SLAC-PUB-2991
November 1982
(I/A)

PROGRESS ON THE SLAC SNOOP DIAGNOSTIC MODULE FOR FASTBUS*

Helmut V. Walz and D. B. Gustavson
Stanford Linear Accelerator Center
Stanford University, Stanford, California 94305
and
Robert Downing
University of Illinois, Urbana, Illinois 61801

Abstract

A SNOOP Diagnostic Module for FASTBUS is under development at SLAC. The SNOOP Module resides on a FASTBUS crate segment and provides diagnostic monitoring and testing capability. It consists of a high-speed ECL front-end to monitor and single-step segment operations, a simple master interface, and a control processor with two serial communication ports. Module features and specifications are summarized, and prototype hardware is shown.

I. Introduction

Data acquisition and control systems based on the proposed FASTBUS Specifications[1] are designed to permit monitoring and control for diagnostic purposes by means of a SNOOP Module which can observe and delay bus operations (Fig. 1). The SNOOP Module has access to all si;nals involved in operations, since they are bussed to all module positions. Thus a history can be kept of all operations on a segment, no matter what the locations of master and slave modules may be relative to the SNOOP Module.

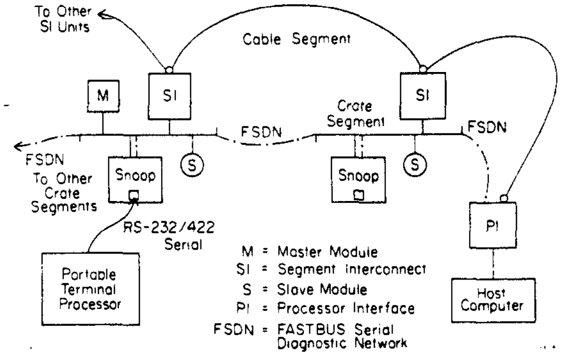

Fig. 1. Typical FASTBUS system with SNOOP Modules.

The SNOOP Module can delay operations by asserting the Wait signal. This inhibits transitions of all handshake timing lines so that all bus signals involved in the current bus cycle remain static until Wait is removed. The system can thus be single-stepped by asserting Wait. Similarly, comparison logic in a SNOOP Module can be used to detect a particular address or data transfer, and then assert Wait to halt the segment.

The serial bus lines in each crate segment may be used as an independent communication path between diagnostic modules and system host computers, when connected to a serial network which bypasses segment interconnect units.

A prototype SNOOP Module has been developed with a programmable Wait generation logic, address and data traps, a history silo memory and interface logic for master and slave operations. These functions are implemented with high-speed 100 K ECL parts to optimize response capability. Control and supervision of the fast front-end section are handled by a compact microprocessor section based on a powerful 16-bit CPU (MC 68000). The processor section includes interface ports to the FASTBUS serial diagnostic network, and to a general-purpose, UART-type serial connection for a terminal which provides floppy disk storage for diagnostic programs and data.

The proposed design for this module and supporting software have been described in previous papers.[2,3] The present protytype differs from the original design proposal in circuit board size, segment connection pin allocation, MS and SS codes, increased history silo memory word width, and an upgraded processor section. Table I contains a summary of specifications. The changes in the 100K ECL front-end section are primarily the result of changes in the tentative FASTBUS specifications, while the new processor section provides a threefold increase in on-board memory and almost twice the execution speed by capitalizing on latest microprocessor system chip technology.

Table I

Features and Basic Specifications

- Fast front-end implemented with 100 K ECL devices in 24-pin flatpaks.
- Programmable wait-step logic with 5 ns response time.
- Address and address-data combination traps with 8 ns response time.
- Parity-error trap with 15 ns response time.
- Activity history silo with 56-bit × 256-word RAM; 100 MHz recording speed; programmable recording modes; FB or real-time synchronized; logic analyzer mode with internal or external clock source.
- Simple master and slave capabilities with geographical address recognition, bus arbitration, IO register programmed protocol control, and software emulation of CSR registers.
- MC 68000 CPU (12 Mhz).
- 25 level programmable priority interrupt structure (Z8530A and Z8536A peripherals).
- 48K-word × 16-bit memory with variable configuration of RAM/ROM sizes.
- 24K-word × 16-bit maximum size static RAM (100 ns access and cycle time).
- 32K-word × 16-bit maximum size EPROM (250 ns).
- 3-channel counter-timer with programmable inputs and modes (Z8536A).
- Dual channel serial Communication Controller peripheral (Z8530A) with baud rate generators.
- RS232/422 asynchronous serial port with selectable baud rate (50-19.2K baud).
- Synchronous Serial interface for FSDN port (300K baud).
- Processor driven front panel with status display, manual WAIT execution switch, interrupt sense switch for processor selftest, CPU reset switch; NIM level test outputs from wait-step logic, and external clock input for silo memory recording.
- One unit wide FB module type A with 210 IC packages on a 6-layer PC board; estimated 85 watt power dissipation.

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

(Presented to the Nuclear Science Symposium, Washington, D.C., October 20-22, 1982.)

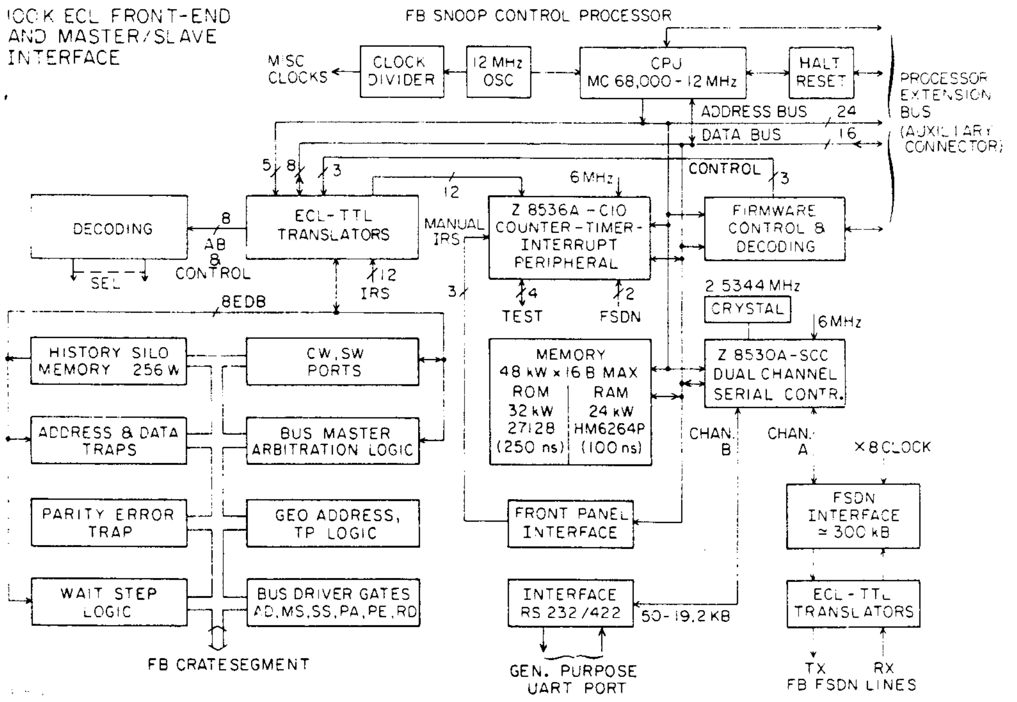

Fig. 2. SNOOP Module Block Diagram

II. Control Processor Section

Figure 2 shows a block diagram of the prototype SNOOP Module. The processor section is based on the MC 68000 (12 MHz) CPU in a leadless chip carrier package. Total on-board memory size is 48 K words (16-bit), when 32 KW EPROM and 16 KW static RAM are used. Patch options on the pc board permit other configurations (16 K ROM - 24 K RAM, for example), and use of 64 K- or 128 K-bit EPROMs. A memory mapping switch under CPU I/O control is available to substitute the on-board ROM with RAM memory connected to the processor extension bus for software development and testing. A multifunction peripheral circuit (Z8536A CIO) handles twelve interrupt lines from the front-end section, and three switch interrupts from the front panel. It also provides three counter-timer functions. Severl interrupt inputs may also be used at the same time as counter-timer inputs. Therefore counting and timing measurements related to segment activity are possible. One counter-timer function is dedicated to the FASTBUS Serial Diagnostic Network (FSDN) interface.

A dual channel serial communication controller peripheral (Z8530A - SCC) implements two serial ports. One is used as a general-purpose UART port, which is programmable with standard rates from 50 baud to 19.2 K baud. By means of jumpers the serial interface standard (RS232/422) and modem control signals are selected.

The other port is used as the FSDN port, which operates with a synchronous transmission rate of approximately 300 K bits per sec. The SCC channel, operating in SDLC mode, takes care of network address recognition, cyclic redundancy code generation and checking, and interrupt handling. A network interface consisting of two finite-state logic sequencers, a sampling register, and ECL-TTL translators handles transmit data encoding, receive data and clock recovery, and collision detection. The CIO and SCC peripherals are connected with the Zilog-type daisy chain interrupt lines forming a vectored interrupt structure. In addition the CPU can accept other interrupt inputs from the processor extension bus. All input-output, memory, interrupt, and handshake control is implemented in firmware with programmable logic arrays and sequencers.

III. Prototype Development Progress

A pc prototype board with the ECL front-end section of the SNOOP Module has been fabricated and is being tested. The high-density six-layer pc board is shown in Figure 3. The top area of the board with several temporary spare DIP patterns and two connectors for flat cables is reserved for the processor secticn.

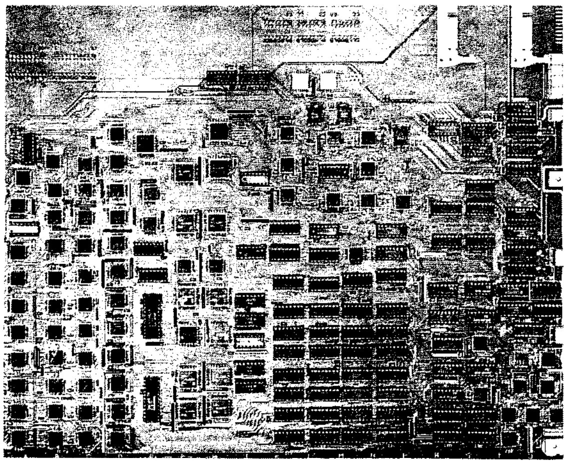

Fig. 3. ECL Front-End section prototype board. 4396A3

The processor will have 33 I.C. packages. Three wirewrap prototype modules of the first SNOOP processor design[2] have been built and tested and used for serial network development. A wirewrap prototype of the new processor configuration described above is expected to be fabricated in December 1982. A complete SNOOP Module pc prototype including the processor section is expected in February 1983. In the meantime testing of the SNOOP Module will proceed by connecting one of the wirewrap processors via the two temporary flat cable connectors to the ECL front-end prototype.

The FASTBUS Serial Diagnostic Network (FDSN) will allow the coordination of several SNOOP Modules for solving multisegment problems in a FASTBUS system. It also allows connection of a portable smart terminal with disk drive and optional printer to the network via the asynchronous serial port of the SNOOP Module at different points in the system. The network protocol is presently a prototype implementation and the interface utilizes four IC packages. However we would like to adopt one of the industry standard protocols such as Ethernet, onceVLSI support chips will be available to reduce the hardware and board space required significantly. Gateway nodes may be used in the meantime to connect the FSDN prototype network to an industry standard network.

Data rate requirements for FSDN are limited by the microporcessor to several hundred kilobaud, in order for it to simultaneously handle network messages and the realtime control of the SNOOP Module. The operating system software for the SNOOP Module and its supporting terminal is presently being rebuilt based on experience gained with the wirewrap prototype processors.

IV. Summary

The development of the first prototype of a FASTBUS SNOOP Module is expected to be completed in the next several months. This effort has been a tedious and lengthly one. This was not the result of complicated module details or FASTBUS requirements. It was primarily the result of some major changes in the FASTBUS specifications such as board configuration and signal pin assignments, and the process of gaining experience in the printed circuit layout of FASTBUS-size boards with high density packaging based on integrated circuits in flat packages. This requires pc layout design at a 4:1 scale, and we are at present not equipped with work stations and plotters to accommodate such sizes. A Computer Automated Design (CAD) system should relieve some of these problems in the future.

Acknowledgments

The authors would like to acknowledge the firm support and encouragement for this project by R. S. Larsen, and the software contributions by T. Holmes.

References

1. FASTBUS Tentative Specification, U.S. NIM Committee, available from L. Costrell, NBS, Washington, D.C.
2. FASTBUS SNOOP Diagnostic Module, Helmut V. Walz, Stanford Linear Accelerator Center, and Robert Downing, University of Illinois, IEEE Trans. on Nucl. Sci., NS-28, No. 1, (Feb. 1981), pp 380-384.
3. A. "Front Panel" Human Interface for FASTBUS, David B. Gustavson, Terry L. Holmes, Leo Paffrath, and John P. Steffani, Stanford Linear Accelerator Center, IEEE Trans. on Nucl. Sci., NS-28 No. 1, (Feb. 1981), pp 343-345.