SLAC-PUB-3236
October 1983
(I)

Status of the IEEE P896 Future Backplane Bus*

David B. Gustavson
Stanford Linear Accelerator Center
Stanford University, Stanford, California 94305

Abstract

The IEEE P896 Future Backplane Bus project has been influenced by and has influenced FASTBUS and several other contemporary bus designs. This paper summarizes the current status of that project, which is directed toward the needs of modern 32-bit microprocessor systems with multiple processors.

Some of the technology developed for P896 will be important for future non-ECL implementations of FASTBUS and other buses. In particular, new bus drivers and receivers should greatly improve the performance and reliability of backplane buses and cable buses.

The current status of the P896 serial bus is also summarized.

Introduction

The IEEE P896[1] project began as an outgrowth of standardization work on the S-100 bus (IEEE 696) and Multibus (IEEE 796) in 1978, becoming an independent project in June 1979. There has been a long history of interaction between the IEEE P896 project and the FASTBUS[2] (IEEE P960) project. Both projects are designing 32-bit address/data multiplexed buses, but FASTBUS interconnects multiple multiprocessor backplane bus segments and cable bus segments, while P896 is a single multiprocessor backplane bus.

This paper will use FASTBUS as a reference model with which to compare P896, on the assumption that this audience is generally familiar with and has an interest in FASTBUS already.

P896 is more closely tailored to the needs of microprocessors, dealing with the problems of byte addressing and part-word transfers. FASTBUS is optimized more for large data-acquisition systems, and supports only full-word addressing and full-word transfers on the bus.

P896 has had a long and painful evolution, and has nearly become extinct several times along the way, but now appears likely to reach completion.

The Evolution of P896

P896 started with a generally-perceived need for a 32-bit high-performance bus standard. Many individuals joined the effort, anxious to see an elegant design which would solve their past and future system problems. Representatives of various industrial concerns appeared as well, but the ones in the microprocessor business, whose needs P896 was presumably addressing, did not usually stay long. Generally they expressed doubts that such a committee could do anything useful and timely in a subtle area like computer bus design, and left to continue independent work on proprietary buses.

Several manufacturers offered their own bus designs as a basis for standardization. All were examined and rejected, though sometimes with useful criticisms which were adopted by the manufacturer to improve his proprietary product. A subset of FASTBUS was proposed and similarly rejected.

Then a significant European influence appeared, in the form of an EDISG (European Distributed Intelligence Study Group) subgroup. Out of this collaboration came a bus design which was prototyped and tested, written up as a draft standard, and submitted to the supervising committee of P896, the Microprocessor Standards Committee of the IEEE Computer Society.

This draft was rejected by the MSC in January, 1982, on the grounds that it failed to meet the speed objective and that the electrical specification was not adequate for reliable operation. This rejection caused the resignation of the P896 chairman, and the project nearly died during the next year because of personnel turnover and divergent opinions within the working group. The fourth (and present) chairman was able to revive wide interest by bringing in many new participants with relevant experience, and has kept the project on a tight schedule, producing the present draft 6.1b which is about to be considered by the MSC.

The speed problem was related both to the electrical problems and to the choice of handshake protocol, which was optimized for permitting intervention by a third-party bus supervisor. The handshake protocol was speeded up by adopting a modified form of the FASTBUS protocol, sacrificing supervisor intervention capability.

Examination of the problems of the electrical specification led to a deeper understanding of the physics of buses, and after considering many approaches to a solution, a new type of bus transceiver[3] was developed by a manufacturer to meet the needs.

The committee has been fairly consistently in agreement that the P896 bus will be packaged in the IEC standard "Eurocard" mechanical structure, and will use the "DIN 96-pin" (IEC 603-2) connector. There has been strong pressure from time to time to have a 64-pin subset standard, but due to the passage of time and the availability of competing buses and the difficulty of defining a compatible subset, this pressure has now nearly vanished.

The 96-pin connector turns out to be a severe constraint in the design of a 32-bit bus (FASTBUS uses 132 pins), which has helped keep the committee from adding features recklessly, whatever the technical shortcomings of the connector.

* Work supported by the Department of Energy under contract number DE-AC03-76SF00515.

Invited paper presented at the Nuclear Science Symposium, San Francisco, California, October 19-21, 1983.

Another point of general agreement has been the principle of distributed control, permitting no central failure points except the bus and terminators. This has led to significant complications relative to FASTBUS in the bus arbitration and broadcast mechanisms.

Technology independence, in the form of a fully asynchronous protocol, has also been accepted as a requirement. Though it is not possible to make a perfectly asynchronous system (timeouts and skew delays violate the principle), P896 has carried the principle about as far as possible, again paying some price in complexity.

However, P896 is now also pursuing a synchronous version based on the MIT Nubus as developed by Western Digital and Texas Instruments, with modifications due to interaction with Intel's Multibus-II and others in the P896 committee. The synchronous version is to share the electrical and mechanical specifications with the asynchronous version, and be compatible otherwise where possible. Both versions are expected to support the same functionality.

This effort was motivated by a proposal from the developers of the MIT/Western Digital/TI Nubus, and by the information that the industry's new high-performance buses are mostly going to be synchronous, e.g. Multibus-II. The backers claim that synchronous buses are much simpler, easier to test in production environments, and that they are now faster than the asynchronous buses and will remain so for several years. Furthermore, asynchronous buses are reaching fundamental physics limits which will prevent them from ever becoming much faster than today's synchronous buses.

Despite the participants' claims that each company is willing to bend toward a single standard, it seems likely that none will bend far enough, because of their own proprietary interests, and so a future P896 synchronous bus will likely be just one of a family of industry standards. These buses will share much more commonality than they would have if there had been no P896 interaction, however.

Discussions in the P896 forum have benefitted industry and users in many ways, and these benefits alone would be enough to make P896 a worthwhile effort even if no useful standard ever emerged from the committee.

Interaction with FASTBUS

P896 adopted the basic FASTBUS arbitration mechanism, as did IEEE 696, but with modifications to make it fully asynchronous and to eliminate the central arbitration control used by FASTBUS.

FASTBUS adopted an arbitration protocol which it calls Assured Access from P896, where it is called Fairness. This protocol allows round-robin scheduling of most bus users regardless of priority, eliminating bus starvation of low-priority modules, while permitting strict priority scheduling of a few (or all) modules if desired by the system configurer. P896 subsequently changed its scheme again, to a rather less elegant one, partly because P896 now uses a module's position on the backplane as its normal arbitration priority. An extra arbitration bit was added, to signal the use of strict priority and to provide an additional set of priority levels which are independent of position. A second additional arbitration bit was added to provide parity checking. Since each bit adds significant delay in today's P896 technology, the non-essential bits seem likely candidates for future removal when actual performances are measured.

P896 has also adopted a modified form of the FASTBUS protocols and handshakes for its asynchronous version. Some of the differences are arbitrary, but others reflect differences in the goals of the bus, or different opinions about the best tradeoff of complexity versus central shared control in order to achieve the best reliability.

FASTBUS uses two handshake signals, AS and AK, to control the addressing and connection of the two communicating devices. For broadcasts, the AK response is generated by a central piece of logic on the back of the backplane, since no individual module can be selected to provide it. P896 adds a third signal, AI, which is the logical inverse of AK, and which allows the slowest module on the backplane to set the timing of the operation. Thus, the first responder causes the assertion of AK, but AI will not be de-asserted until the last responder's response. This relies on a Wire-OR, with its transient problems[4], so integration of signals and additional delay in the Master is required (another violation of true asynchronous philosophy). Similarly, P896 adds DI to the FASTBUS DS and DK so that the slowest responder controls the transfer rate, while FASTBUS generates all responses in the backplane support logic, and relies on modules meeting a minimum-speed specification for broadcast cycles.

The P896 broadcast protocol, combined with byte-lane-disable signals which can turn off the active drivers, thus provides the capability for cycle-by-cycle intervention by a third party, such as the bus supervisor provided in the previous draft. However, P896 did not accept the FASTBUS WT (wait) signal which FASTBUS uses for diagnostic purposes in lieu of the supervisor concept, and so no provision exists for diagnostic intervention except during broadcasts.

In FASTBUS, the RD read signal is on a bus line and is timed so that it can be used to steer the bus interface chips directly. P896 includes the read signal in a command field instead, and uses it during the address cycle to provide a warning to a processor that the following cycle is a read or a write. During data cycles, the read signal applies to the current cycle itself instead. FASTBUS allows a mixture of single word and block transfers to a given address during a given connection, by providing appropriate signals on dedicated control lines. P896 uses the command field during data cycles to control byte-wide data lanes for part-word transfers, and can thus only signal the type of transfer during the address cycle.

The FASTBUS protocols were designed to support circuit-switched interconnection of multiple bus segments dynamically, as required. P896 avoids this mode of operation entirely, and relies on packet forwarding, gateways etc. for inter-bus communication. However, P896 permits multiple buses on the multiple backplane connector positions of the Eurocard system. Thus a processor may choose from several buses to avoid contention and to provide redundant paths for reliable communication. Non-P896 buses are also allowed to occupy these connectors, providing a convenient way to interface to a different bus for I/O or other purposes.

P896 has 2^{32} bytes addressable on the backplane,

while FASTBUS has 2^{32} 32-bit words addressable in the entire connected system. P896 uses part of this address space for accessing control and status registers in a module position-dependent way, allowing 2^{20} bytes per module. FASTBUS provides each module with its own 2^{32}-word internal address space for control and status registers, by using an additional cycle to transfer the internal address pointer, and allows connection either through a position-dependent (geographic) address or through any of the module's normal logical addresses.

Bus-Driving Technology

FASTBUS was forced to the ECL (Emitter Coupled Logic) technology by bus performance considerations which showed severe disadvantages to the use of the usual TTL open-collector or tristate bus transceivers, and P896 has also deserted pure TTL for these reasons. However, semiconductor industry support of P896 development has resulted in the design of special bus transceivers[3] which eliminate the TTL disadvantages while retaining TTL characteristics on the side away from the bus. These circuits use slew-rate limiting and filtering in the receiver to eliminate crosstalk and many other kinds of noise, and have low capacitance and high impedance to eliminate bus loading and impedance shifting. They also use low voltage swing to contribute to these performance objectives.

Though the use of special circuits violates a highly desirable objective of P896, the problem is so severe that this step is necessary. Buses which use TTL or CMOS drivers presently only work if they incorporate (perhaps inadvertently) delays which allow the bus to settle after transitions, or if the buses are not loaded in a worst-case manner.

The fundamental problem is that the impedance of even a well-designed backplane is greatly reduced due to the distributed capacitance of connectors, plug-in board traces, and transceivers. Furthermore, the impedance varies from point to point along the backplane, depending on the distribution of plugged-in boards. In order to achieve reliable, glitch-free signalling on both rising and falling edges of signals, it is necessary to terminate the bus with the right compromise values of resistors and to drive the bus with enough current so that turning off a driver generates an adequate instantaneous voltage swing. That voltage step must be large enough to carry the signal above the threshold for triggering the receivers, and stay above the threshold even after reflections due to imperfect termination or varying impedance arrive.

The optimum termination resistors may be quite small. P896 specifies 39 ohms at each end. Ordinary TTL drivers cannot supply enough current for TTL logic voltage swings (well over 100 milliamperes), and have other undesirable properties such as very fast edge transition rates. Reducing the voltage swing and controlling the edge rates brings the problem under control, however, when a receiver with corresponding specifications is used. In addition, adding an isolating diode in series with the driver reduces the capacitive loading of the bus significantly.

A suitable transceiver is under development at National Semiconductor, as a result of interaction with P896. Three parts are planned: an octal transceiver, DS3896, with common enable signals; a quad transceiver with independent drivers and receivers and common enables as well as individual driver enables; and an octal version of the original DS3662, the DS3862, without the capacitance-reducing driver isolation diodes, optimized for cable timings. Sample quantities of these circuits are expected during the first quarter of 1984. Furthermore, other manufacturers have expressed a willingness to second-source them.

FASTBUS, meanwhile, is developing another solution to a bus-driving problem, using differential current sources to allow true superposition and Wire-OR behavior on a differential cable bus[5].

The Serial Bus

The Serial bus concept, which was introduced to FASTBUS as part of the diagnostic system, has been adopted in P896 in a different form and for other reasons. Other industry buses have also adopted the idea, and an effort is underway to bring all to a single standard.

There are many different opinions about what the use of the Serial bus should be, if any. Some think of it as a general-purpose network; some wish to use it as a redundant, though slow, communication path if the parallel bus should fail; some think it should implement special functions that the parallel bus cannot handle easily; and others think it should be removed to free two pins for other use.

This divergence within the committee seems to mirror the divergence within the industry, so it is not clear whether any single standard can develop and survive. Multibus-II, for example, expects to provide a serial-bus equivalent for every parallel-bus function; this would seem likely to be incompatible with other serial-bus designs.

The version of the serial bus in the current P896 draft is fairly general-purpose, providing for emergency system control, I/O-type activity, and providing facilities for supporting certain types of multiprocessor task dispatching algorithms. It is an optional feature, however, and is not relied on in any way by the parallel bus.

References

1. 896/D6.1b, A Proposed Standard Backplane Bus Specification for Advanced Microcomputer Systems, July 1983, available from Paul Borrill, Chairman, IEEE P896 Working Group, University College London, Mullard Space Science Laboratory, Holmbury St. Mary, DORKING, Surrey, RH5 6NT, England.

2. FASTBUS Specification, November 1982 (Rev. 15 July 1983), U. S. NIM Committee, available from L. Costrell, Dept. of Commerce, National Bureau of Standards, Washington, DC 20234.

3. R. V. Balakrishnan, "Cut Bus Reflections, Crosstalk with a Trapezoidal Transceiver", EDN August 4, 1983, pp 151-156.

4. D. B. Gustavson and John Theus, "Wire-OR Logic on Transmission Lines", IEEE MICRO, June 1983, pp 51-55.

5. A. Cojan, L. Pregernig, J. C. Tarle, and R. Downing, "The FASTBUS Cable Segment", paper presented at this symposium.