanced Photon Source

nally this separate board for the BITBUS interface was chosen so that communication could be implemented over either a "twisted pair" RS485 subnet, or a fiber optic interface. The RS485 subnet has been abandoned at the APS in favor of the more noise immune and cost effective fiber optic interface.

The fiber optic BITBUS interface card is constructed using transmitters and receivers with AT&T ST type connectors. These bayonet–style connectors provide a simple and accurate fiber connection. The BUGs in a fiber optic BITBUS subnet are connected in a daisy chain configuration, linked from the VME IOC BITBUS master, and serially one after the other with duplex (two fibers per cable) fiber optic cable. The APS BITBUS fiber optic subnet utilizes standard 62.5/125μm fiber optic cable.

C. The BITBUS Controller (CPU) Board

The CPU board is the middle board of the BUG three–board configuration. It contains the Intel 8044 BITBUS enhanced microcontroller, data memory, code memory, and additional support circuitry. The 8044 provides both the processing and communication ability of the BUG. Unique firmware for each type of interface resides in the external code memory of the BUG. The external code memory of the BUG uses either a 256Kbit RAM chip, or an identical size (E)PROM chip. The type of memory is selected by a configuration jumper. Software may be down–loaded over the BITBUS link to the external code RAM, which is useful for BUG software development. When suitable code is developed, a PROM, or EPROM may be created for more permanent use.

D. The I/O Board

The I/O board is the BUG's interface to the controlled equipment. The I/O board interface was chosen to be the iSBX† (single board extension) bus. Since the iSBX bus is an Intel and IEEE standard (IEEE 959–88) there are many such I/O boards commercially available. These boards, however, can often be made "in house" at a fraction of the cost of commercially available products. Currently at the APS there has been "in house" design of both hardware and software for the following boards: a single port GPIB interface with the ability to control 15 GPIB instruments, a dual port RS232 interface, and a discrete binary and analog signal interface with 16 optically isolated binary inputs, eight optically isolated binary outputs, four 12–bit analog inputs, and two 12–bit analog outputs.

E. The Interface to the APS Control System

The BITBUS subnet interface to the APS control system is accomplished through the BITBUS master. The master is a modified BITBUS node with an interface to the VME host computer bus. Currently the BITBUS subnet master in the APS control system is the Xycom XVME–402 VME module modified for use with the APS BITBUS optical fiber subnet.

V. THE GPIB BUG AT THE APS

A basic timing study was done on GPIB message passing over the BUG link compared to GPIB message passing using the National Instruments GPIB 410 VME module. The GPIB instruments being controlled were four identical Hewlett Packard 34401A digital multimeters. The four multimeters were initially connected directly to the GPIB 1014 VME module and a series of read–back commands were sent to them. Next, the four multimeters were connected to a single BUG which in turn was connected to the VME IOC via the BITBUS subnet. The BUG showed no significant decrease in data throughput for this read command.

Next, two BUGs were connected to two multimeters each, and placed on the same BITBUS subnet. Following that, four BUGs were each connected to one multimeter each and placed on the same BITBUS subnet. There was an increase in read–back command data throughput in each case. The single BUG per multimeter configuration even showed faster read–back data throughput times than the single GPIB 1014, four multimeter configuration. This increase in read–back data throughput is due to the fact that when a GPIB instrument is sending data back to the control system using the GPIB bus, it takes command of the bus, thereby allowing no other instruments to use the bus until it is finished. The remaining instruments must wait in turn for the previous instrument to send its data back to the control system before they may respond. When multiple BUGs are used, the per instrument distribution on the GPIB bus decreases. Commands and responses may be sent to, and read back from, multiple BUG–isolated GPIB instruments at the same time. The BUG will in turn relay the data to the control system. Since the BITBUS subnet does not need to wait for individual node responses before commands are sent out, data throughput increases.

The multimeter used for the timing study was a relatively quick responding instrument. Far more significant increases in

data throughput were evident when instruments, which upon receiving a command take several seconds to complete a calculation or data read before replying, were isolated behind GPIB BUGs on the BITBUS subnet.

A disadvantage to using the BUGs was discovered when interfacing to GPIB instruments that send several hundred data bytes back to the control system in response to a single read command. The current implementation of the VME BITBUS master limits each BITBUS message to 13 data bytes. When an instrument such as an oscilloscope was connected to a BUG and a waveform was sent back to the control system, the limited message size required a multitude of BITBUS messages to be sent to the control system as the result of a single read command. This, combined with the limited speed of the BITBUS subnet, produced a bottleneck.

The BUG, in its current state, is not recommended for such large data transactions. Fortunately, the vast majority of instruments in the APS control system respond with fewer than 50 bytes as the result of a single read command. For these applications the BUG is well suited.

Simple GPIB write commands that implement instrument control and require no response are a different issue. The GPIB 1014 VME interface is almost eight times faster than an individual BUG interface for sending this type of command. However, the data throughput for GPIB write commands remains constant on the GPIB 1014 regardless of the number of GPIB instruments added to the link. Write commands to GPIB instruments, isolated behind GPIB BUGs, show a near linear increase in data throughput as GPIB instruments connected individually to BUGs are added.

VI. THE RS232 BUG AT THE APS

In addition to the multidrop and distance extension abilities of running RS232 over BITBUS, another advantage was discovered. RS232 instruments are often very different in both their command sets and how they respond to commands. For instance, some instruments may or may not echo each received byte back to the control computer. Some units respond to commands after a carriage return, others need the command placed in parentheses. A variety of instruments return data with a carriage return and/or line feed, others use odd termination or data return framing characters. For these specialized cases, unique software for each type of controlled instrument may be created for the BUG. Despite this additional software design work, there are distinct advantages to allowing a universal version of RS232 software to reside at the point of the control system IOC.

The BUG can also be given the ability to remove worthless data bytes or interpret data sent by an RS232 instrument into a more concise package to be sent to the control system. For these reasons, the RS232 BUG is currently the only recommended method for RS232 instrument interfacing to the control system.

VII. CONCLUSION AND FUTURE PLANS

The BUG has proven itself as an extremely popular method of GPIB, RS232, and discrete I/O signal interfacing to the APS control system. Already in this early stage of construction at the APS, as many as 35 BUGs are in operation in various test stands and sections of the accelerator. Optimization of the BUG hardware and software will continue by reducing the size of the BUG to a two–board configuration due to the exclusive use of the fiber optic BITBUS interface, investigating faster VME BITBUS masters, possibly increasing the size of the BITBUS message to greater than 13 bytes, and adding more supported I/O modules.

VIII. REFERENCES

[1] N.D. Arnold, G.J. Nawrocki, R.T. Daly, M.R. Kraimer, W.P. McDowell, "I/O Subnets for the APS Control System," *Proceedings of the 1991 IEEE Particle Accelerator Conference*, pp. 1496–1498, 1991.

† BITBUS and iSBX are registered trademarks of Intel Corporation and its affiliates.