A mining research contract report
JUNE 30, 1984

UNDERGROUND MINE MONITORING & CONTROL TESTING CRITERIA

Contract J0113057
West Viıginia University

Bureau of Mines
United States Department of the Interior
Mining Health and Safety Technology

OFR
85-51

DISCLAIMER

(this page replaces old Title Page)

REPORT DOCUMENTATION PAGE

1. Report No.	2.	3. Recipient's Accession No.
4. Title and Subtitle Underground Mine Monitoring & Control Testing Criteria		5. Report Date JUNE ~~September~~ 1984 6.
7. Author(s) R.S. Nutter, R.S. Chapman, A.M. Kissel, C.A. Winkelmann, W.R. Voltz		8. Performing Organization Report No.
9. Performing Organization Name and Address West Virginia University Dept of Electrical Engineering P.O. Box 6101 Morgantown, WV 26506-6101		10. Project/Task/Work Unit No. 11. Contract or Grant No. J0113057
12. Sponsoring Organization Name and Address Assistant Director--Mining Bureau of Mines Dept of the Interior Washington, DC 20241		13. Type of Report Final Report for 7/30/81 - 9/29/84 14.
15. Supplementary Notes		

16. Abstract

The proliferation of computer-based monitoring and control systems for underground coal mines has created the need for criteria to be used in evaluating the performance of these systems. This report contains the findings and conclusions of West Virginia University Department of Electrical Engineering's research on the development of test criteria for monitoring and control systems. The design of a test facility for control systems performance measurement of monitoring and control systems is presented Tests were conducted and the results were used to develop recommended criteria. The implementation of the test facility, test conclusions, and resultant objectives are described in the executive summary. The appendix presents the detailed design of test facility hardware and software.

17. Originator's Key Words mine monitoring, reliability, criteria, computer-based testing, test facility,		18. Availability Statement Release Unlimited	
19. U. S. Security Classif. of the Report unclassified	20. U. S. Security Classif. of This Page unclassified	21. No. of Pages 155	22. Price

FOREWARD

This report was prepared by West Virginia University, Department of Electrical Engineering, Morgantown, West Virginia, under USBM Contract No. J0113057. The contract was initiated under the Coal Mine Health and Safety Research Program. It was administered under the technical direction of the Pittsburgh Mining and Safety Research Center with Mr. Ray Kacmar and Mr. Ed Fries acting as the Technical Project officers. Mr. Doyne Teets was the contract administrator for the Bureau of Mines.

This report is a summary of the work recently completed as part of this contract during the period July 30, 1981 through June 30, 1984. A draft of this report was submitted on July 1, 1984.

*Reference to specific brands, equipment, or trade names in this report is made to facilitate understanding and does not imply endorsement by the Bureau of Mines.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 ABSTRACT

The proliferation of computer-based monitoring and control systems for underground coal mines has created the need for criteria to be used in evaluating the performance of these systems. This report contains the findings and conclusions of West Virginia University, Department of Electrical Engineering's research on the development of testing criteria for monitoring and control systems. The design of a test facility for control systems performance measurement of monitoring and control systems is presented. Tests were conducted and the results were used to develop recommended criteria. The executive summary of this report describes the research objectives, implementation of the test facility, test conclusions, and resultant criteria. The Appendix presents the detailed design of test facility hardware and software.

The work was performed for the U.S. Bureau of Mines under Contact No. J0113057.

2.0 Executive Summary

2.1 Introduction

The objective of this research was to develop a monitoring/control system testing tool. This tool would not only test existing monitoring/control systems that were proposed for use underground but would also be flexible and expandible enough to test future systems that will exist, for example, in the next ten years. This tool will provide a procedure for evaluating the capabilities of these underground monitoring/control systems. This development included architecture, algorithms, hardware, and software for the prototype test fixture.

The system provides the capability of performing evaluations, including the communication saturation point of monitoring/control systems. This saturation point is extremely important, especially during emergency conditions underground, to evaluate delay-time from alarm condition to alarm annunciation. The system also provides the capability of accelerated system testing with regard to the software that normally operates a modern system. The only realistic method for testing monitoring/control system software was by using a test fixture such as this one.

2.2 Test Fixture System Overview

The purpose of the test fixture is to provide a method of testing monitoring equipment hardware and software without requiring actual installation in an underground coal mine. By using simulation, rather than field testing, it is possible to accelerate this testing. All of the software in the monitoring system (the target system) can be exercised, including the software modules normally executed only during emergency situations. The test fixture also provides for consistent testing of different target systems, by allowing strict control of the test procedure and 'mine' parameters.

Analysis of currently available coal mine monitoring systems showed that they could generally be described as central computer based telemetry systems [1]. Common configurations consist of one or more remote outstations connected to analog sensors or alarm contact closures. Implementation of outstations varies widely, from 'dumb' multiplexors, to sophisticated processor-based data gathering devices. These outstations report the status of the sensors through a communications system (usually multiplexed) to a centralized computer where the data is analyzed for alarms and reported to the operator. Again, the actual hardware designs vary widely, but two consistent points were observed. All of the systems connect to sensors and all of the systems report alarms.

Based on this analysis, the design of a test fixture for testing monitoring systems was performed. The test fixture is intended to

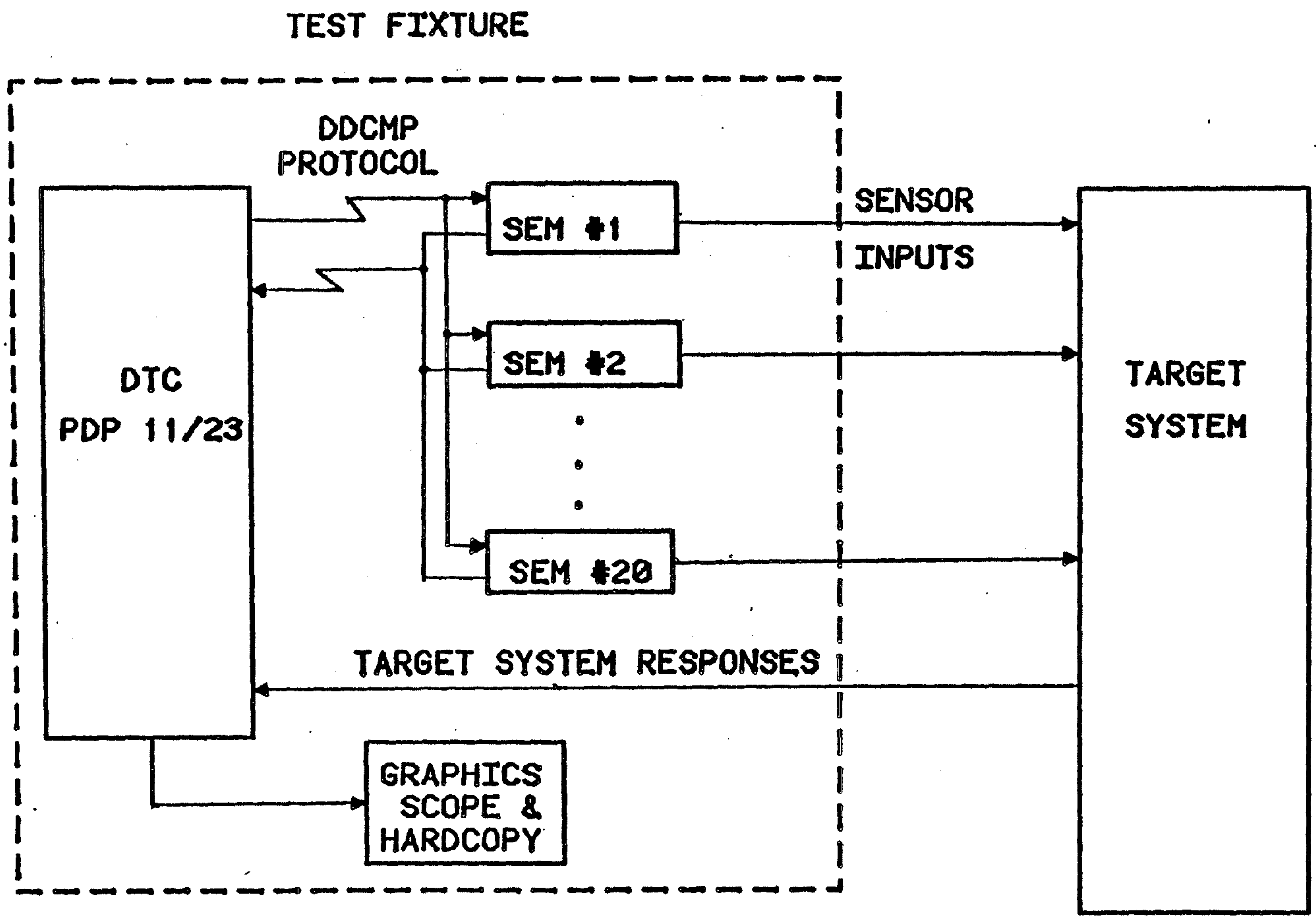
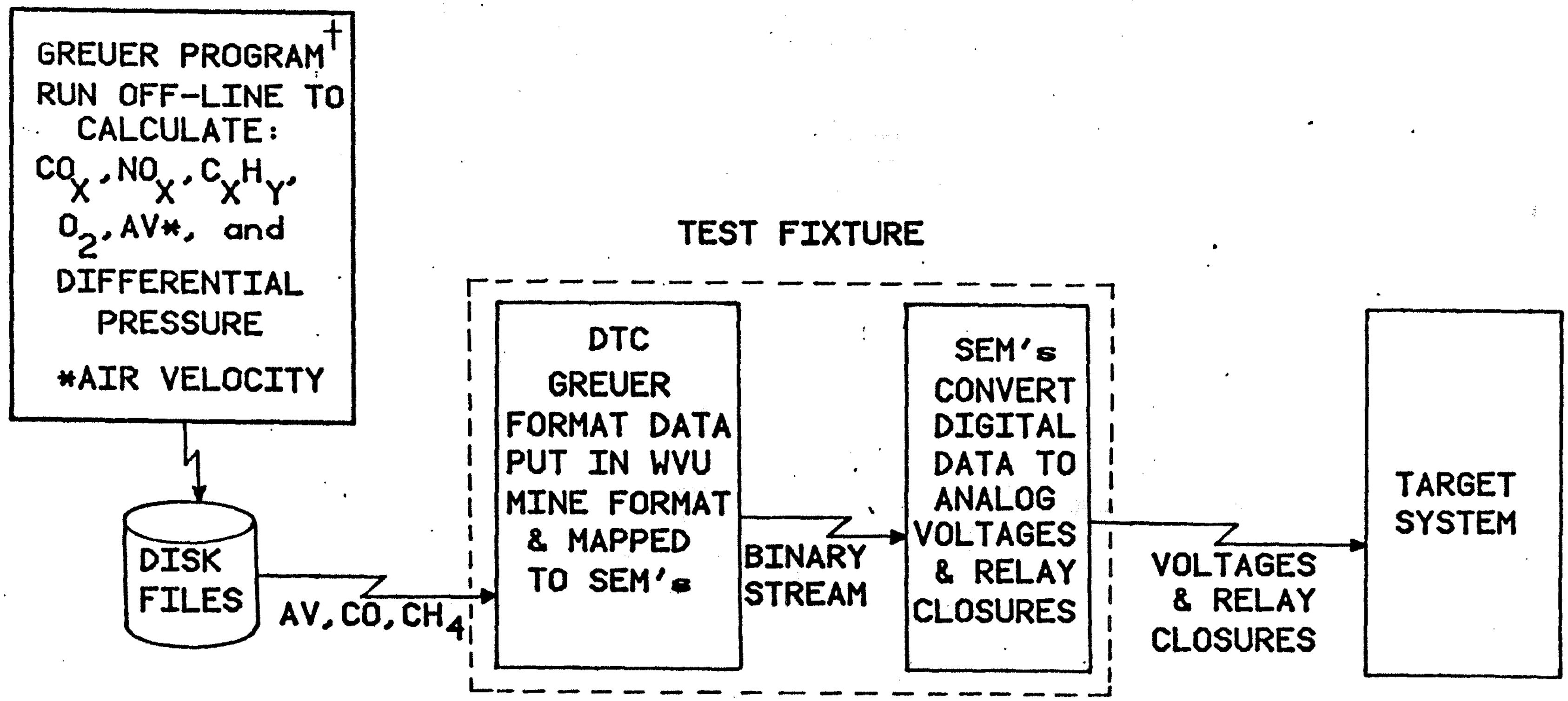


Fig. 2.1.1 Block Diagram of the Test Fixture



† Can be run on any system supporting FORTRAN

Fig. 2.1.2 Data Transfer for the Test Fixture

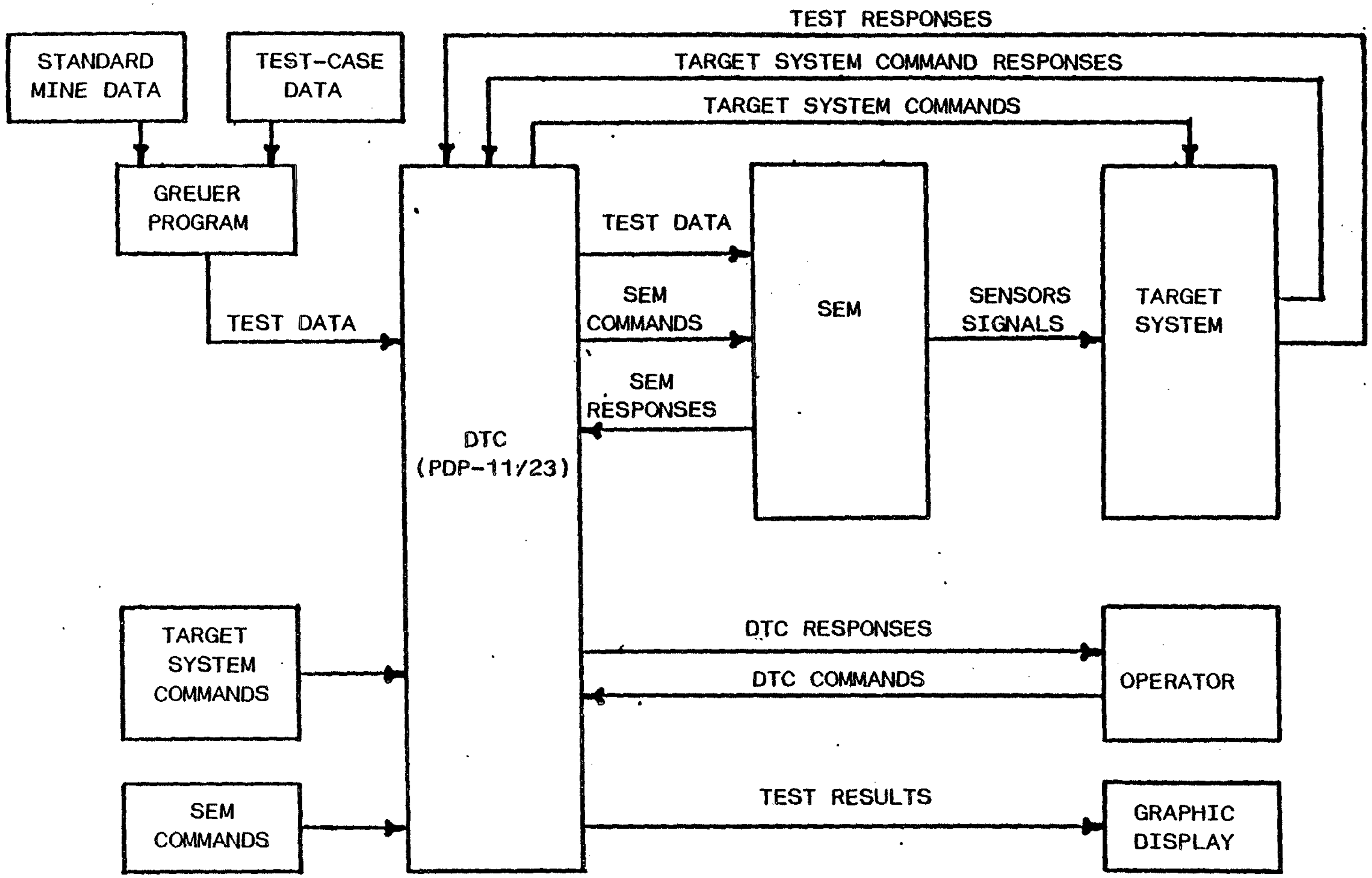


Fig. 2.1.3 The Logical Function System Block Diagram of the Test Fixture

meet the criteria for comprehensive testing of monitoring systems hardware and software. It was decided that a reasonable test facility would be able to provide a large number of simultaneously varying stimuli to the system under test. Figure 2.1.1 shows a block diagram of the test fixture and Figures 2.1.2 and 2.1.3 show the data transfer and the logical function system block diagram for the test fixture. The test fixture interfaces to the target system outstations by replacing the sensors usually associated with the monitoring system. The interface acts as a Sensor Emulator (SEM), providing data to the target system in either analog or digital form, depending on the type of outstation inputs specified. These outstation interfaces are controlled and synchronized by a central processor unit called the Dynamic Test Controller (DTC). The DTC interfaces to the target system's alarm reporting device (usually a serial terminal or printer, but possibly an annunciator panel) to record alarm responses to the SEM data provided. The DTC also generates the raw SEM data and control messages. The data and messages are communicated during the test via a double twisted-pair cable in full duplex operation. The protocol is serial and asynchronous using the Digital Data Communications Message Protocol (DDCMP).[2]

2.2.1 HARDWARE

Dynamic Test Controller

The DTC hardware consists of a Digital Equipment Corporation (DEC) LSI-11/23 computer system configured with an LSI-11/23 CPU, 128K words of memory, floating point processor, dual RL02 10 Megabyte disk drives with removable disk packs and controller, four RS232 serial line interfaces, one 16-bit parallel interface, one VT100 CRT terminal, one Tektronix 4025 graphics terminal, and one Tektronix 4662 multi-pen plotter. The system includes cabinet, power supplies and cables for interfacing to the terminal equipment.

The configuration of the DTC processor is given in the Appendix of this report. The communications link and the Tektronix plotter operate at 1200 baud. The baud rate of the target system's console must be matched to the serial line used to record target system responses. The baud rate varies according to the system being tested, and may require modification of a DTC serial interface. Jumper selectable baud rate options are given in the DL11-J hardware reference manual and included in the Appendix.

Sensor Emulator

The SEM consists of a processor circuit board and an input/output circuit board. Both boards were designed by project personnel after determining that no ready-to-use processor boards could be obtained that would meet the special requirements of the SEM. The processor board utilizes an 8085 cpu, 16K bytes of read-only-memory, 4K bytes of read-write memory, a serial communications interface, and timing and control circuitry. The input/output board provides four analog outputs for modeling up to four coal mine parameter sensors and four analog-to-digital inputs, for verification of the analog outputs. Most of the monitoring systems available provide analog sensor interfaces; however, to be able to test a target system that monitors contact closures, the input/output board also provides 16 relay contact outputs and a 16-bit input port.

Communications

Also included is a special communications interface that allows multiple SEMs to be accessed with a common communications bus. This interface connects to a 1200 baud serial line and a serial interface on the 11/23 system. The serial line is used for full duplex communications of data and commands from the DTC to the SEM's. Test synchronization is provided by a dedicated 'SYNC' signal sent over the communications cable. This signal is generated by the communications interface on command from the 11/23 over its parallel I/O port. The interface also generates a 'TICK' interrupt at 2ms (500 HZ) intervals to the 11/23, which is used as a clock to 'time-tag' target system alarm messages as they are received.

2.2.2 SOFTWARE

Dynamic Test Controller

The operator is able to interact with the test software to select menu driven options for the test, react to DTC or SEM faults, get information on status of test, and instruction on test fixture operation. The operator can also view data (tabular and graphic) before generating hard copies of the data. Software provides ease of use so that expertise in computers is not needed. However, it must be assumed that the operator will have knowledge of the functions of the monitoring system under test.

Most of the DTC software is written in the high-level languages FORTRAN, Pascal, or 'C'. The real-time multitasking operating system RSX-11M was chosen as the nucleus of the DTC software.[3] The major components of the DTC software are the test data generator, SEM interface, communications control, target system alarm device interface, operator interface, and data analysis modules. The software modules are shown in Figure 2.1.4.

Test Data Generator

The standard fire model provides a single small fire source. The Greuer computer program calculates the propagation of products of combustion throughout the ventilation network. The DTC software 'maps' these values into the target system configuration being tested.

The exerciser data generating algorithm uses a set of fixed equations to generate alarm conditions to exercise the target system hardware and software. The purpose of the exerciser data generator is to provide a method of describing the alarm rate function that will be used in testing the target system.

SEM Interface

The SEM's interface procedures provide the necessary sequences of commands to set up SEM test operation. This software provides the proper commands for the test desired and assures that the data is available. Error recovery is provided through SEM fault messages allowing operator action.

The SEM's require commands to set up the test operation and data for output to the target system. The software provides the proper commands for the test desired and assures that the data is available. The software also reacts to SEM fault messages (if any) through operator notification.

The DTC assigns a mine simulation data point to an SEM output. The valid sensor types are air velocity, temperature, carbon monoxide, carbon dioxide, oxygen, and methane. Each sensor point corresponds to an output of the test simulation scenario. For standard fire, air velocity is assigned to a branch in the mine ventilation network, temperature and gas concentrations are assigned to junctions in the mine ventilation network.

Communications Control

Communications between the DTC and the remote SEM's is accomplished using Digital Equipment Corporation's Digital Data Communications Management Protocol (DDCMP).[2] This software was written in the 'C' programming language for the DTC and in PL/M for the SEM.

Target System Alarm Collection

Since the target system alarm device specifics are not known, the software to interface to the reporting device is designed to be easily modified. This software collects target system output and saves it on the DTC data disk (time-tagged) for analysis.

Operator Interface

The operator interface modules of the DTC allow the operator to configure the hardware and software for the specific characteristics of the target system while maintaining a standard test so that equivalence from system to system is achieved. The test setup software provides on-line help for connecting the sensor emulators to the target system, setting up the SEM communications trunk, and connecting the target system console to the DTC. The operator is able to define the target system console message formats so that the DTC can discard unwanted messages. The operator selects a test algorithm to be followed, runs system diagnostics, verifies SEM operation, and initializes the data base for the test. Operator intervention during the test is not desirable, except for extreme circumstances that require some error recovery action to prevent invalidating the test data. The operator is provided continuous feedback during the test. The status of various test parameters are shown in a continuously updated console display. This display contains such information as elapsed time, time remaining, number of SEM points in alarm, time of day, date, etc.

The operator is able to interact with the test software to select menu driven options for the test, react to DTC or SEM faults, get information on status of test, and help in operating the test fixture. The operator can view data (tabular and graphic) before generating hard copies of the data. (Software provides easy-to-use operator interface so that expertise in computers is not needed.) It can be assumed that the operator has some knowledge of the functions of the monitoring system under test. The software provides a comprehensive log of system activity without unnecessarily inundating the operator with output.

Data Analysis

The data analysis software of the DTC provides options for selecting data to be displayed and allows the operator to select displays without being inundated with output. This software generates tabular data (ie. reports) neatly formatted for a printer and allows the operator to view data on the terminal before printing hard copies. Graphics representation of the data on both the hard copy plotter and the graphics terminal is provided.

The DTC test software written by West Virginia University is included with the system. This software package includes finished test software in ready-to-run form and documentation on the use of the test software. Also included is a validation package to verify the correct operation of the DTC. The test software includes Digital Equipment Corporation's RSX11-M operating system and the necessary licenses to use this operating system. Other licensed products used in the development of the software include DEC Fortran IV Compiler, Tektronix Plot10 Graphics Package, and Whitesmith's Portable 'C' Compiler with its companion Pascal Compiler.

Sensor Emulator

The heart of the SEM is its software package. Key elements of this package include: diagnostics, test synchronization, terminal and remote communications, and SEM verification processes. Nearly all of the SEM software is written in the high-level language PL/M for ease of modification and understanding.

The diagnostic programs consist of routines to test the 8085 instruction set, the memory, input/output, and timing and control hardware. The routines are executed upon power-on reset or through operator commands. In order for an SEM to function in a test, all of the diagnostics must be executed without any errors being detected. In addition, during the normal operation of the SEM software, a hardware 'watch-dog' timer monitors execution to detect run-time errors.

Communications

Data and commands are received by the SEM from the DTC through the communications procedures. Communications uses the DDCMP protocol with each SEM acting as a "remote" station connected to the communciations link. As data and commands are received they are stored in the SEM's internal buffers until they are needed. The SEM can also send messages to the DTC using this protocol in the event of errors or in response to an operator command.

The communications software provides a full-duplex data link, as shown in Figure 2.1.4. These software modules of the DTC provide a DDCMP control station for link traffic synchronization, error recovery, and message sequencing.

2.3 RECOMMENDED CRITERIA

Currently, the Mine Monitoring Test Facility (MMTF) is configured to generate two different types of tests: a square wave test and a pulse test. The square wave test is used to determine the maximum alarm frequency and stress the software of a SUT. The pulse test is used to determine the scan time, as well as stress the software of a SUT.

2.3.1 THE SQUARE WAVE TEST

The square wave test produces a square wave signal at the input of one point on the SUT. The square wave causes the SUT to produce a series of alarm and reset reports, while all other points are held in their reset state (see Figure 2.3.1).

Square wave tests are performed on a single point, while the frequency of the square wave is changed from low to high. The low-frequency tests give the SUT time to respond properly to the incoming signal. The frequency of subsequent square wave tests is increased until the maximum alarm frequency of the SUT is found. A few more tests are then run with frequencies greater than the maximum alarm frequency to complete this testing sequence.

The maximum alarm frequency is the frequency at which the system under test fails to report an alarm condition. Such a failure could be caused by at least three conditions: the alarm condition was too

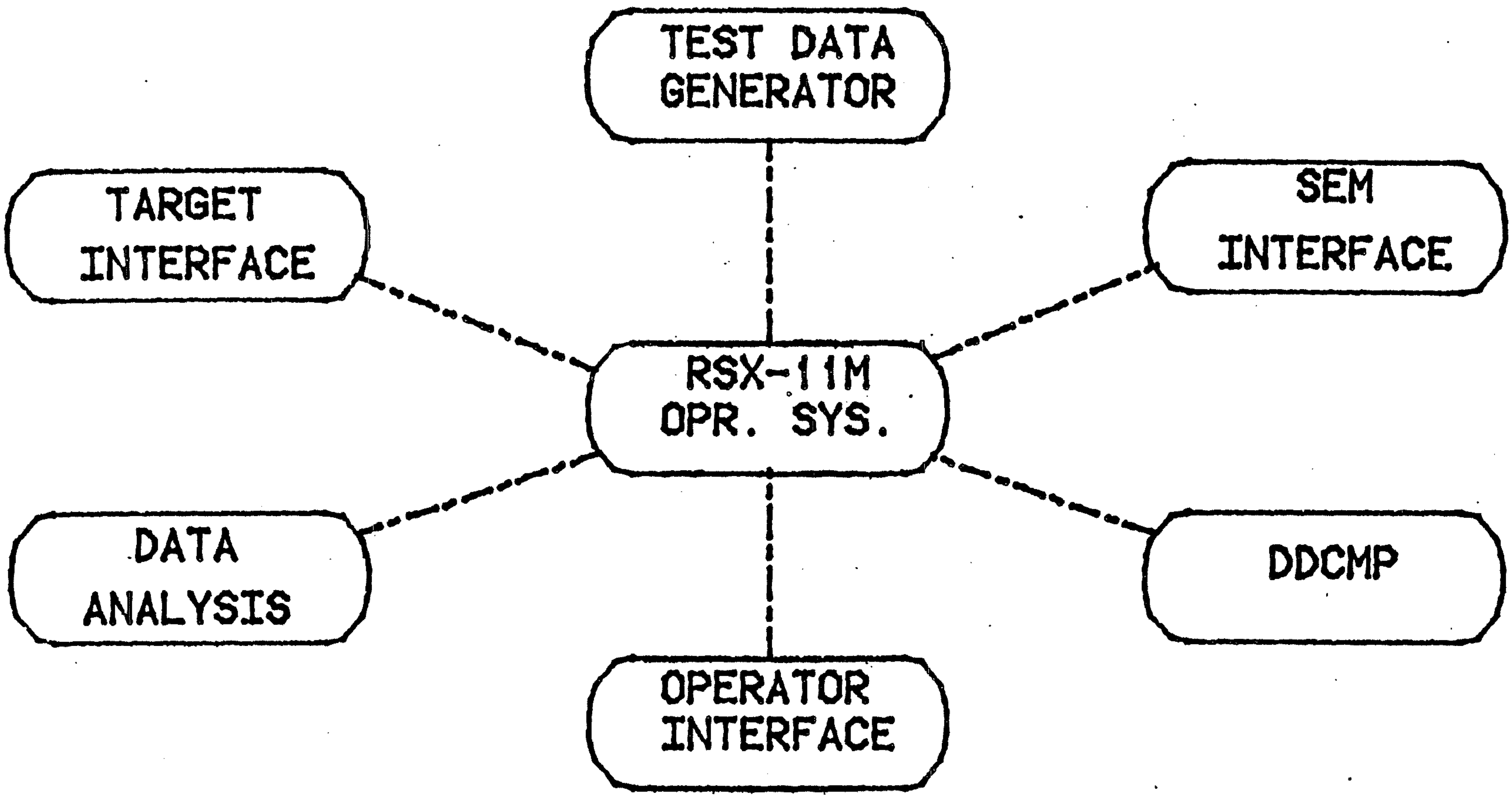


Fig. 2.1.4 DTC Software Modules

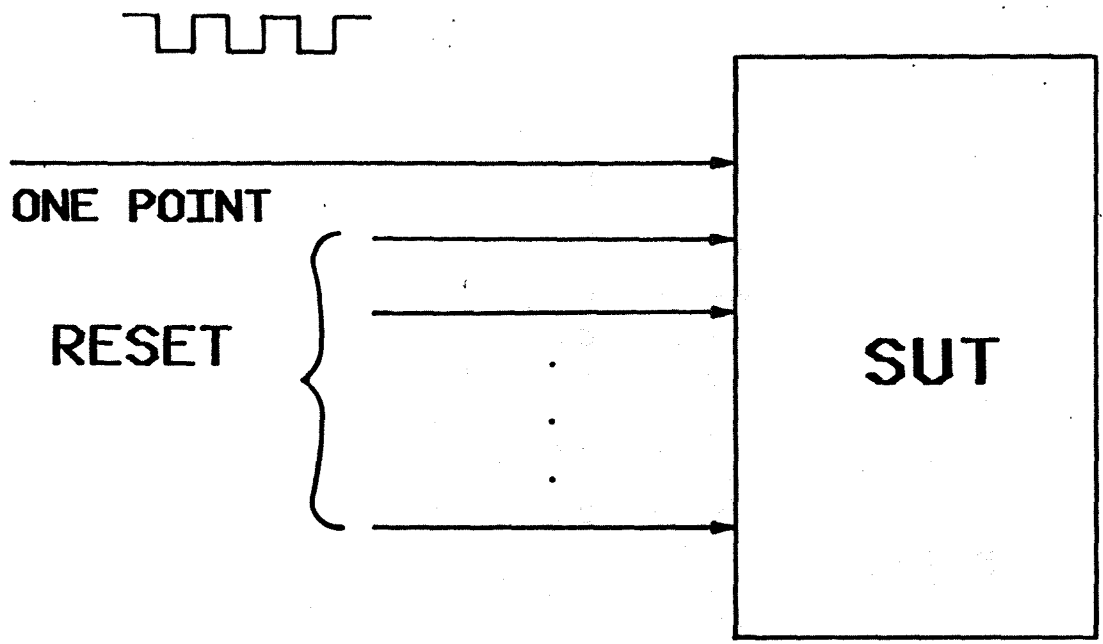


Fig. 2.3.1 Square Wave Test

brief for the SUT to detect, the SUT was busy performing other operations and never detected the alarm condition, or the SUT could no longer keep up with the number of alarm conditions. The first and second conditions are attributed to the time required for either the SUT to scan all of its inputs (known as the scan time) or for its hardware to respond to the input signal. The third condition is attributed to the SUT's memory capability. All of these conditions assume that the SUT works properly and has no obvious software or hardware deficiencies. If deficiencies exist, then they too may cause the SUT to fail.

NFPA Guidelines for Single Point Alarms

The National Fire Protection Association has established rules for system time response of proprietary protective signaling systems for fire protection in buildings and publishes them in their annual National Fire Code publication. The 1981 Fire Code (the latest publication) states that

> "The maximum allowable lapse, from sensing a fire alarm at an initiating device or initiating device circuit until it is recorded or displayed at the central supervising station, shall not exceed 90 seconds.[4]

This rule is for a single status change at any instant in time, as opposed to multiple simultaneous status changes. The rule also implies that no signals can be lost (not reported). It is recognized that no such recommendation is given in the Code of Federal Regulations, Title 30, for mining. It's noted here, however, that such an evaluation and comparison should be helpful for mine-related monitoring evaluation as well.

When the Data Analysis Software Package (DASP) performs the analysis on each square wave test, it checks for two conditions: did the SUT report all alarm conditions and were the alarm conditions reported within 90 seconds after the initiation of the alarm signal. The first condition has a count associated with it that's known as the missed alarm count. The second condition is known as the timing aspect of the Fire Code. If the test has not missed alarm conditions and passes the timing aspect of the Fire Code, then the test has passed the Fire Code. While performing the two checks, the DASP accumulates the total lapsed time of the alarm reports and from this data calculates the average alarm report time delay to be used in each of the graphs.

The reset reports are currently ignored by the data analysis program for two reasons: the Fire Code doesn't specifically include reset conditions in the system response time and the proposal specification to the USBM concerned itself with alarm annunciation only. The authors believe that reset annunciation is not as crucial as alarm annunciation but that both should be included in a mine monitoring

2.3.2 THE PULSE TEST

The pulse test produces a single rectangular pulse signal at the input of all points on the SUT simultaneously (see Figure 2.3.2). The pulse consists of an alarm level signal that can be varied in time from short to long. At all other times during the test, the Mine Monitoring Testing Facility (MMTF) produces a reset level signal.

Analogous to the square wave test procedure, the first few tests use a large pulse width to allow the SUT ample time to respond to all incoming signals. The subsequent tests use decreasing pulse widths until the scan time of the SUT is found. A few more tests are then run with even smaller pulse widths to the point where the SUT fails to respond to the pulse or the MMTF reaches its own internal 2-millisecond pulse width limit.

The scan time is the minimum time required by the SUT to scan all of its inputs for a status change. If the scan time of the SUT is larger than the pulse width of the pulse test, then the SUT will not detect the status changes of all of the inputs.

Failure to report all alarm conditions during a pulse test could be attributed to four conditions: the alarm conditions were too brief for the SUT to detect, the SUT was busy performing other operations and never detected the alarm conditions, the SUT couldn't keep track of the number of alarm conditions received, or the SUT couldn't complete the scan of all the points before the pulse went away. Again, all four conditions assume that the SUT works properly and has no obvious software or hardware deficiencies.

NFPA Guidelines

The National Fire Code states that

> "To facilitate the prompt receipt of fire alarm signals if multiple simultaneous status changes of any type occur when the system covers other types of signals, the requirement of either of the following shall be met: a) In addition to the maximum processing time for a single alarm, the system shall record simultaneous status changes at a rate not slower than a quantity of 50, or 10 percent of the total number of initiating device circuits connected (whichever is smaller) within 90 seconds without loss of any signal. b) In addition to the maximum processing time, the system shall display or record fire alarm signals at a rate not slower than one every 10 seconds when any number of status changes occur at any rate without loss of any signals."[4]

This rule is interpreted as allowing each point 90 seconds to report for a system with 10 points or less, 10 seconds to report each alarm

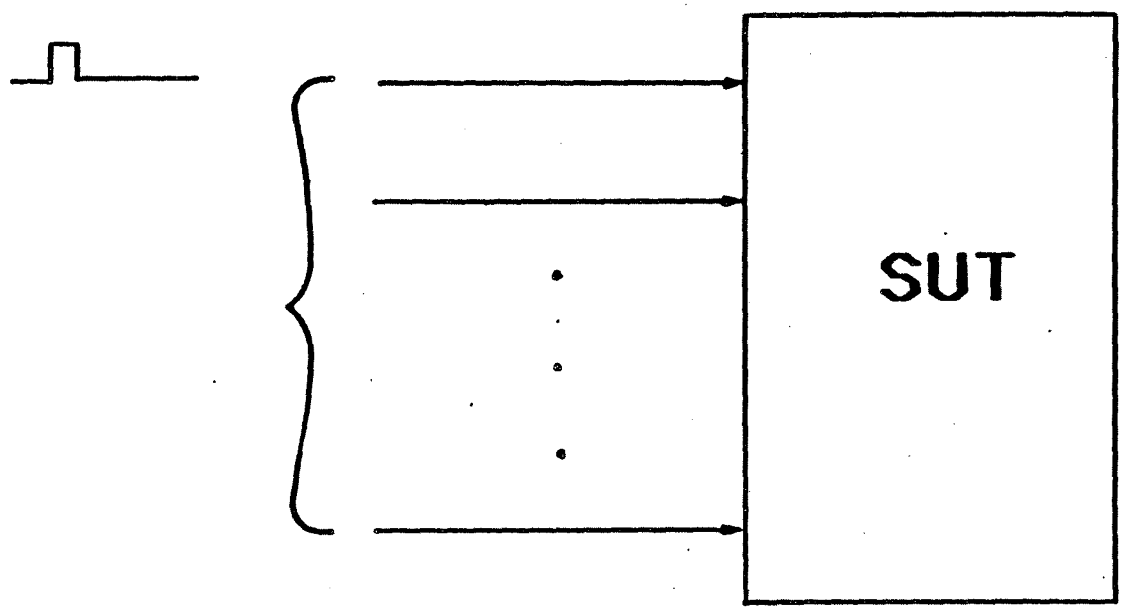


Fig. 2.3.2 Pulse Test

for a system with 90 points or more, and a calculated proportional value between 10 seconds and 90 seconds for a point count between 10 and 90.

When the MMTF performs the analysis on each pulse test, it first calculates the Fire Code lapse time between simultaneous reports and then checks for two conditions: did the SUT report all alarm conditions and were the alarm conditions reported within the calculated Fire Code lapse time. The first condition has a count associated with it that's known as the missed alarm count. The second condition is known as the timing aspect of the Fire Code. If the test has no missed alarm conditions and passes the timing aspect of the Fire Code, then the test has passed the Fire Code. While performing the checks, the program accumulates the total lapsed time of the alarm reports and from that data calculates the average alarm report time delay. Like the square wave test the reset condition reports are ignored. It is the opinion of the authors that the 1981 NFPA Codes, as referenced above, are entirely reasonable criteria for underground mines.

2.4 TEST CONCLUSIONS

Four different types of graphs are generated from the test results:

1. Graph 1: The percentage of missed alarms versus the square wave frequency; generated from the square wave tests.

2. Graph 2: The percentage of missed alarms versus the pulse width; generated from the pulse tests.

3. Graph 3: The average alarm time delay versus the square wave frequency; generated from the square wave tests.

4. Graph 4: The average alarm time delay versus the pulse width; generated from the pulse tests.

Each graph shows a different aspect of the system under test (also known as the SUT).

Graph 1 shows the maximum alarm frequency of the SUT (see Figure 2.3a). The maximum alarm frequency is the square wave frequency at which the SUT fails to report an alarm condition. For low frequencies, the percentage of missed alarms is low, but as the frequency increases the percentage quickly rises. The maximum frequency for this particular SUT is around 0.25Hz or one alarm every 4 seconds. (Only one point on the entire SUT is used in square wave tests.)

Graph 2 shows the scan time of the SUT (see Figure 2.3b). The scan time is the minimum time required by the SUT to scan all of its inputs for a status change. If the scan time is larger than the

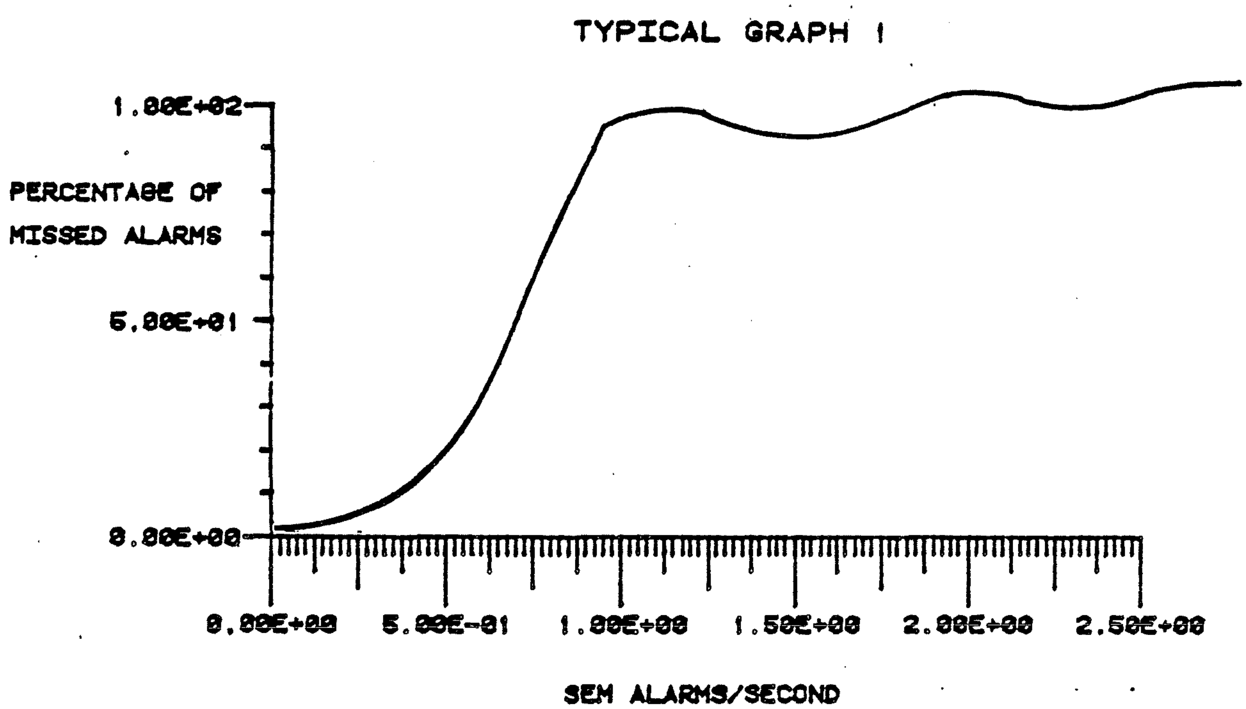


Fig. 2.4a Percentage of Missed Alarms vs Square Wave Frequency

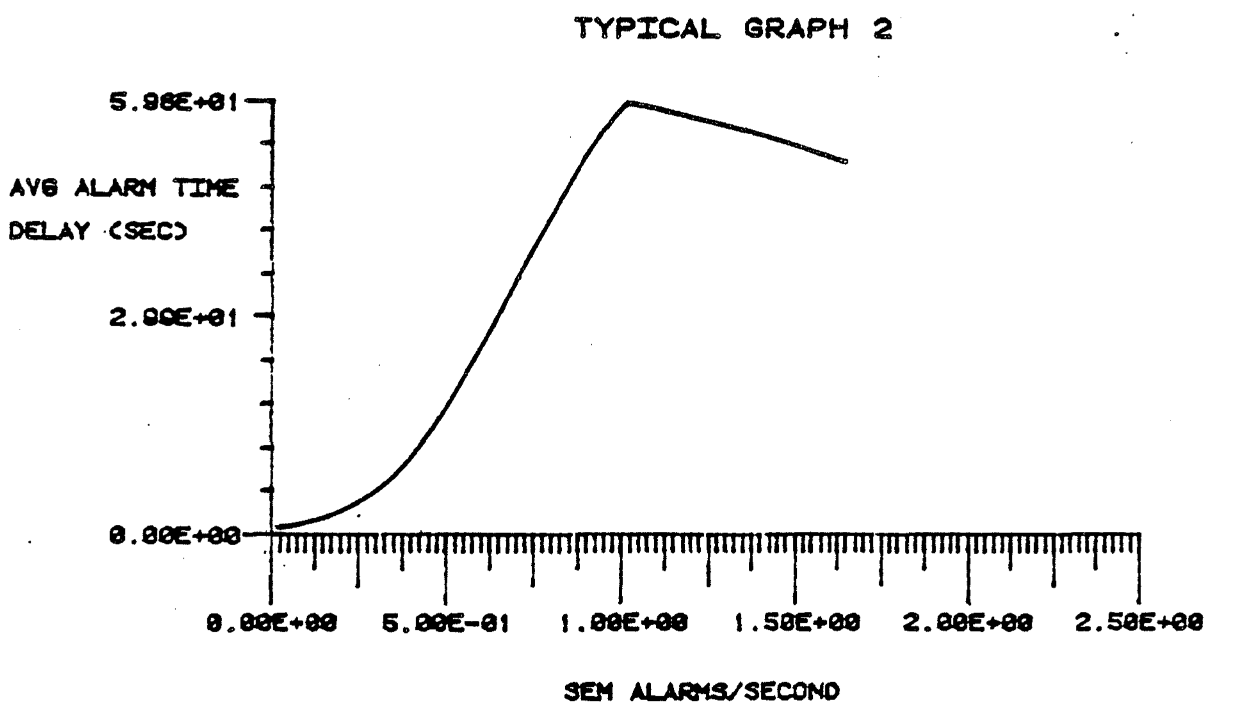


Fig. 2.4b Average Alarm Time Delay vs Square Wave Frequency

pulse width of the pulse test, then the SUT will not detect the status changes of all of its inputs.

For large pulse widths the percentage of missed alarms is small; ideally, it should be none. As the pulse width shortens, though, the missed alarm percentage starts to rise. The point on the x-axis at which the percentage of missed alarms dramatically rises is the scan time of the SUT. The scan time of this particular SUT is approximately 1 second. (The SUT contained 420 points of which 348 were used in the test.)

Graph 3 shows the phase differences between the scan cycle of the SUT and the input square wave (see Figure 2.3c). As the alarm frequency increases, the average alarm report time delay increases then starts to decrease as the square wave approaches a harmonic frequency of the scan cycle. A small drop in the percentage of missed alarms can be noticed in Graph 1 as the square wave approaches a harmonic frequency. The drop indicates that the SUT was able to detect more of the alarm conditions. Generally, only SUT's using a polling technique to scan their points will have these characteristics. The scan time of the SUT and the maximum alarm frequency should be very close to one another.

Graph 4 shows that the SUT responds at a consistant rate of 220 milliseconds over a large range of pulse widths until the pulse width reaches the scan time of the SUT (see Figure 2.3d). There were indeed some missed alarms during the test because a few of the points of the points of the system under test were faulty.

These graphs present the "measured" values for the SUT in a given configuration of hardware and software. For sample systems, these values reflect the hardware capabilities of the system under test. For example, data transmission rates, outstation scanning period, and physical alarm reporting system (i.e., printer throughput) are hardware limited performance specifications. As system complexity increases, the measured values are affected by SUT software performance. Particularly in computer-based monitoring and control, additional features are often provided by software, not hardware. The test criteria employed measures that integrated system performance by exercising both the hardware and the software of te SUT. This approach is increasingly important to the evaluation of systems that have computer software as a main component.

"Unmeasured" parameters of the SUT are also provided by the testing methodology employed. In order to conduct an integrated test of hardware and software, it is necessary to set up the SUT as if it were to be used in an actual mine environment. This setup provides installation experience with SUT hardware and software. The result is an unmeasured (and admittedly subjective) appreciation for "ease" of installation and maintenance of the SUT.

TYPICAL GRAPH 3

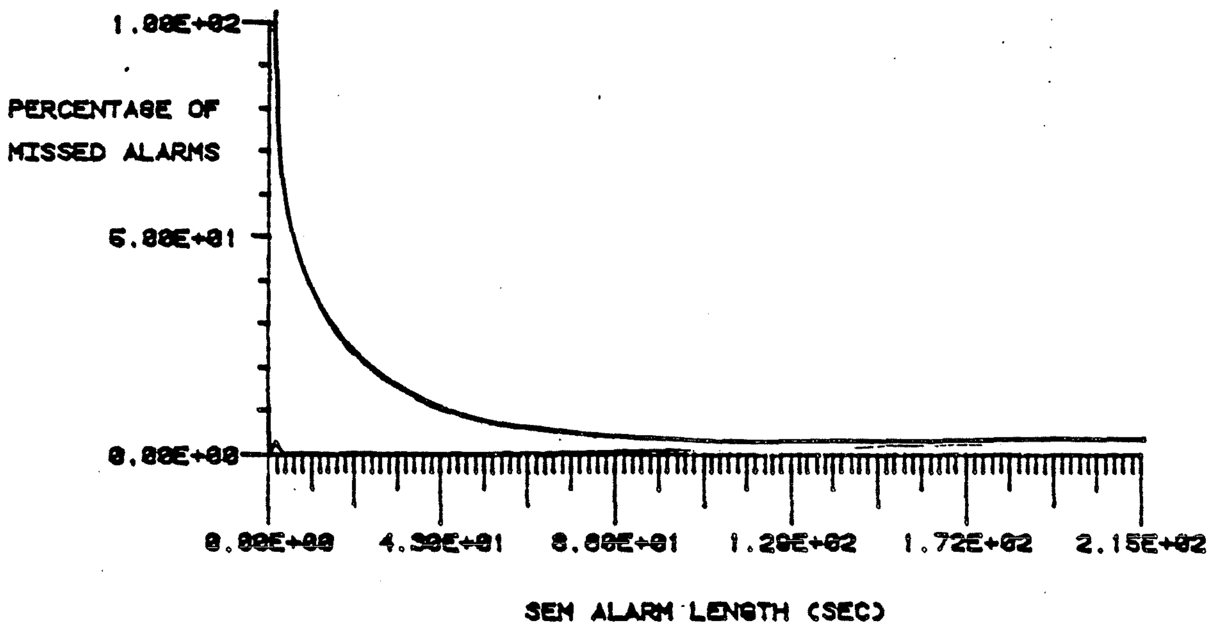


Fig. 2.4c Percentage of Missed Alarms vs Pulse Width

TYPICAL GRAPH 4

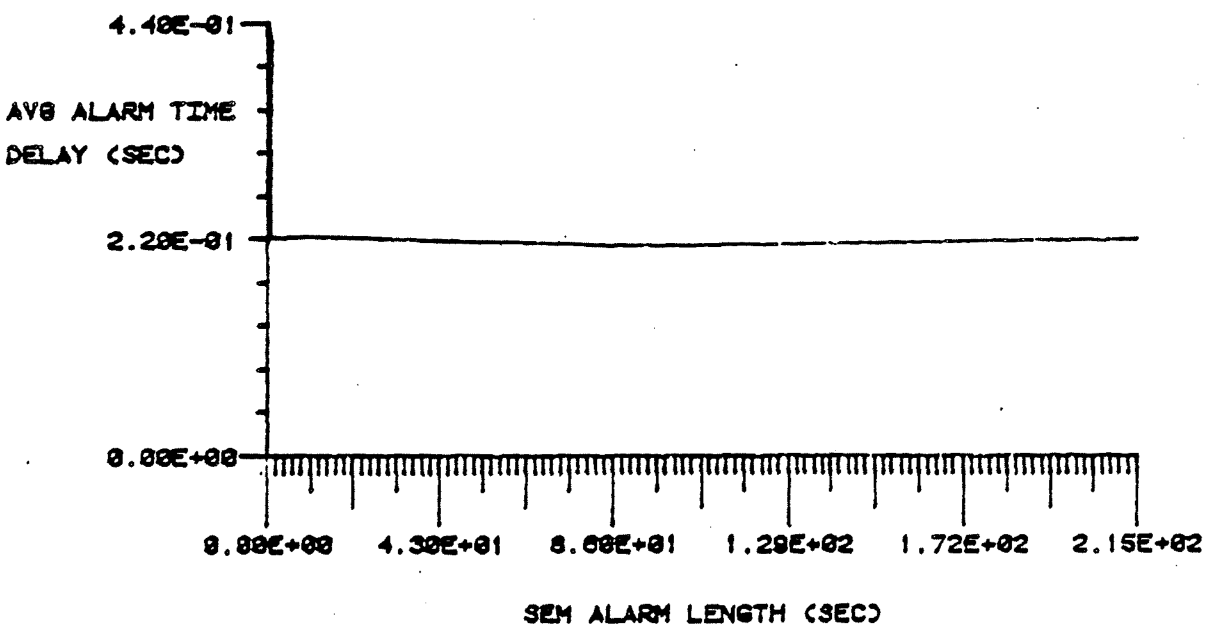


Fig. 2.4d Average Alarm Time Delay vs Pulse Width

Even though it is difficult to quantitize this unmeasured data, it is valuable information in the overall evaluation of a monitoring and control system.

3.0 APPENDIX

3.1 SEM

The Sensor Emulator (SEM), based upon an Intel 8085 microprocessor, is designed to work together with the Dynamic Test Controller (DTC) to evaluate the reliability of coal mine monitoring systems. It emulates the analog or digital signals from sensors installed in coal mines, and fluctuates these signals over time as real sensor outputs do. Each SEM is able to offer five basic capabilities for the testing system: first, communication with the DTC; second, acceptance of commands and test data from the DTC; third, distribution of test data to the target system (the monitoring system under test); fourth, monitor its own operations; and fifth, monitor the signals from another SEM (SEM partner). Figure 3.1 shows the SEM system configuration.

3.1.1 SEM Hardware Handbook

Figure 3.1.1 shows the block diagram of the SEM. The communications between the DTC and the SEM are carried out through an RS-232C serial interface. Test data from the DTC are also brought in over this serial input line. After being processed, the test data are stored in the buffer residing in internal RAM to wait for commands from the DTC to be sent out to the target system. Four analog output channels and two 8-bit digital output channels are provided in the SEM to meet the target system requirements. To allow for SEM self and partner monitoring, four analog and two 8-bit digital input channels, as well as a power failure control circuit are also provided.

The SEM consists of two major functional modules, the CPU board and the I/O board. These two boards are plugged separately into two 100 contact PC edge connectors and interconnected with each other through a universal motherboard.

<u>The CPU Board</u>

The CPU board performs the processing and communicating functions for the SEM. While the 8085A microprocessor provides the board with basic processing capabilities, the performance potential is further enhanced by the remaining logic on the board.

Up to 16K bytes of EPROM and 8K bytes of RAM on the board provide a sufficient storage capability for programs and data. Bus control logic resolves exchanges of bus control between the CPU board and the I/O board. Serial communications capability is obtained through the use of a Universal Synchronous Asynchronous Receiver Transmitter (USART). Three timers are provided on the board to offer accurate interval timing references. A power failure control circuit eliminates the possibility of SEM malfunctions caused by power failures. An edge connector (J2) on the top of the board provides

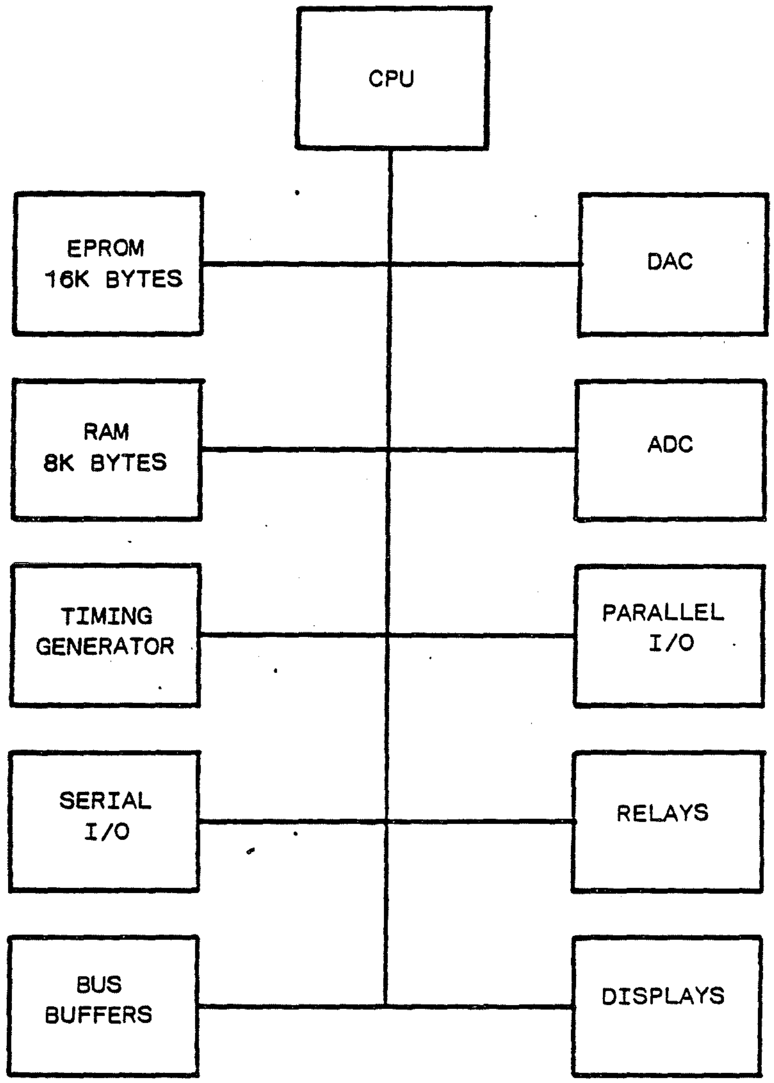


Fig. 3.1. The SEM System Configuration

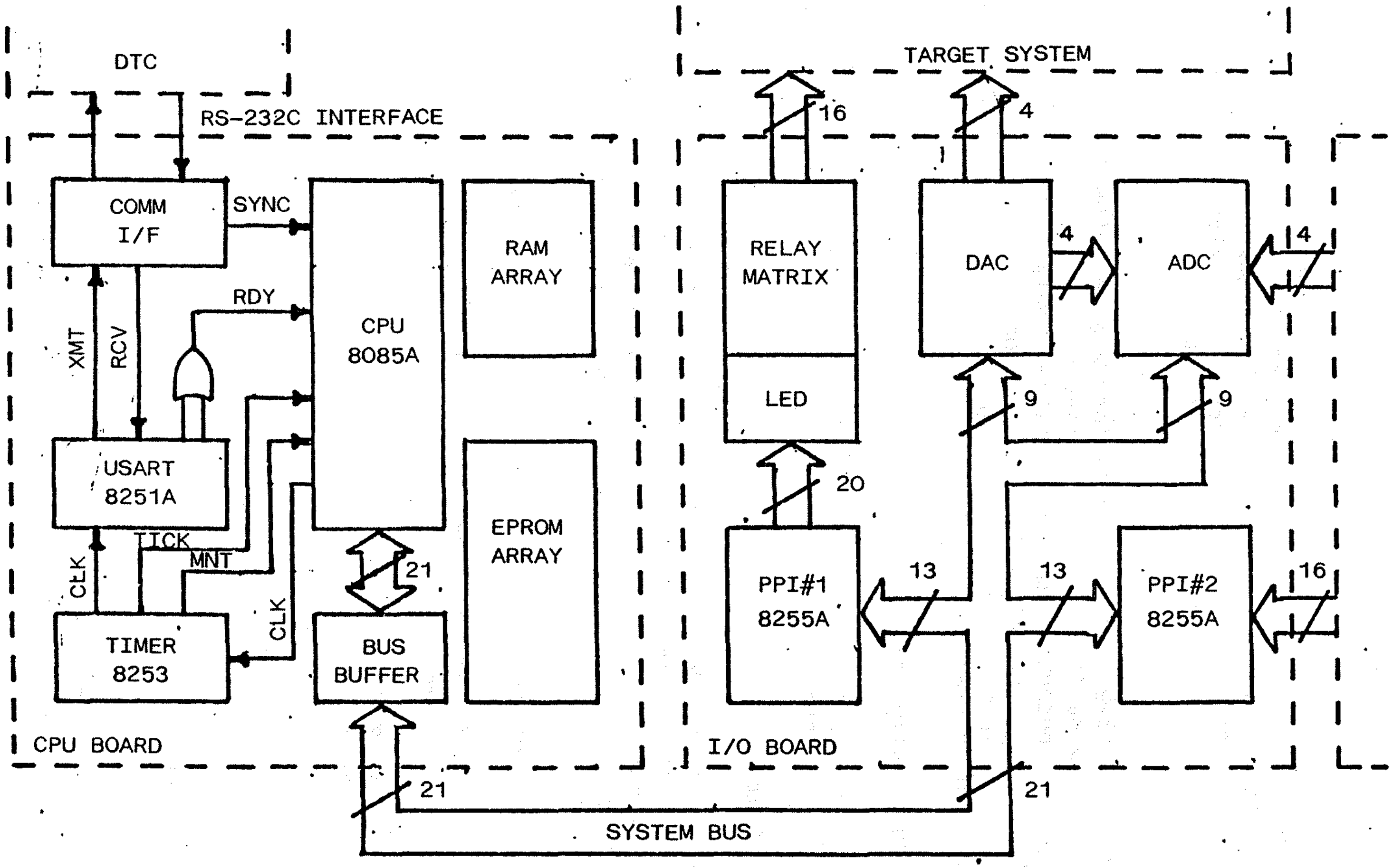


Fig. 3.1.1. The SEM System Block Diagram

access to the DTC, console, or modem. In addition, a five-level interrupt priority scheme makes it possible to resolve simultaneous interrupt requests on a priority basis.

The I/O Board

The basic functions implemented by the I/O board are data acquisition and input/output operations. There are four analog output channels on the board, each supplies an analog signal ranging from 0 to 10 volts to the target system. The provision of eight analog-to-digital channels allows the SEM to monitor its own analog output operations.

As many as sixteen relay-contact-closure output lines are included on the board. A built-in self-monitoring system, consisting of sixteen relay-status output lines and sixteen relay-status input lines, provides assurance that the SEM digital operations are accurate. The analog and digital interfaces to the target system are through three on-board edge connectors (J2, J3, and J4).

I/O Address Assignment

The I/O address assignment for the SEM is summarized in Table 3.1.1. It will be described in detail later.

THE CPU BOARD

The CPU board has been designed specifically to perform the processing and communicating functions for the SEM. Its on-board memory permits it to provide 16K bytes of program and 8K bytes of data storage capacity. The primary features required on this board are described as follows:

(1) a microprocessor to perform the SEM system controls,

(2) a 16K byte program memory to store the system programs,

(3) an 8K byte data memory to store both test data before being sent out to the target system and other data necessary for program execution,

(4) an RS-232C interface for serial communications,

(5) three programmable timers to provide the SEM with the necessary interval timing references, and

(6) bus drivers to buffer the address, data, and control bus signals.

It requires three DC power supplies, at levels of +5, +12, and -12 volts. All circuitry is mounted on a 10 by 5.325-inch printed circuit board. The power and bus signal connections enter the board through a 100 contact PC edge connector. A 10-pin connector is also present to connect four communications signals and four modem control signals to the panel. The block diagram of the CPU board is shown in Figure 3.1.2.

Table 3.1.1 The SEM I/O Address Assignment

address	I/O operation
00	8251A Data Mode
01	8251A Control Mode
02-03	Not Used
04	8253 TIMER 0
05	8253 TIMER 1
06	8253 TIMER 2
07	8253 Write Mode
08-1F	Not Used
20	8255A#1 PORTA (8 Relays, #1 to #8)
21	8255A#1 PORTB (8 Relays, #9 to #16)
22	8255A#1 PORTC (4 LED Indicators)
23	8255A#1 Control Mode
24	8255A#2 PORTA (8 External Relay Check Inputs, #1 to #8)
25	8255A#2 PORTB (8 External Relay Check Inputs, #9 to #16)
26	8255A#2 PORTC (AD7581 RAM Address Control) (Operational Mode Selections)
27	8255A#2 Control Mode
28	AD558#1
29-2B	Not Used
2C	AD558#2
2D-2F	Not Used
30	AD558#3
31-33	Not Used
34	AD558#4
35-37	Not Used
38	AD7581

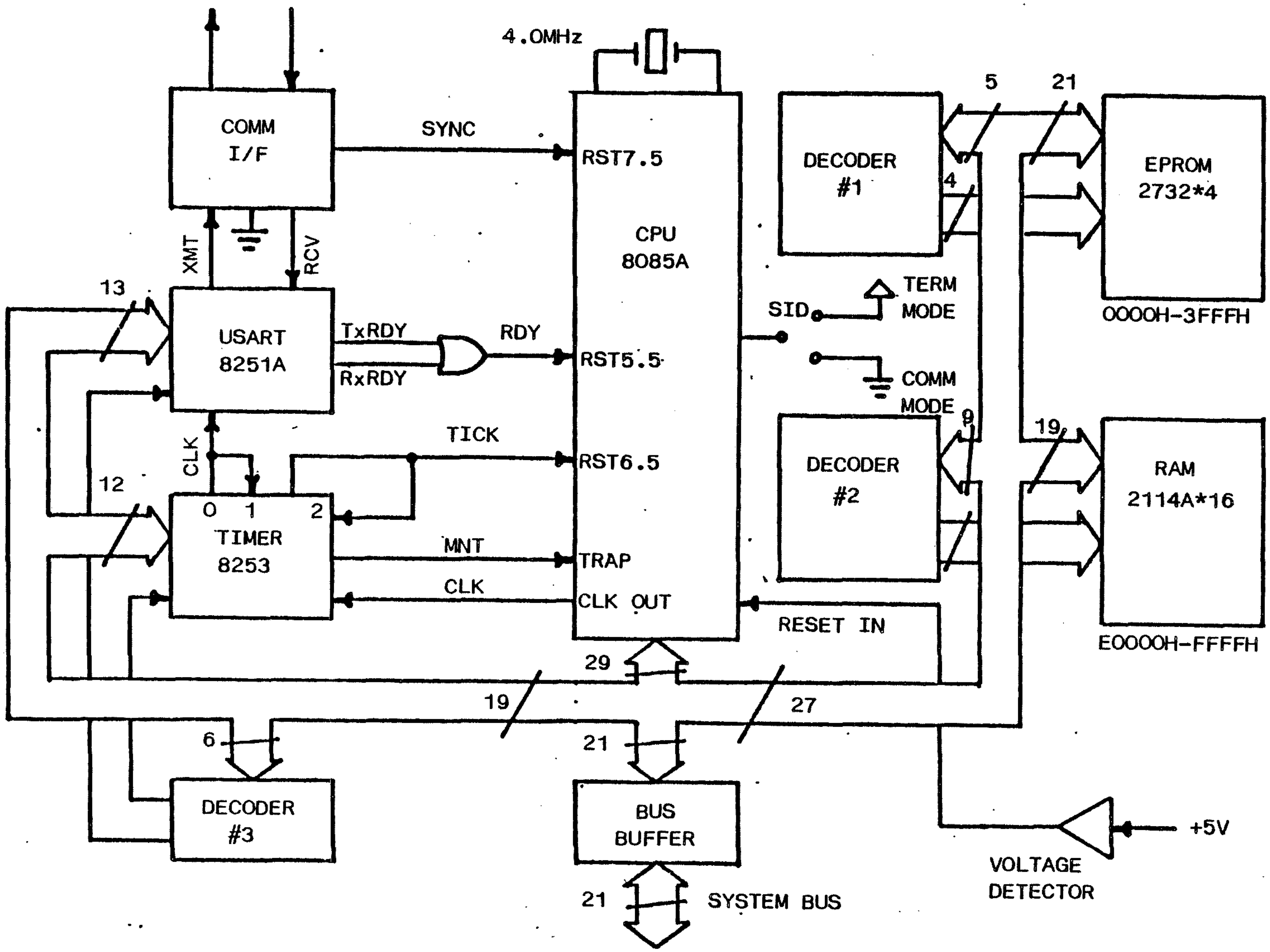


Fig. 3.1.2. The CPU Board Block Diagram

Specifications

(1) Electrical Characteristics:

Voltage	Maximum Current	Maximum Power Dissipation
+5V DC	3A	15W
+12V DC	30mA	0.36W
-12V DC	25mA	0.30W

(2) Physical Characteristics:

Width	10 in.
Height	5.325 in.

(3) Environmental Characteristics:

Operating Temperature	0 to 55 degree C.
Operating Relative Humidity	0 to 90%

Central Processing Unit

The basic processing capabilities of this board are obtained through the use of an Intel 8085A microprocessor. This processor contains an 8-bit accumulator, six 8-bit general purpose registers and an 8-bit parallel Arithmetic and Logic Unit (ALU). A 16-bit program counter and a 16-bit stack pointer permit flexible handling of subroutines and multilevel interrupts.

The 8085A's clock is derived from a 4 MHz crystal. This clock frequency is divided by 2 to give the processor an internal operating frequency, as well as provide a stable system timing reference for all circuitry in the SEM. This frequency also permits a basic clock cycle of 500ns for the 8085A's instruction executions. The specifications of this chip are given in the Intel Component Data Catalog, available from Intel Corporation.

Bus Structure

Sixteen address lines, eight data lines, and six control lines are used for all memory and input/output operations. Since the 8085A multiplexes the low-order 8-bit address with 8-bit data on the address/data bus, an Intel 8282 is used to latch the low-order 8-bit address when the ALE signal is activated. A buffered system bus for all input/output operations to the I/O board is provided on this board. Hence, the on-board operations do not tie into the system bus.

Address Bus Buffer: An address bus driver (74LS244), which can source a 15mA current at high logic level and sink a 24mA current at low logic level, buffers the eight least significant address bits before sending them out of the board. These eight address lines are transparent to the system bus at all times.

Control Bus Buffer: A 7407 is designated to buffer six control bus signals (RD, WR, RESET OUT, IO/M, and ALE) sent out from the processor. This device is enabled unconditionally and provides all the internal and external control signals with a 40mA current-drive capability.

Data Bus Buffers: Two bidirectional bus drivers (8216's) are used to buffer the external data bus. These two devices isolate the internal data bus from the system bus, as well as provide a high current-drive capability (50mA) to the external data bus. The control logic is designed so that it is enabled only when out-of-board I/O operations are necessary.

BDBE is a control signal that enables both 8216's when it is in the low state. During an out-of-board input/output operation, IO1, IO2, and IO/M are all in high states. The high-level signals are gated, producing a high-level IOE signal. This signal is gated with a high-level RWE, which is gated from RD and WR signals, to produce an active BDBE signal. When BDBE is active, the direction of the data flow is determined by the WR signal. The truth table for controlling the 8216's is given in Table 3.1.2, in which * signs indicate "don't care" states.

In the output mode, the internal data bus content is loaded into the 8216's that, in turn, drive the external data bus. In the input mode, data from the external data bus is transferred to the internal data bus. The 8216's are switched off at other times.

Table 3.1.2 The Truth Table for Controlling 8216's

IO1	IO2	IO/M	RD	WR	8216's
Low	*	*	*	*	Disabled
*	Low	*	*	*	Disabled
High	High	Low	*	*	Disabled
High	High	High	Low	High	Data In
High	High	High	High	Low	Data Out

<u>Power Failure Control</u>

The power failure control circuit is intended to monitor a power failure and reset the processor automatically. On power-up, the zener diode (1N748) clamps the inverting input of the voltage comparator (LM393) at approximately 4.0 volts a short time after the power supply reaches 4.0 volts. At the same time, the voltage on the non-inverting input is kept at less than 4.0 volts by the RC time constant produced by R3, R5, and C38, which causes the output of the voltage comparator to be low. When the voltage on the non-inverting input rises above 4.0 volts, the output of the voltage comparator goes high and the processor is reset. The non-inverting input is kept at approximately 4.36 volts by the voltage divider created by R3 and R5.

An 1N4148 diode is used to ensure that the voltage in the RC network stays below the supply voltage. If for any reason the voltage on the non-inverting input rises above the voltage given by the power supply plus the diode voltage, the diode will begin to conduct and dissipate the excess power.

A push-button switch that is connected from the front panel to the output of the voltage comparator enables a manual reset operation. The capacitor across the switch is necessary to stablize the voltage comparator.

Random Access Memory (RAM)

The RAM array has been designed to provide the SEM with a data storage capacity of 8K bytes. This RAM array is comprised of sixteen 2114A's (1K by 4 static RAM), which can complete a read cycle or a write cycle in 450ns.

The RAM decoding logic consists of an 8205 three-to-eight decoder and three 74LS08 AND gates. The six most significant address bits (AB10 - AB15) and a control signal (IO/M) are used to select the proper chip pair whenever a data read or data write operation is activated. The remaining ten least significant address bits (AB0-AB9) are applied to each 2114A to identify each memory element.

Because the 2114A's only have a memory-write enable input (WE) to decide the read or write operation of these chips, it is necessary to generate a RW signal to the RAM decoder (8205), avoiding contention problems on data bus. This signal is generated by using a 74LS08 AND gate to get RD and WR signals AND'ed together.

Erasable Programmable Read Only Memory (EPROM)

The system programs and interrupt vectors reside on four 2732 EPROM's that have been designated to provide 16K bytes of storage capacity. These EPROM's can be accessed in 450ns.

The EPROM decoding logic consists of an 8205 three-to-eight decoder only. Address lines AB12-AB15 and control signal IO/M are fed to the 8205 to select the proper chip in a data read operation. The remaining address lines, specified by AB0-AB11, are connected to each 2732 to identify each memory element. Each EPROM is enabled by an associated decoder's output and a RD signal.

Memory Allocation

Table 3.1.3 illustrates the address assignment of the memory system. This arrangement allows further RAM or EPROM expansion without conflict.

Table 3.1.3 Address Assignment of the Memory System

component	unit	address	function
2732	4	0000H - 3FFFH	program
2114A	16	E000H - FFFFH	data

Input/Output Interface

There are two input/output components, 8253 and 8251A, provided on this board to offer timing reference and serial input/output capabilities. Using an 8205 three-to-eight decoder, each component is selected by the address bus from AB2 through AB6. AB0 and AB1 are connected to the 8253 to select one of the three timers to be operated on and to address the control word register for mode selection. However, the mode selection of the 8251A is determined by AB0 only. An input/output address assignment is shown in Table 3.1.4.

Table 3.1.4 Input/Output Address Assignment for the CPU Board

address	I/O operation
00	8251A Data Mode
01	8251A Control Mode
02	Not Used
03	Not Used
04	8253 TIMER 0
05	8253 TIMER 1
06	8253 TIMER 2
07	8253 Write Mode

Serial Communications

A programmable communications interface using Intel's 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on this board to provide full-duplex, double-buffered receive and transmit capabilities for the SEM. The inclusion of the RS-232C compatible interface, which consists of a line driver (MC1488) and a line receiver (MC1489), allows the SEM to communicate with the DTC.

The required clock for both receiver and transmitter is obtained from one of the 8253 timers, TIMER 0. The RESET OUT and CLK OUT signals from the 8085A are also needed for the 8251A's operation. The former is used as an initialization signal, and the latter is used for the internal sequencing of the 8251A.

A set of control inputs and outputs that can be used to simplify the interface to almost any modem are also available on this board, as listed below:

(1) DSR (Data Set Ready),

(2) DTR (Data Terminal Ready),

(3) RTS (Request To Send),

(4) CTS (Clear To Send).

Another signal, SYNDET, is used in synchronous communications mode to detect a synchronous character. The four modem control signals, SYNDET signal, and RS-232C serial data lines are brought out to a 10-pin edge connector (J2).

The 8251A can be programmed by the system program to select the desired mode of operation and data format. The command instruction format and status read format are also under program control. It should be noted that the 8251A requires careful programming to conform to a specified sequence for proper device operation. For the detailed theory of operation, definitions of formats, and programming sequence of the 8251A, refer to the Intel Component Data Catalog.

There are two types of serial communications that the SEM might use. The first one is the communications mode that handles the serial communications with the DTC; the second one is the terminal mode that deals with serial communications with a terminal. On a jumper selection basis, these two communications types are identified by the SID input of the 8085A.

Programmable Timers

The SEM provides three fully programmable and independent timers utilizing an Intel's 8253 Programmable Interval Timer. The frequency of the input clock to TIMER 0, which is used as a programmable baud-rate generator for the USART, is derived directly from the clock output (CLK OUT) of the 8085A. This input clock to TIMER 0 (2 MHz) is further divided by a value to give the receiver and transmitter of the 8251A a proper real time clocking. The output of the TIMER 0 is also routed to the input of TIMER 1, which generates a TICK signal to interrupt the processor. For instance, the TICK signal can be programmed to generate a time interval, ranging from one millisecond to one second, if it is activated by a 19.2 KHz frequency. The third timer (TIMER 2) in the 8253, with its input clock coming from the clock output of TIMER 1, is specified to generate a TRAP interrupt signal.

The gate inputs of both TIMER 0 and TIMER 1 are hardwired to +5 volts power to enable these two timers continuously. The gate input of TIMER 2 is jumper selectable to give the user a capability to enable or disable this timer.

Programming the 8253 is quite simple. Whenever a given baud rate or time delay is needed, a control word and a count are to be loaded to the corresponding timer. The detailed operational descriptions of the 8253 are described in the Intel Component Data Catalog.

Interrupt Processing

Several hardware interrupt vectors of the 8085A are defined to enable some interrupt services:

(1) RST 0 - The power failure control circuit is designed to detect a power voltage drop. When a power failure has been detected or a RESET switch has been pushed, the program will start from location 0000H.

(2) TRAP - An overflow of TIMER 2, indicating that program execution has been aborted for a specified time, generates a TRAP interrupt signal which causes the 8085A to enter into a trap service routine through vector location 0024H. The processor responds by sending out an error message to the DTC and zeroing all SEM's outputs.

(3) RST 5.5 - The Receiver Ready and Transmitter Ready outputs from the 8251A are OR'ed together and then connected to RST 5.5. Either one, if activated, will request a communications service routine to receive data from or transmit data to the DTC.

(4) RST 6.5 - A synchronous signal (SYNC), which is actually a pulse signal, comes from the DTC through a line receiver (MC1489) to enable RST 7.5 and in turn start a series of output operations.

(5) RST 7.5 - RST 7.5 is connected to TIMER 1 and generates a TICK signal to interrupt the processor. In the TICK service routine, the 8085A inputs the available data from an 8-channel

analog-to-digital converter (AD 7581) to implement point-to-point comparisions.

Table 3.1.5 summarizes the interrupt assignment for the SEM.

Table 3.1.5 The Interrupt Assignment for the SEM

name	priority	trigger type	branch address	usage
RST 0	1	Rising Edge	0000H	RESET
TRAP	2	Rising Edge and High Level	0024H	MNT
RST 7.5	3	Rising Edge	003CH	TICK
RST 6.5	4	High Level	0034H	SYNC
RST 5.5	5	High Level	002CH	RDY

<u>Pin and Parts Lists</u>

The connector pin allocations on the CPU board are described as follows. The pins and their designated signal functions for the bus connector (J1) are listed in Appendix A. The same information for the communications connector (J2) is shown in Appendix B. The parts list for this board is listed in Appendix C. All of the parts listed may be obtained directly from vendors.

<u>Printed Circuit Board</u>

Figure 3.1.3 shows the printed wiring assembly of the CPU board. The printed circuit board layouts and some other miscellaneous drawings prepared for mastering or assembly are given in [5].

<u>THE I/O BOARD</u>

The I/O board interfaces directly to the CPU board via the system bus to facilitate the analog and digital input/output capacities between the SEM and the target system. This board basically consists of two 8255A Programmable Peripheral Interfaces, four digital-to-analog converters, and one 8-channel analog-to-digital converter, together with sixteen reed relays and four LED indicators. It also contains two jumpers to identify the current operational modes. Figure 3.1.4 shows the block diagram of this board.

It operates with three DC power supplies, at levels of +5, +12, and -12 volts. All circuitry is mounted on a 10 by 5.325 inch printed circuit board through a 100 contact PC edge connector. One 10 pin

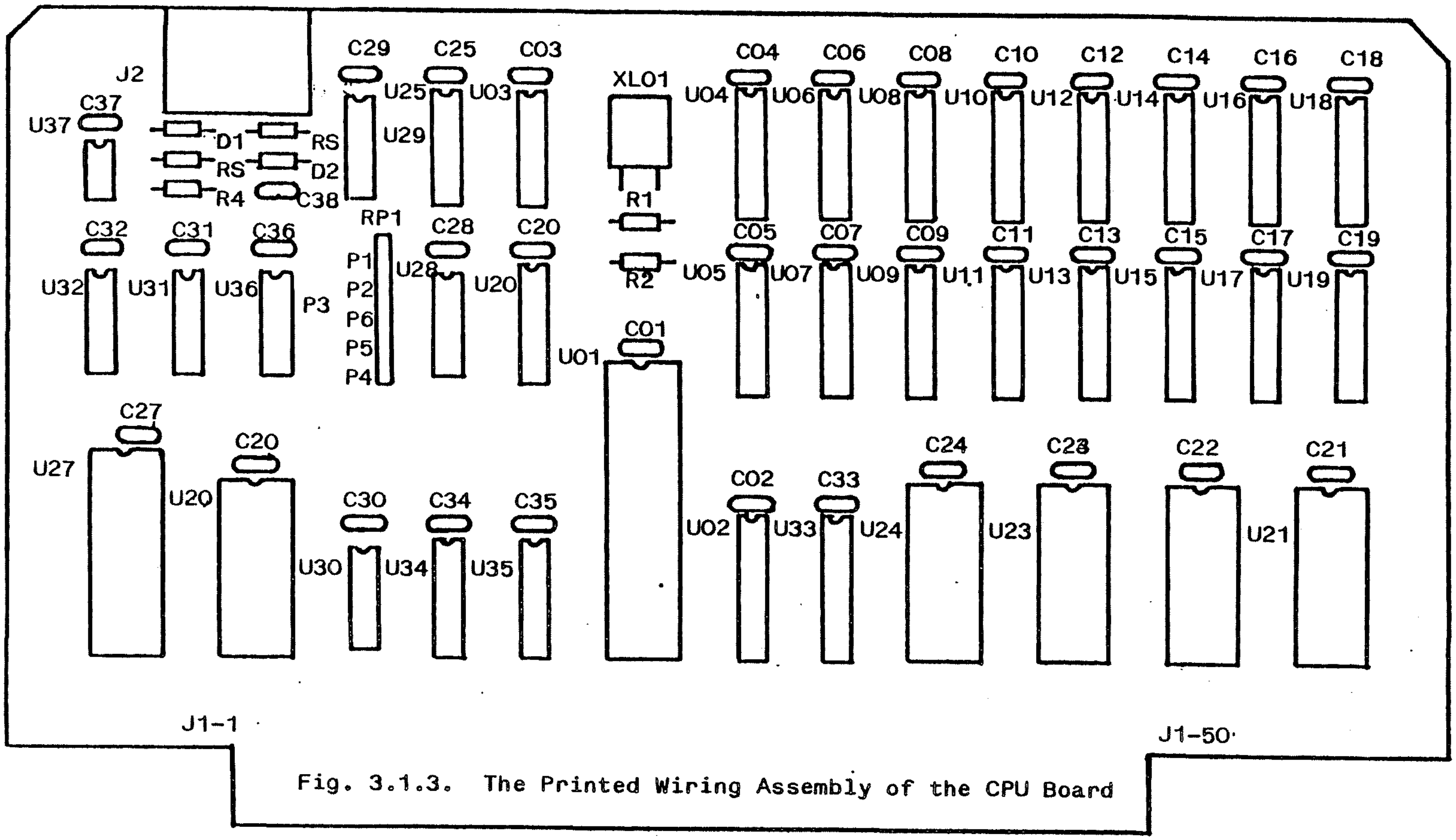


Fig. 3.1.3. The Printed Wiring Assembly of the CPU Board

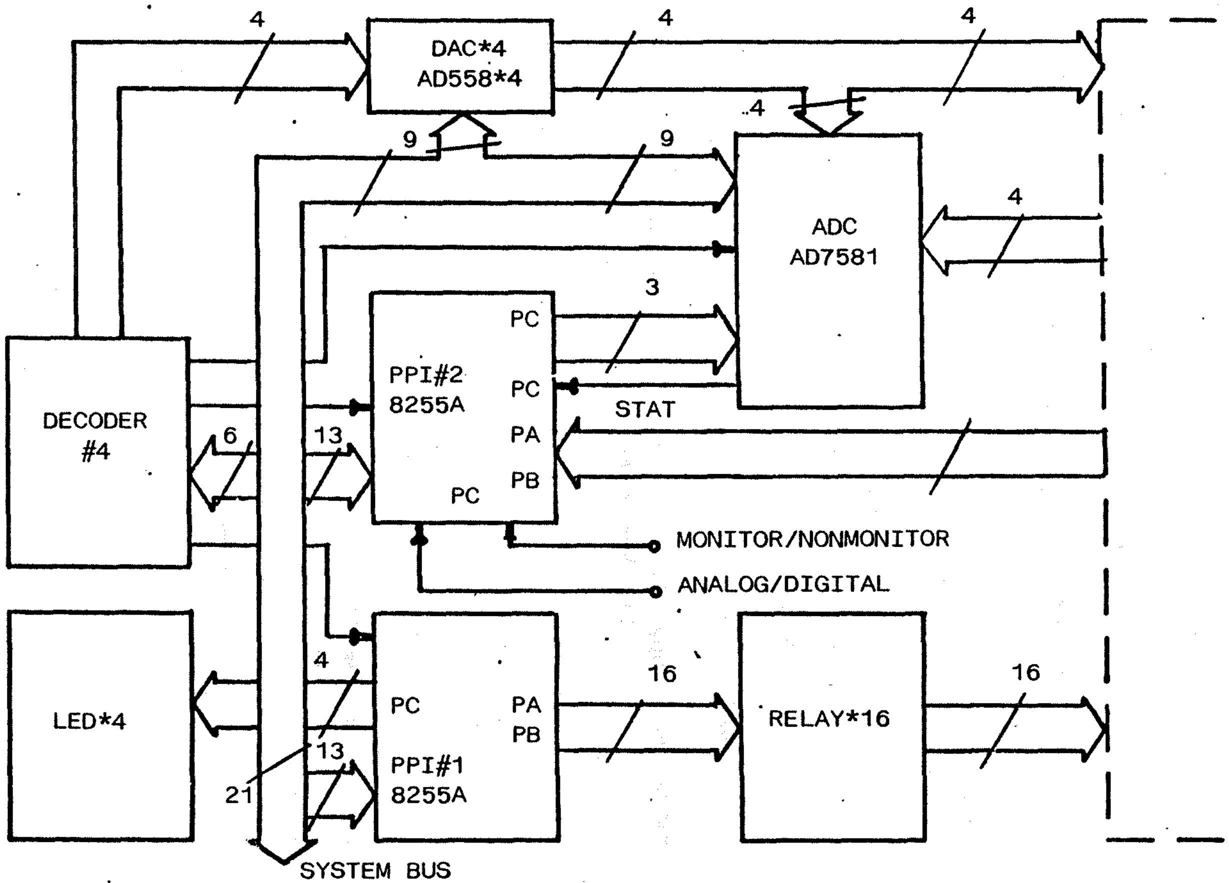


Fig. 3.1.4. The I/O Board Block Diagram

and two 34 pin edge connectors are contained on the board for interfacing four analog inputs, four analog outputs and sixty-four relay-related input/output signals to the target system and SEM partner. The functions of the SEM partner is described in the SEM System Handbook.

Specifications

(1) Electrical Characteristics:

voltage	maximum current	maximum power dissipation
+5V DC	1.1A	5.5W
+12V DC	0.1A	1.2W
-12V DC	45mA	0.54W

(2) Physical Characteristics:

Width	10 in.
Height	5.325 in.

(3) Environmental Characteristics:

Operating Temperature	0 to 55 degree C.
Operating Relative Humidity	0 to 90%

(4) Analog Input:

Number of Input Channels	4
Resolution	8-bit
Accuracy	+/- 1 and 7/8 LSB
Input Voltage Range	0 to +10 volts
Input Impedance	10/20/30K Ohm min/typ/max
Temerature Coefficient	+/- 150uV/degree C.

(5) Analog Output:

Number of Output Channels	4
Resolution	8-bit
Accuracy	+/- 1/2 LSB
Output Voltage Range	0 to +10 volts
Output Current	+5mA
Temperature Coefficient	-50uV/degree C.

(6) Digital Input:

Maximum Voltage	50V DC
Maximum Current	0.25A

and two 34 pin edge connectors are contained on the board for interfacing four analog inputs, four analog outputs and sixty-four relay-related input/output signals to the target system and SEM partner. The functions of the SEM partner is described in the SEM System Handbook.

Specifications

(1) Electrical Characteristics:

voltage	maximum current	maximum power dissipation
+5V DC	1.1A	5.5W
+12V DC	0.1A	1.2W
-12V DC	45mA	0.54W

(2) Physical Characteristics:

Width	10 in.
Height	5.325 in.

(3) Environmental Characteristics:

Operating Temperature	0 to 55 degree C.
Operating Relative Humidity	0 to 90%

(4) Analog Input:

Number of Input Channels	4
Resolution	8-bit
Accuracy	+/- 1 and 7/8 LSB
Input Voltage Range	0 to +10 volts
Input Impedance	10/20/30K Ohm min/typ/max
Temerature Coefficient	+/- 150uV/degree C.

(5) Analog Output:

Number of Output Channels	4
Resolution	8-bit
Accuracy	+/- 1/2 LSB
Output Voltage Range	0 to +10 volts
Output Current	+5mA
Temperature Coefficient	-50uV/degree C.

(6) Digital Input:

Maximum Voltage	50V DC
Maximum Current	0.25A

(7) Digital Output:

Maximum Voltage	50V DC
Maximum Current	0.25A

I/O Addressing Logic

The I/O board utilizes the I/O mapping technique to address any I/O port on the board. It accepts the seven least significant address bits (BAB0-BAB6) and some control signals (RD, WR, ALE, RESET OUT) from the CPU board. Both BAB0 and BAB1 control the selection of one of the three ports or the control word registers in each 8255A. The five high-order address bits (BAB2-BAB6) together with a control signal (IO/M) are used to specify one of the seven on-board I/O devices to be accessed.

Three NAND gates (74LS00) are provided to gate I09 [the chip enable line of the analog-to-digital converter (AD7581)] with a control signal (RD) to avoid data bus contention problems. In addition, the I/O address assignment is given in Table 3.1.6.

Table 3.1.6 The I/O Address Assignment for the I/O Board

I/O Addr	Description
20H	8255A#1 PORTA (8 Relays, #1 to #8)
21H	8255A#1 PORTB (8 Relays, #9 to #16)
22H	8255A#1 PORTC (4 LED Indicators)
23H	8255A#1 Control Mode
24H	8255A#2 PORTA (8 External Relay Check Inputs, #1 to #8)
25H	8255A#2 PORTB (8 External Relay Check Inputs, #9 to #16)
26H	8255A#2 PORTC (AD7581 RAM Address Control) (Operational Mode Selections)
27H	8255A#2 Control Mode
28H	AD558#1
2CH	AD558#2
30H	AD558#3
34H	AD558#4
38H	AD7581

Programmable Peripheral Interfaces

The 8255A Programmable Peripheral Interface (PPI) contains four ports, two with eight bits each (PORTA and B), and two with four bits each (PORTC UPPER and LOWER). Each port can be programmed via the mode control register inside the 8255A to be either all inputs or all outputs. In addition, PORTC UPPER and PORTC LOWER may be combined together to become an 8-bit port, denoted by PORTC, under program control. For the detailed operational descriptions of the 8255A, refer to the Intel Component Data Catalog.

PORTA and PORTB of the 8255A#1 are designated to control sixteen reed relays. These sixteen output lines go to three 74LS06 buffers to provide a higher current-drive capability (40 mA) for reed relays. The upper four bits of PORTC, which are also programmed as the output mode, go to another 74LS06 buffer to drive four LED indicators.

Each reed relay has two separate contacts with the maximum contact ratings of 0.25A/50V DC. One of them is used to provide a digital signal to the target system. The other is assigned to offer a current relay status to another SEM for carrying out a checking procedure to determine if the digital output operation is correct.

Four LED indicators are mounted on this board to display any processing failure, memory failure, analog failure, or digital failure. The physical locations of these LED's are given in Figure 3.1.5, the printed wiring assembly. The definitions of failures are described in the SEM System Handbook.[5]

PORTA and PORTB of the 8255A#2 are programmed as the input mode to accept sixteen relay status signals from another SEM. The four most significant bits of PORTC are specified as the output mode to provide the AD7581 with three address decoding inputs. The four least significant bits of PORTC are programmed to be the input mode to accept one status signal from the AD7581 and two functional-mode signals from two jumpers. The other two modes are used to notify which kinds of signals, analog or digital, are required in conjunction with the testing of the target system.

Digital-To-Analog Converters

Four microprocessor-compatible 8-bit digital-to-analog converters (AD558's) are used to provide four analog outputs (ranging from 0 to +10 volts) to the target system. These analog signals are also fed to the AD7581 to be converted back into digital data for checking that the analog output operations are correct.

All voltage references, output amplifiers, and logic connections are made inside the AD558; therefore, there is no need to perform any calibration except the output voltage range calibration. To do this calibration, first apply 1111,1111 to the AD558 that is to be calibrated; then adjust a corresponding 1K-Ohm gain-potentiometer until a +10V analog output is achieved. This results in a 39.2mV analog output change for a change of 1 LSB in digital input code. The potentiometer assignments for these four devices are given below.

3.1.7 The Potentiometer Assignments for the D/A's

't no.	resistance	assignment
/R01	1K Ohm	AD558#1 Output Range Adjustment
/R02	1K Ohm	AD558#2 Output Range Adjustment
/R03	1K Ohm	AD558#3 Output Range Adjustment
/R04	1K Ohm	AD558#4 Output Range Adjustment

il Converter

successive-approximation analog-to-digital converter 'ts four on-board and four external analog signals, to +10 volts, into digital data. The AD7581 scans the iput channels sequentially about 0.8 milliseconds after The digital value is loaded into an on-chip 8 by 8 RAM each conversion. It takes about 0.64 milliseconds for complete scanning through all eight channels. The CPU m these memory locations when a TICK interrupt is data pair (converted from analog output signals of the iartner) is compared to check the accuracy of analog is.

as an external -10V voltage reference for the AD7581. been designed to supply a current of 3.4mA, so that a ntal tolerance capability can be achieved.

iffset adjustment and gain adjustment potentiometers R13) are used for the device calibration. To prevent these two adjustments, the offset adjustment must med first.

nt: The AD7581's comparator offset is trimmed out r offset pin (pin 1). R07, R08 and VR05 comprise a tap buffered by an operational amplifier (741) and he bipolar offset pin of the AD7581. The calibration follows:

9.5mV to AIN0 (pin9),

0000 to the PORTC UPPER (PC0-PC3) of the 8255A #2,

AD7581,

VR05 until the seven most significant data output in20-pin26) are LOW and the least significant tput bit (pin27) flickers.

Gaiijustment: The following procedure is recommended to adjust the of each analog input channel in the AD7581.

apply +9.941 volts to all analog input channels (pin2-pin9),

select a required channel via the PORTC UPPER (PC0-PC3) of the 8255A #2. The AD7581 channel selection assignment is given in Table 3.1.8,

enable AD7581,

adjust the potentiometer VRn of selected channel until the seven most significant data output bits (pin20-pin26) are HIGH and the least significant data output bit (pin27) flickers,

(select next channel and repeat steps 2, 3 and 4. The ptentiometer assignments for the AD7581 are listed in Tble 3.1.9.

able 3.1.8 The AD7581 Channel Selection Assignments

chael no.	PC0	PC1	PC2	PC3	assignment
0	0	0	0	*	AD558#1 Analog Input
1	0	0	1	*	AD558#2 Analog Input
2	0	1	0	*	AD558#3 Analog Input
3	0	1	1	*	AD558#4 Analog Input
4	1	0	0	*	External Analog Input #1
5	1	0	1	*	External Analog Input #2
6	1	1	0	*	External Analog Input #3
7	1	1	1	*	External Analog Input #4

Note: * means don't care.

Table 3.1.9 the Potentiometer Assignments for the A/D

part no.	resistance	assignment
VR05	50K Ohm	AD7581 Offset Adjustment
VR06	5K Ohm	AD7581 Channel 0 Input Adjustment
VR07	5K Ohm	AD7581 Channel 1 Input Adjustment
VR08	5K Ohm	AD7581 Channel 2 Input Adjustment
VR09	5K Ohm	AD7581 Channel 3 Input Adjustment
VR10	5K Ohm	AD7581 Channel 4 Input Adjustment
VR11	5K Ohm	AD7581 Channel 5 Input Adjustment
VR12	5K Ohm	AD7581 Channel 6 Input Adjustment
VR13	5K Ohm	AD7581 Channel 7 Input Adjustment

Pins and Parts Lists

The connector pin allocations on the I/O board are described as follows. The pins and their designated signal functions for the bus connector (J1) are listed in Appendix D. The same information for the the analog connector (J2) and digital connectors (J3 and J4) are shown in Appendix E, F, G, respectively. The parts list for this board is listed in Appendix H. All of the parts listed may be obtained directly from vendors.

Printed Circuit Board

Figure 3.1.5 shows the printed wiring assembly of the I/O board. The printed circuit board layouts and some other miscellaneous drawings prepared for mastering or assembly are given in [5].

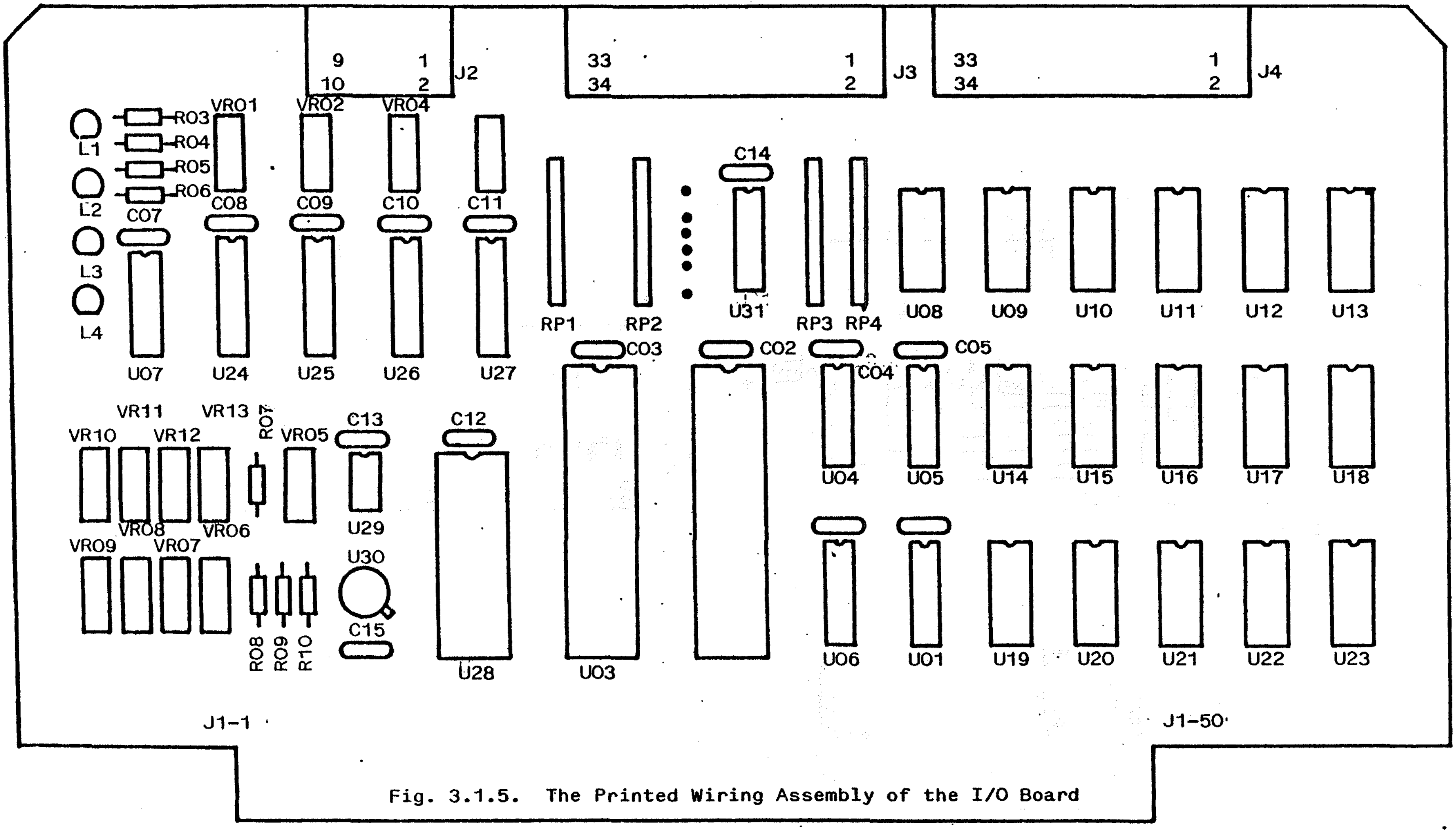


Fig. 3.1.5. The Printed Wiring Assembly of the I/O Board

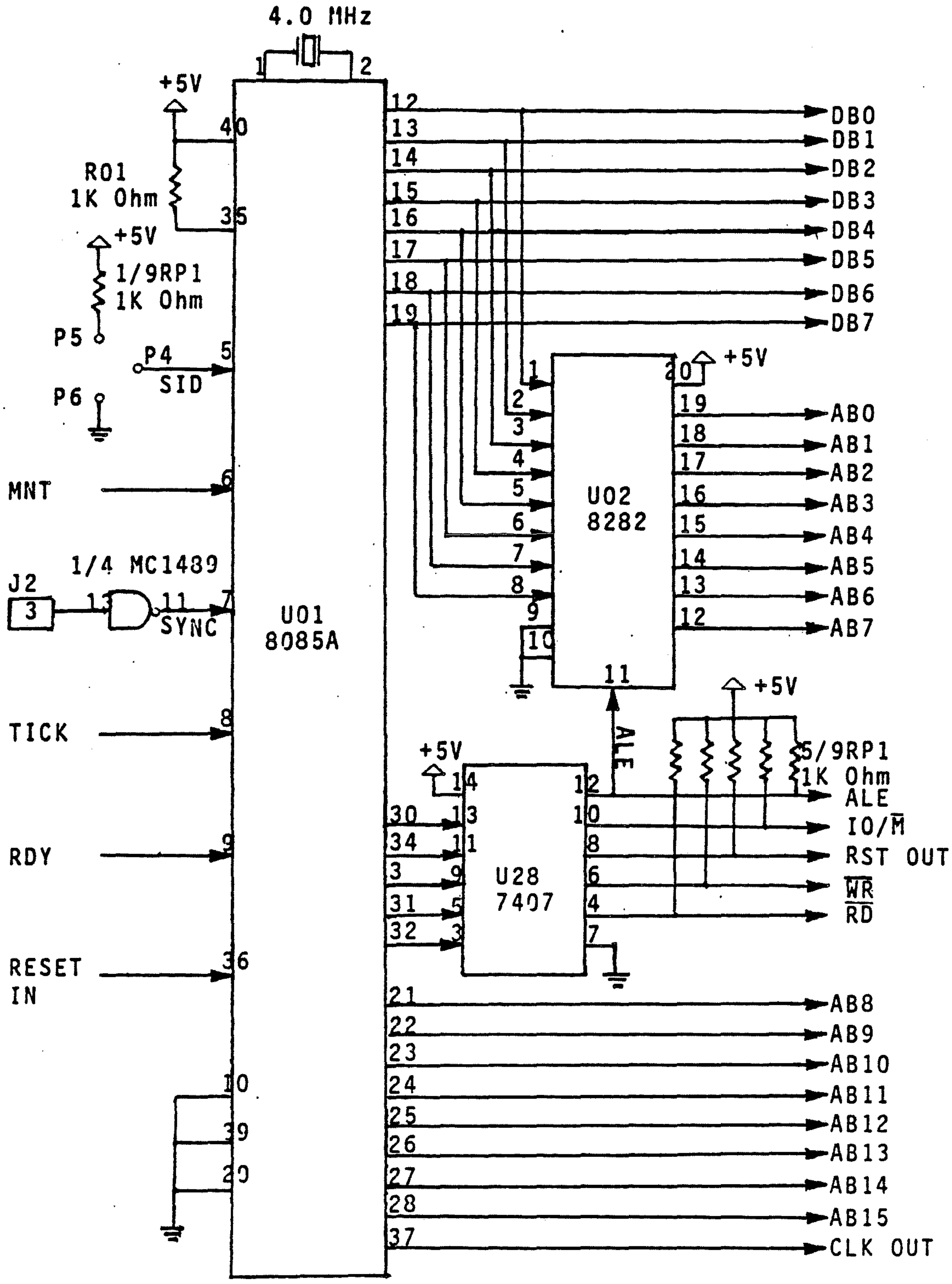


Figure The Circuit Diagram of the 8085A,8282,7407.

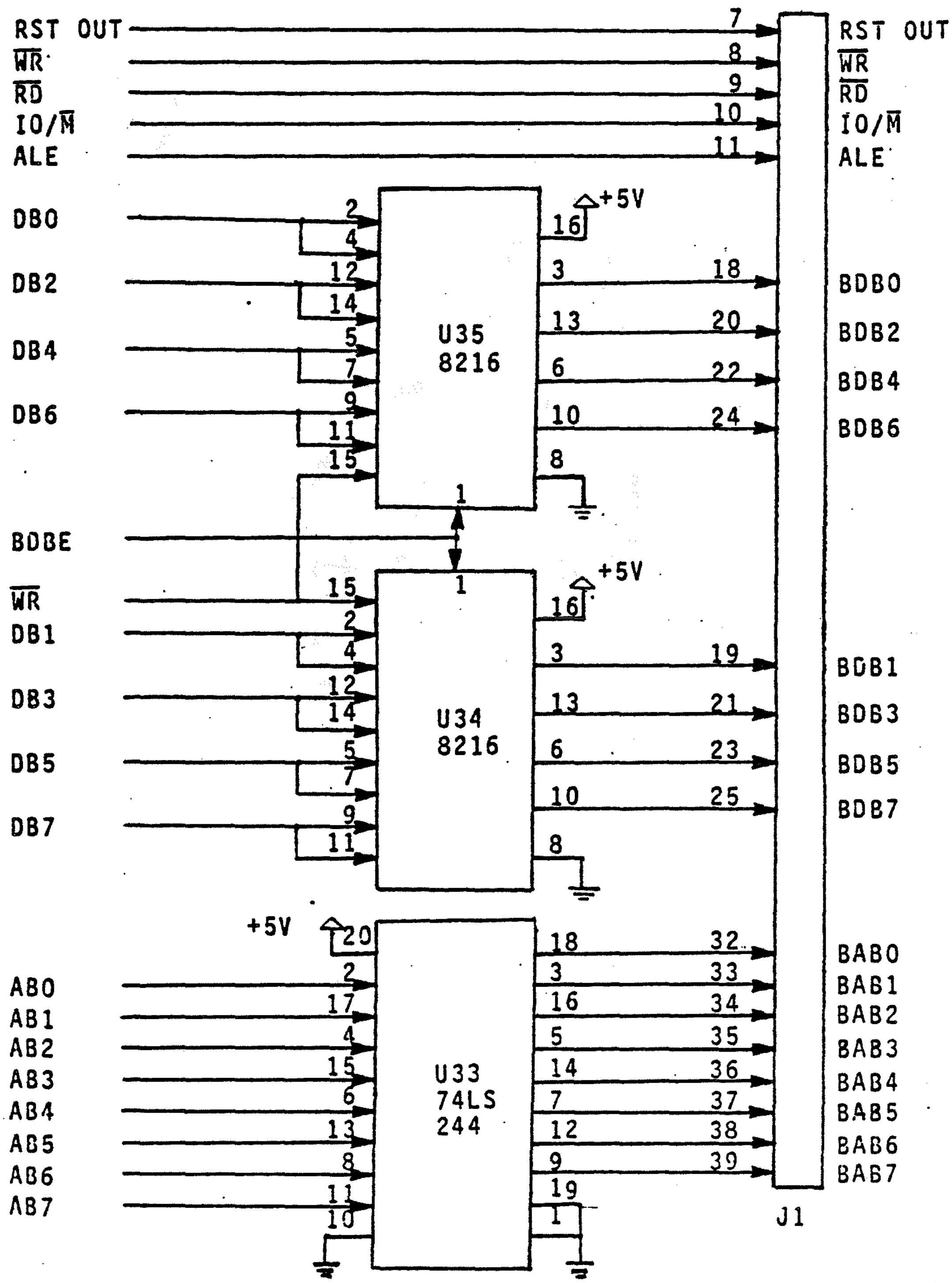


Figure The Circuit Diagram of the Bus Buffers

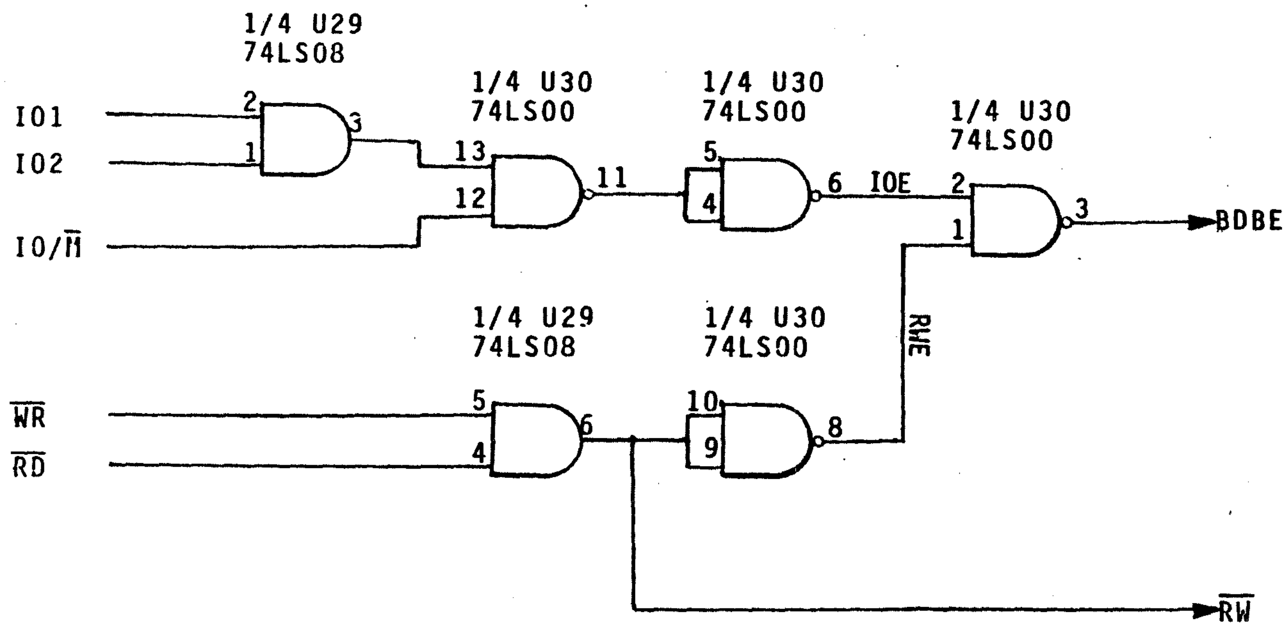


Figure The Circuit Diagram of the 8216's Control Logic

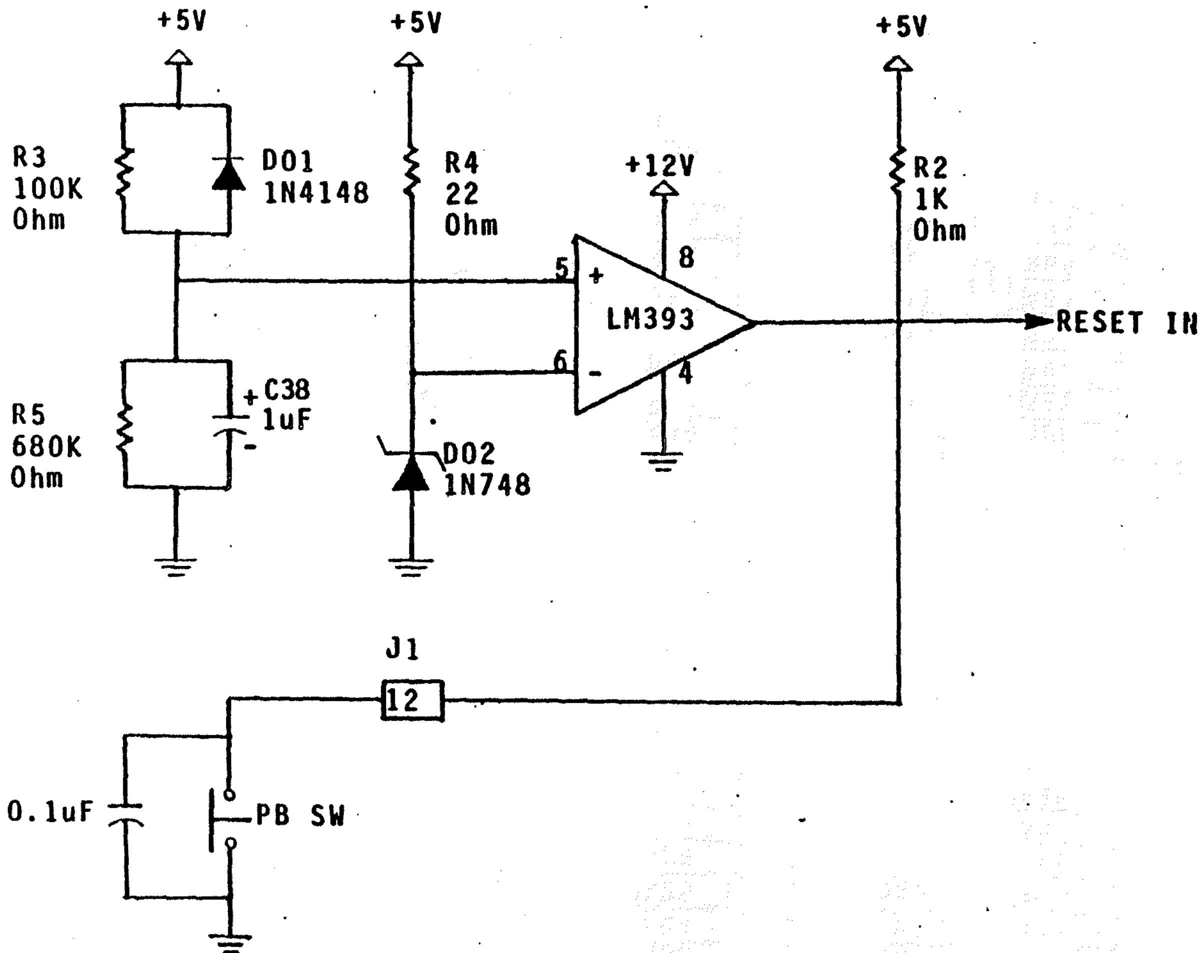


Figure The Circuit Diagram of the Power Failure Control Circuit

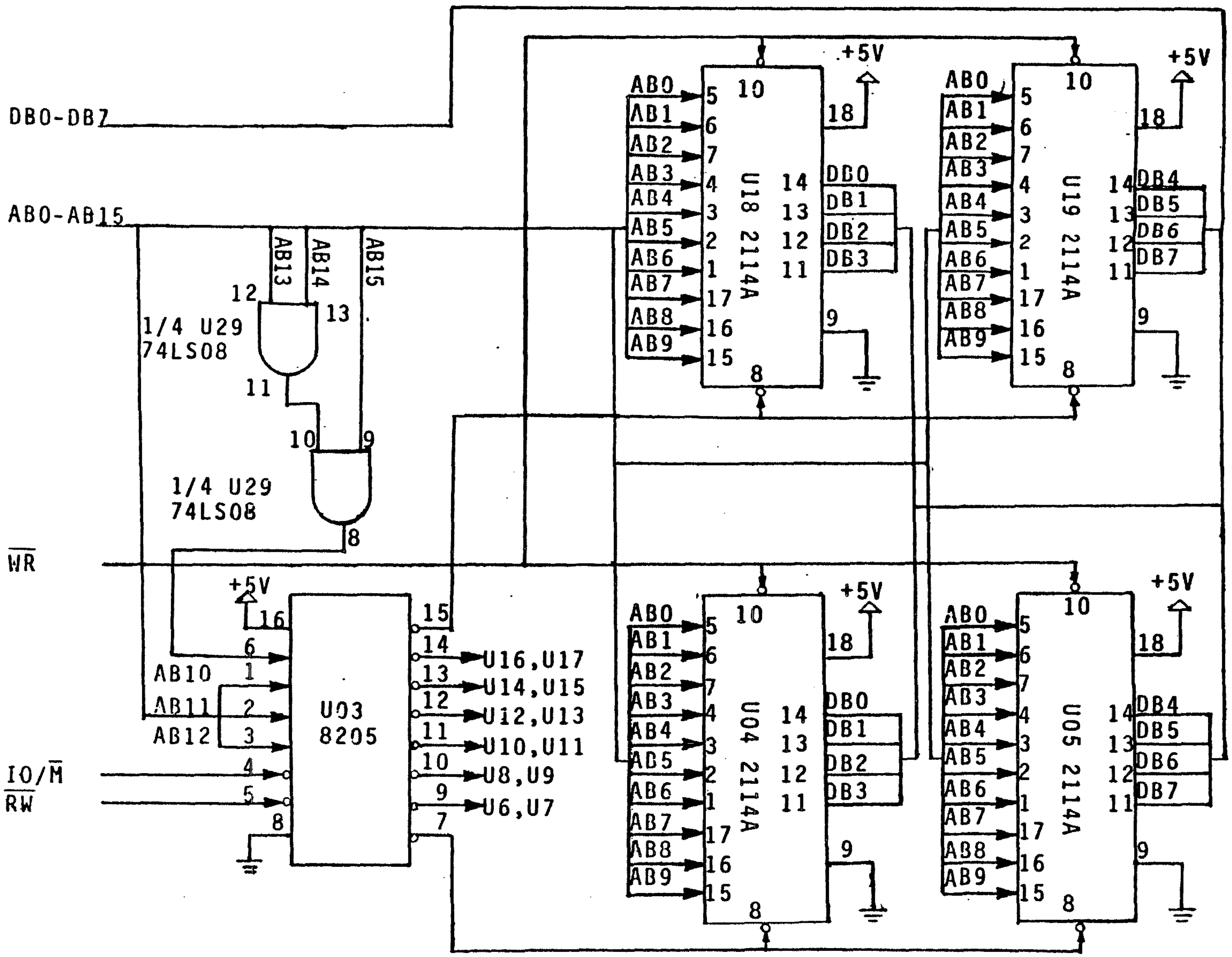


Figure The Circuit Diagram of the RAM

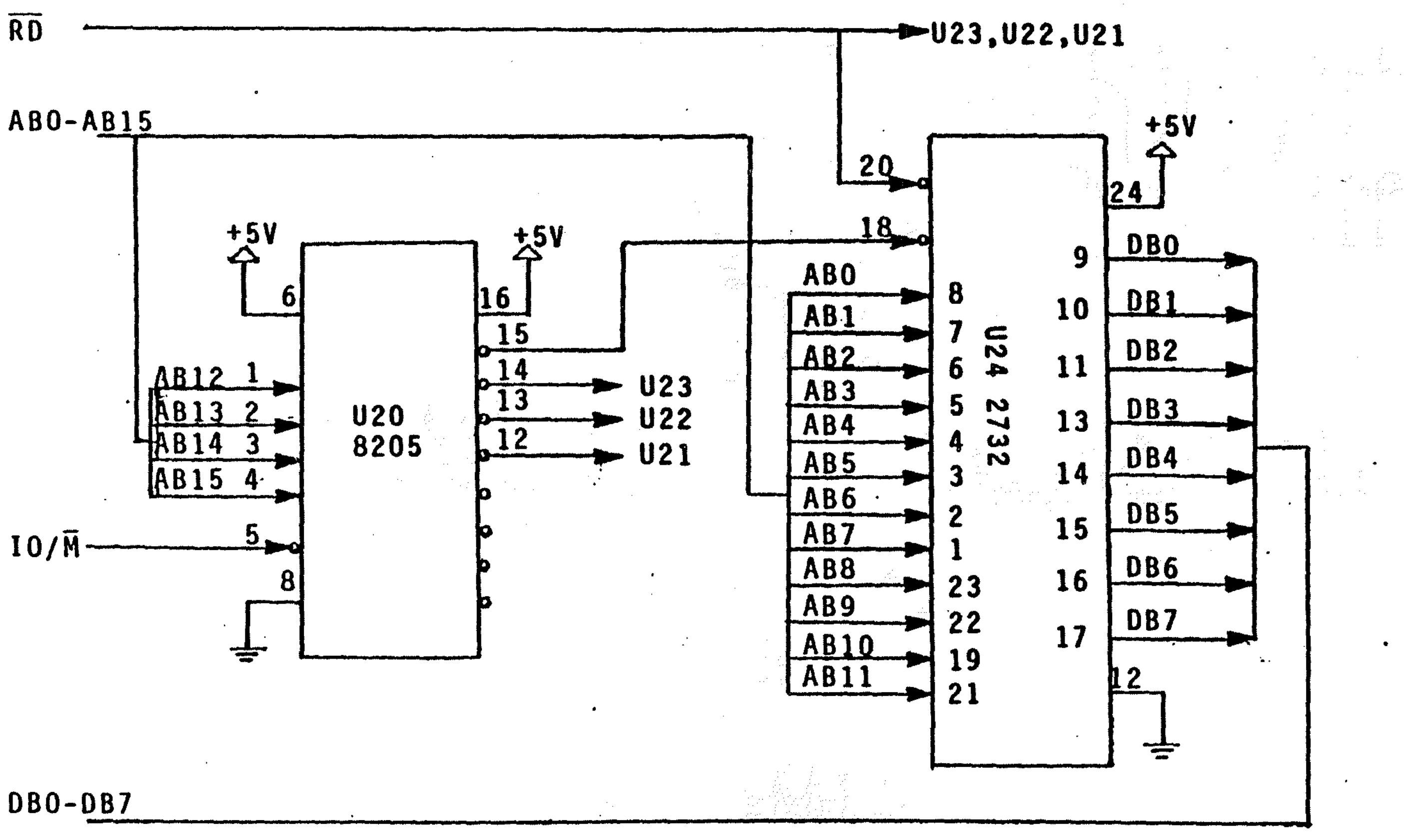


Figure The Circuit Diagram of the EPROM

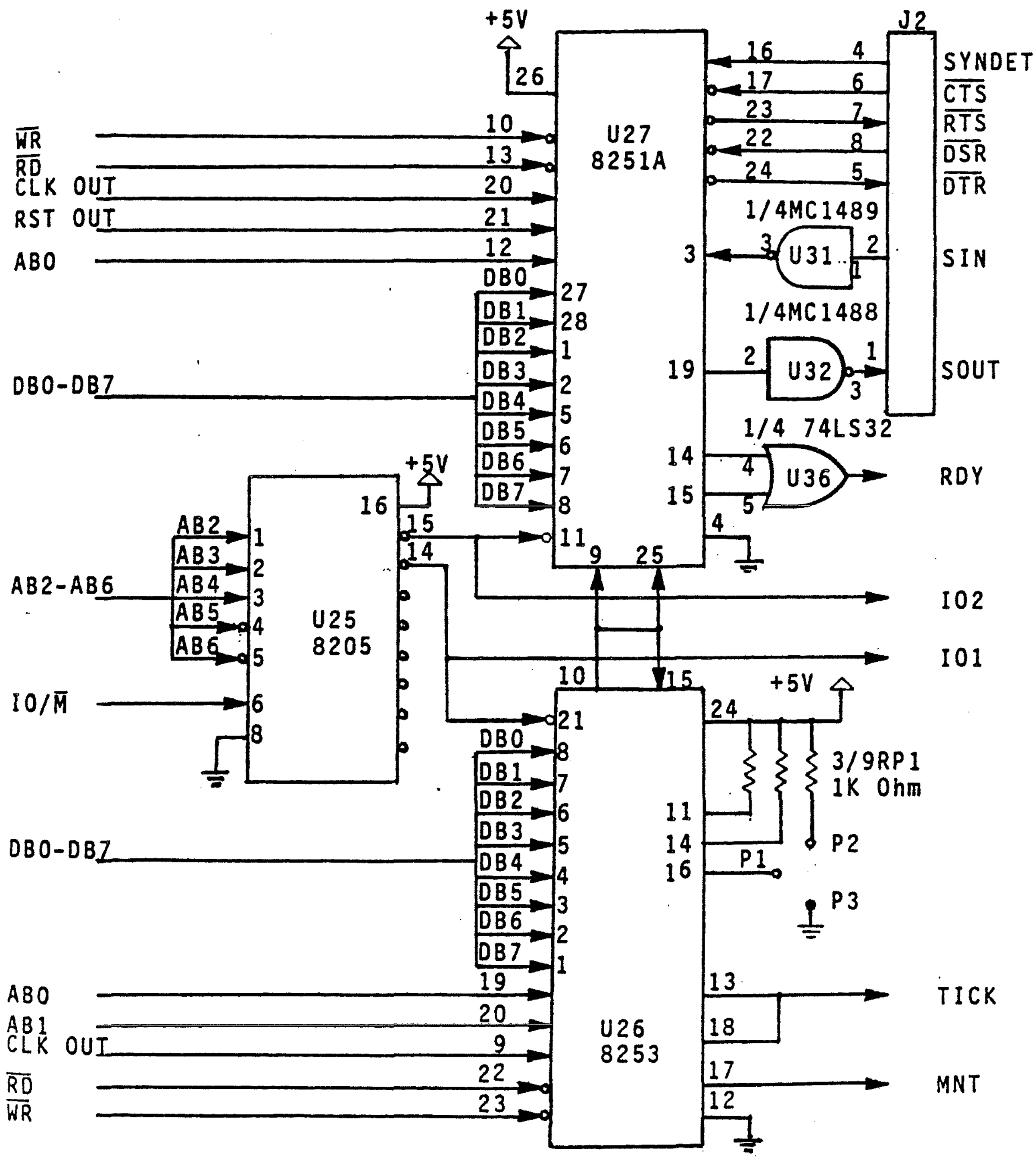


Figure   The Circuit Diagram of the I/O Interfaces

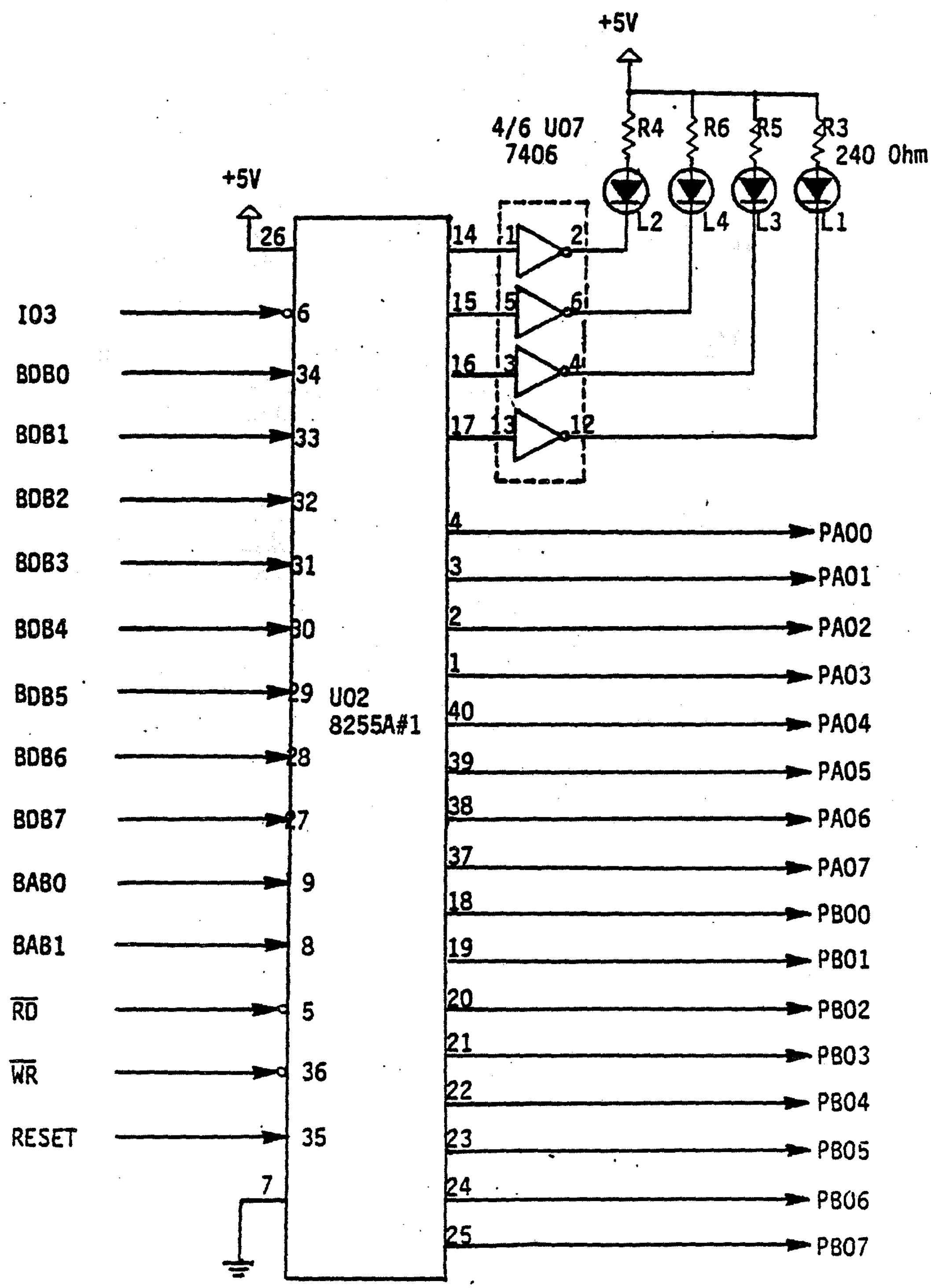


Figure The Circuit Diagram of the 8255A #1

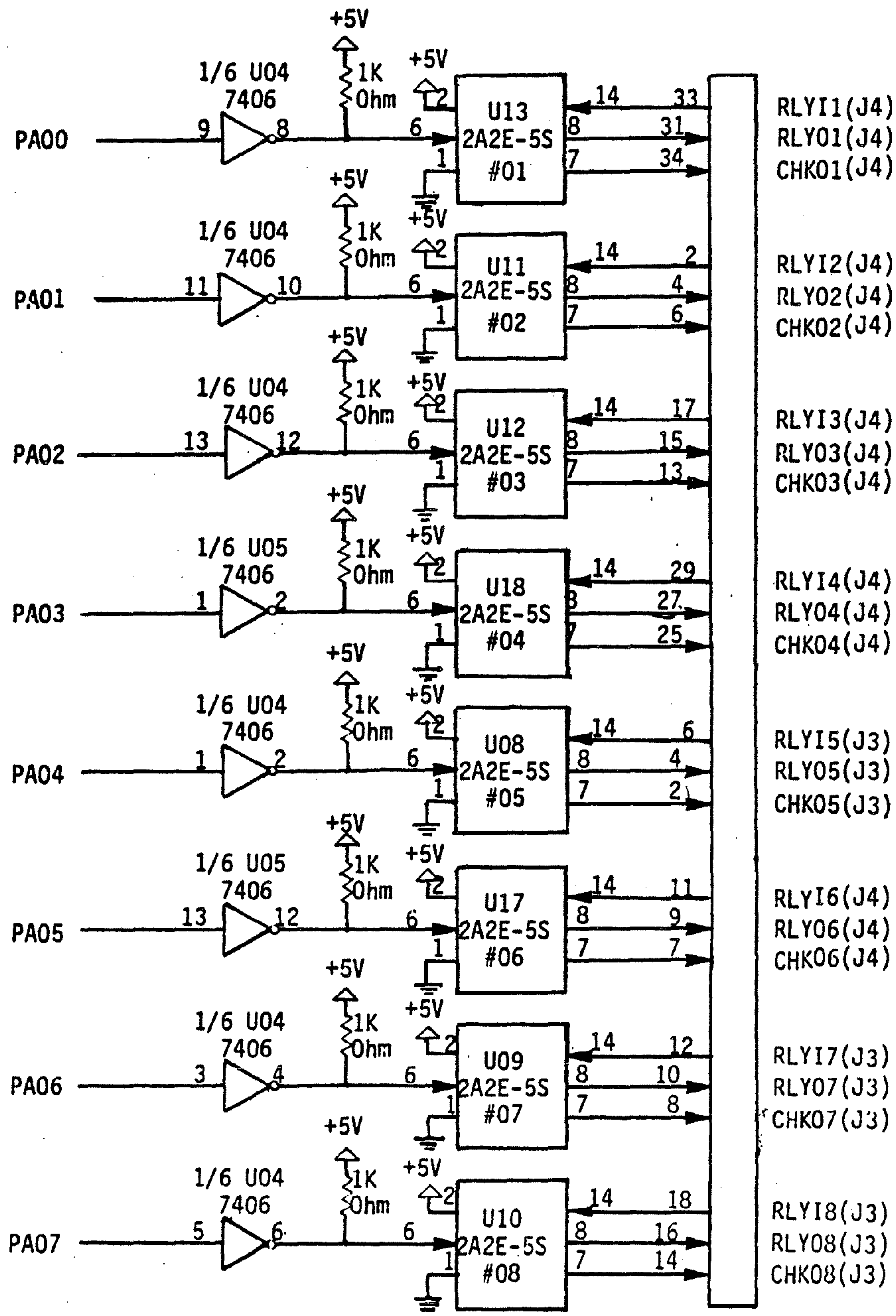


Figure The Circuit Diagram of the 8255A #1

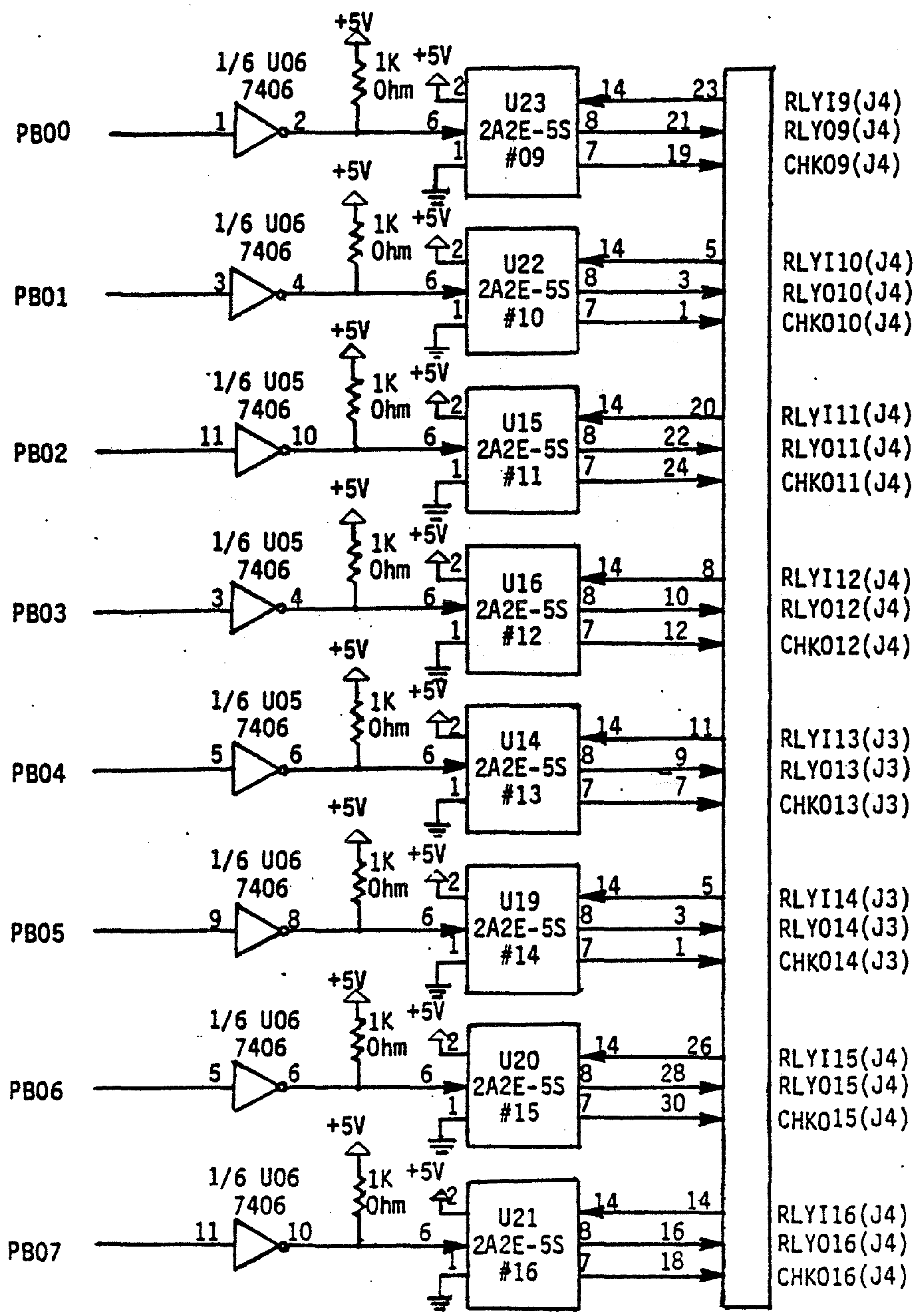


Figure The Circuit Diagram of the 8255A #1

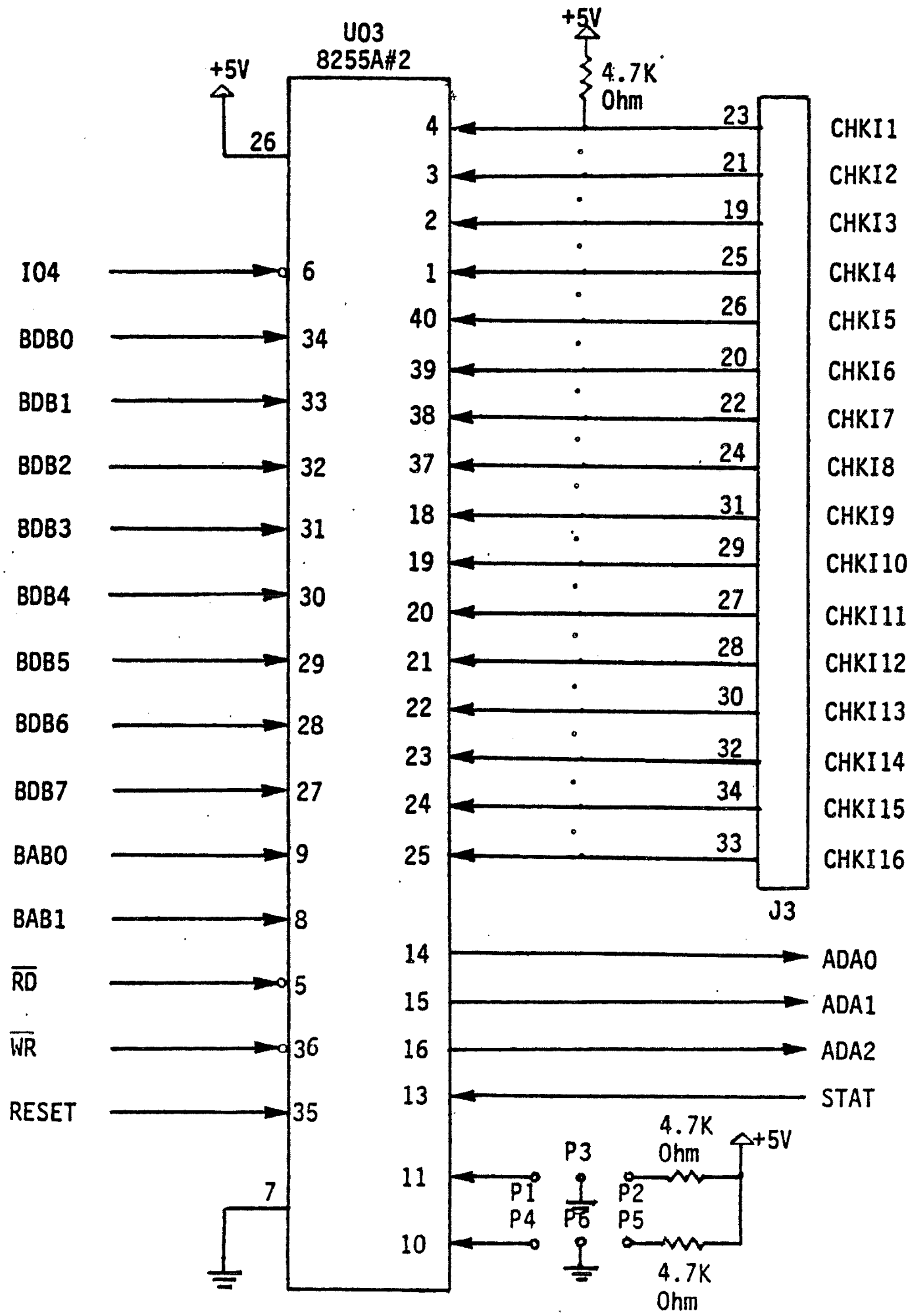


Figure The Circuit Diagram of the 8255A #2

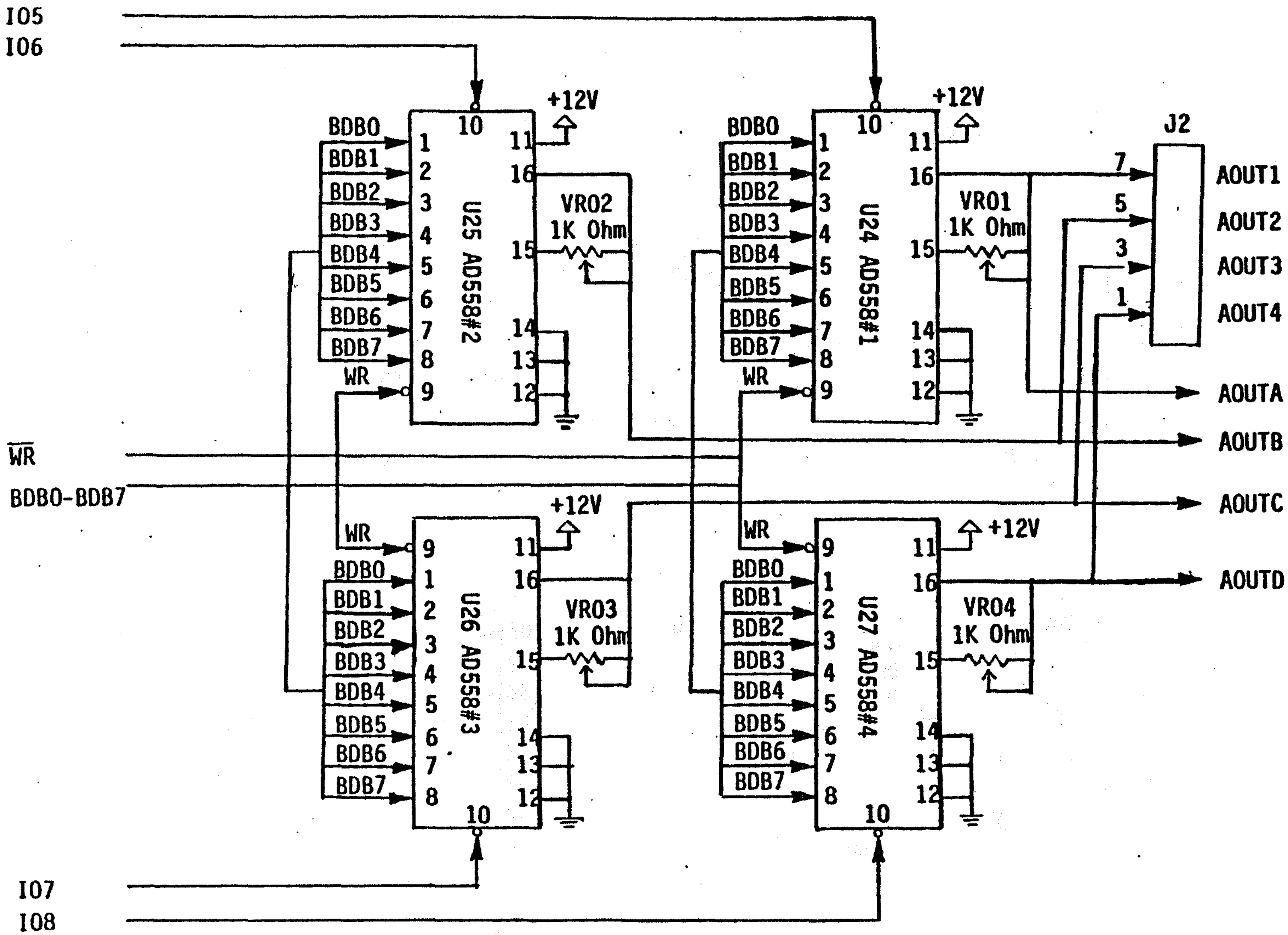


Figure The Circuit Diagram of the D/A Converters

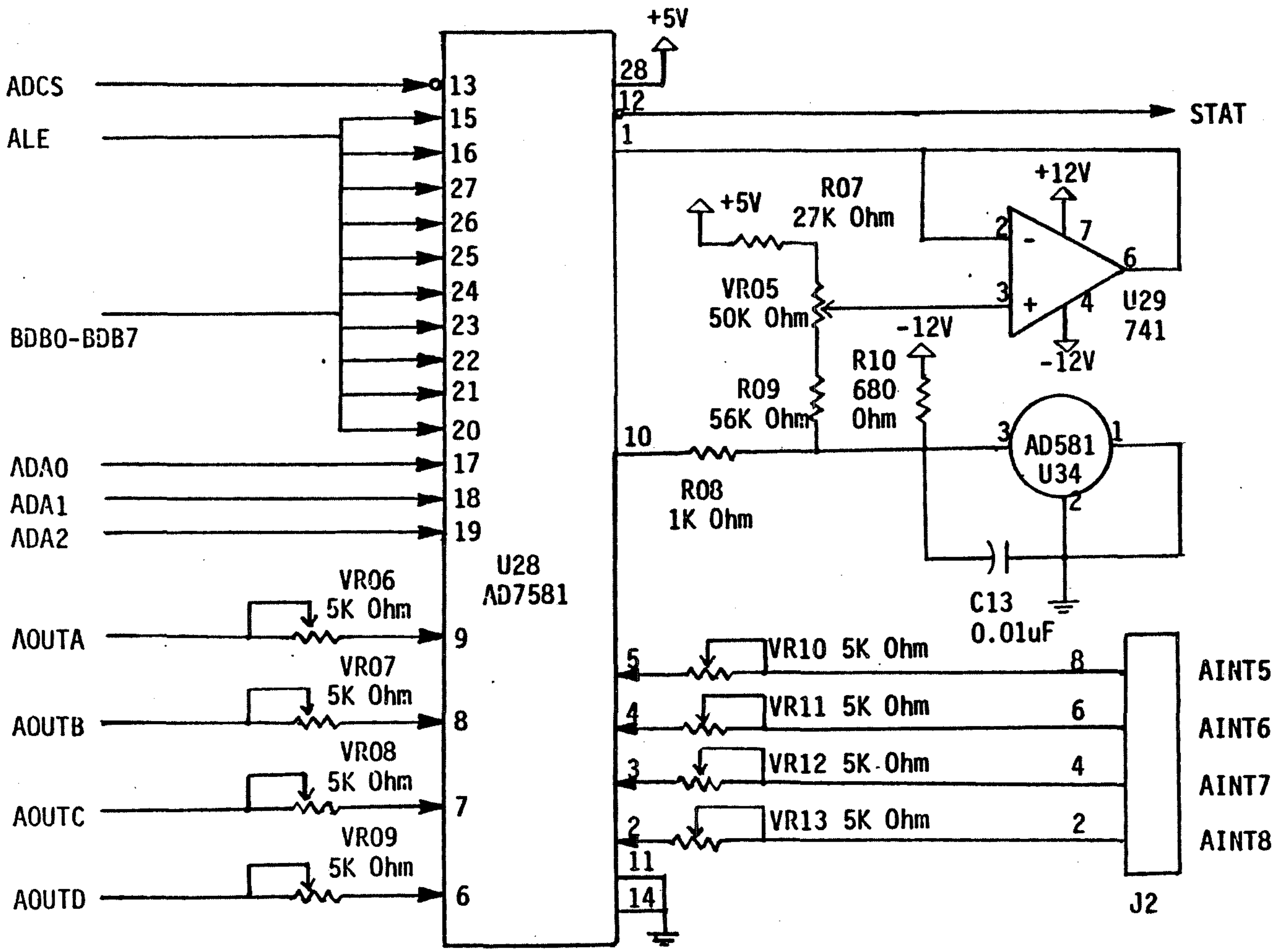


Figure The Circuit Diagram of the A/D Converter

APPENDIX A: THE CPU BOARD J1 CONNECTOR PIN LIST

pin no.	signal	function
1	PWRA	+5V Power
2	PWRB	+12V Power
3	PWRC	-12V Power
4-6	No Connection	
7	RESET OUT	Reset Output Control
8	WR	Write Control
9	RD	Read Conrol
10	IO/M	Input/Output or Memory Operation Control
11	ALE	Address Latch Enable Control
12	RESET IN	Reset Input Control
13-17	No Connection	
18	BDB0	Buffered Data Bus #0
19	BDB1	Buffered Data Bus #1
20	BDB2	Buffered Data Bus #2
21	BDB3	Buffered Data Bus #3
22	BDB4	Buffered Data Bus #4
23	BDB5	Buffered Data Bus #5
24	BDB6	Buffered Data Bus #6
25	BDB7	Buffered Data Bus #7
26-31	No Connection	
32	BAB0	Buffered Address Bus #0
33	BAB1	Buffered Address Bus #1
34	BAB2	Buffered Address Bus #2
35	BAB3	Buffered Address Bus #3
36	BAB4	Buffered Address Bus #4
37	BAB5	Buffered Address Bus #5
38	BAB6	Buffered Address Bus #6
39	BAB7	Buffered Address Bus #7
40-47	No Connection	
48	PWRA	+5V Power
49	PWRA	+5V Power
50-51	No Connection	
52	GND	Ground
53	GND	Ground
54-99	No Connection	
100	GND	Ground

APPENDIX B: THE CPU BOARD J2 CONNECTOR PIN LIST

pin no.	signal	function
1	SOUT	Serial Data Output
2	SIN	Serial Data Input
3	SYNC	Synchronization Signal Input
4	SYNDET	Synchronous Character Detection Input/Output
5	DTR	MODEM Data Terminal Ready Control
6	CTS	MODEM Clear To Send Control
7	RTS	MODEM Request To Send Control
8	DSR	MODEM Data Set Ready Control
9-10	No Connection	

APPENDIX B: THE CPU BOARD J2 CONNECTOR PIN LIST

pin no.	signal	function
1	SOUT	Serial Data Output
2	SIN	Serial Data Input
3	SYNC	Synchronization Signal Input
4	SYNDET	Synchronous Character Detection Input/Output
5	DTR	MODEM Data Terminal Ready Control
6	CTS	MODEM Clear To Send Control
7	RTS	MODEM Request To Send Control
8	DSR	MODEM Data Set Ready Control
9-10	No Connection	

APPENDIX C: THE CPU BOARD PARTS LIST

part no.	type	description	package	quantity
U01	8085A	8-bit Microprocessor	DIP-40	1
U02	8282	Octal Latch	DIP-20	1
U03	8205	RAM Decoder	DIP-16	1
U04-U19	2114A	1K*4 Static RAM	DIP-18	16
U20	8205	EPROM Decoder	DIP-16	1
U21-U24	2732	4K*8 EPROM	DIP-24	4
U25	8205	I/O Decoder	DIP-16	1
U26	8253	Programmable Interface Timer	DIP-24	1
U27	8251A	Programmable Communication Interface	DIP-28	1
U28	7407	Hex Buffers/Drivers	DIP-14	1
U29	74LS08	2-Input Positive AND Gates	DIP-14	1
U30	74LS00	2-Input Positive NAND Gates	DIP-14	1
U31	MC1489	Quad Line Receivers	DIP-14	1
U32	MC1488	Quad Line Drivers	DIP-14	1
U33	74LS244	Octal Buffers/Line Drivers	DIP-20	1
U34,U35	8216	Bidirectional Bus Driver	DIP-16	2
U36	74LS32	2-Input Positive OR Gates	DIP-14	1
U37	LM393	Dual Differential Comparators	DIP-8	1
XL01	---	4.0 MHz Crystal (HC-18U)	---	1
D01	1N4148	General Purpose Diode	---	1
D02	1N748	3.9V Zener Diode	---	1
RP1	---	1K Ohm Resitor Network (9 Resistors, 10 Pin SIP)	---	1
R01,R02	---	1K Ohm Carbon Film Resistor (1/4 W, 5%)	---	2
R03	---	100K Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
R04	---	22 Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
R05	---	680K Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
C01-C37	---	0.1 uF Ceramic Capacitor (25V, 20%)	---	37
C38	---	uF Tantalum Capacitor	---	1
T01-T04	---	IC Socket (24 pin)	---	4
P01-P06	---	Square Pin	---	6
J02	---	Jumper Header (10 Postr)	---	

APPENDIX D: THE I/O BOARD J1 CONNECTOR PIN LIST

pin no.	signal	function
1	PWRA	+5V Power
2	PWRB	+12V Power
3	PWRC	-12V Power
4-6	No Connection	
7	RESET OUT	Reset Output Control
8	WR	Write Control
9	RD	Read Control
10	IO/M	Input/Output or Memory Operation Control
11	ALE	Address Latch Enable Control
12-17	No Connection	
18	BDB0	Buffered Data Bus #0
19	BDB1	Buffered Data Bus #1
20	BDB2	Buffered Data Bus #2
21	BDB3	Buffered Data Bus #3
22	BDB4	Buffered Data Bus #4
23	BDB5	Buffered Data Bus #5
24	BDB6	Buffered Data Bus #6
25	BDB7	Buffered Data Bus #7
26-31	No Connection	
32	BAB0	Buffered Address Bus #0
33	BAB1	Buffered Address Bus #1
34	BAB2	Buffered Address Bus #2
35	BAB3	Buffered Address Bus #3
36	BAB4	Buffered Address Bus #4
37	BAB5	Buffered Address Bus #5
38	BAB6	Buffered Address Bus #6
39	BAB7	Buffered Address Bus #7
40-47	No Connection	
48	PWRA	+5V Power
49	PWRA	+5V Power
50-51	No Connection	
52	GND	Ground
53	GND	Ground
54-99	No Connection	
100	GND	Ground

APPENDIX E: THE I/O BOARD J2 CONNECTOR PIN LIST

pin no.	signal	function
1	AOUT4	AD558#4 Analog Output
2	AINT4	External Analog Input #4
3	AOUT3	AD558#3 Analog Output
4	AINT3	External Analog Input #3
5	AOUT2	AD558#2 Analog Output
6	AINT2	External Analog Input #2
7	AOUT1	AD558#1 Analog Output
8	AINT1	External Analog Input #1
9-10	No Connection	

APPENDIX F: THE I/O BOARD J3 CONNECTOR PIN LIST F-1

pin no.	signal	function
1	CHKO14	Relay #14 Check Output
2	CHKO5	Relay #5 Check Output
3	RLYO14	Relay #14 Output
4	RLYO5	Relay #5 Output
5	RLYI14	Relay #14 Input
6	RLYI5	Relay #5 Input
7	CHKO13	Relay #13 Check Output
8	CHKO7	Relay #7 Check Output
9	RLYO13	Relay #13 Output
10	RLYO7	Relay #7 Output
11	RLYI13	Relay #13 Input
12	RLYI7	Relay #7 Input
13	No Connection	
14	CHKO8	Relay #8 Check Output
15	No Connection	
16	RLYO8	Relay #8 Output
17	No Connection	
18	RLYI8	Relay #8 Input
19	CHKI3	External Relay Check Input #3
20	CHKI6	External Relay Check Input #6
21	CHKI2	External Relay Check Input #2
22	CHKI7	External Relay Check Input #7
23	CHKI1	External Relay Check Input #1
24	CHKI8	External Relay Check Input #8
25	CHKI4	External Relay Check Input #4
26	CHKI5	External Relay Check Input #5
27	CHKI11	External Relay Check Input #11
28	CHKI12	External Relay Check Input #12
29	CHKI10	External Relay Check Input #10
30	CHKI13	External Relay Check Input #13
31	CHKI9	External Relay Check Input #9
32	CHKI14	External Relay Check Input #14
33	CHKI16	External Relay Check Input #16
34	CHKI15	External Relay Check Input #15

APPENDIX G: THE I/O BOARD J4 CONNECTOR PIN LIST G-1

pin no.	signal	function
1	CHK010	Relay #10 Check Output
2	RLYI2	Relay #2 Input
3	RLYO10	Relay #10 Output
4	RLYO2	Relay #2 Output
5	RLYI10	Relay #10 Input
6	CHKO2	Relay #2 Check Output
7	CHKO6	Relay #6 Check Output
8	RLYI12	Relay #12 Iutput
9	RLYO6	Relay #6 Output
10	RLYO12	Relay #12 Output
11	RLYI6	Relay #6 Input
12	CHKO12	Relay #12 Check Output
13	CHKO3	Relay #3 Check Output
14	RLYI16	Relay #16 Input
15	RLYO3	Relay #3 Output
16	RLYO16	Relay #16 Output
17	RLYI3	Relay #3 Input
18	CHKO16	Relay #16 Check Output
19	CHKO9	Relay #9 Check Output
20	RLYI11	Relay #11 Input
21	RLYO9	Relay #9 Output
22	RLYO11	Relay #11 Output
23	RLYI9	Relay #9 Input
24	CHKO11	Relay #11 Check Output
25	CHKO4	Relay #4 Check Output
26	RLYI15	Relay #15 Input
27	RLYO4	Relay #4 Output
28	RLYO15	Relay #15 Output
29	RLYI4	Relay #4 Input
30	CHKO15	Relay #15 Check Output
31	RLYO1	Relay #1 Output
32	No Connection	
33	RLYI1	Relay #1 Input
34	CHKO1	Relay #1 Check Output

APPENDIX H: THE I/O BOARD PARTS LIST

part no.	type	description	package	quantity
U01	8205	I/O Decoder	DIP-16	1
U02,U03	8255A	Programmable Peripheral Interface	DIP-40	2
U04-U07	7406	Hex Inverter Buffers/Drivers	DIP-14	4
U08-U23	2A2E-5S	Reed Relay	DIP-14	16
U24-U27	AD558	8-Bit Digital-To-Analog Converter	DIP-16	4
U28	AD7581	8-Channel 8-Bit Analog-To-Digital Converter	DIP-28	1
U29	741	General Purpose Operational Amplifier	DIP-8	1
U30	AD581	10 Volt IC Reference	TO-5	1
U31	74LS00	2-Input Positive NAND Gates	DIP-14	1
RP1,RP2	---	4.7K Ohm Resistor Network (9 Resistors, 10 Pin SIP)	---	2
RP3,RP4	---	1K Ohm Resistor Network (9 Resistors, 10 Pin SIP)	---	2
VR01-VR04	---	1K Ohm Potentiometer (PC Pins, 0.2W, 5%)	---	4
VR05	---	50K Ohm Potentiometer (PC Pins, 0.2W, 5%)	---	1
VR06-VR13	---	5K Ohm Potentiometer (PC Pins, 0.2W, 5%)	---	8
R01,R02	---	1K Ohm Carbon Film Resistor (1/4 W, 5%)	---	2
R03-R06	---	240 Ohm Carbon Film Resistor (1/4 W, 5%)	---	4
R07	---	27K Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
R08	---	1K Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
R09	---	56K Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
R10	---	680 Ohm Carbon Film Resistor (1/4 W, 5%)	---	1
C01-C14	---	0.1uF Ceramic Capacitor (25V, 20%)	---	14
C15	---	0.01uF Ceramic Capacitor (25V, 20%)	---	1
L1-L4	---	LED Indicator (0.2 DIA. Red)	---	4
P01-P06	---	Square Pin	---	6
J02	---	Jumper Header (10 Posts)	---	1
J03,J04	---	Jumper Header (34 Posts)	---	2

3.1.2 SEM Software User's Manual, Version 3.0

INTENDED AUDIENCE

This manual is intended to be used by those people who have little interest in the actual software of the system, but will be users of the system. This manual is for the SEM user. It is the intent of the author that this manual not get bogged down in details but give a general overview.

Those interested in the details are referred to C.A. Winkelmann's thesis entitled, "A Real-Time, Structured Software Package for the WVU-USBM Sensor Emulator" available from the Evansdale Library of the WVU Library System.

SYSTEM INTRODUCTION

The Sensor-Emulator (SEM)

The SEM is an 8085A based system that is capable of operating in several modes and providing four analog outputs and two eight bit digital outputs. The SEM software is interrupt driven and provides the user with several operational options. The most important of these options are those that are hardware selectable. A brief description of each one of them and what they do can be found in this section.

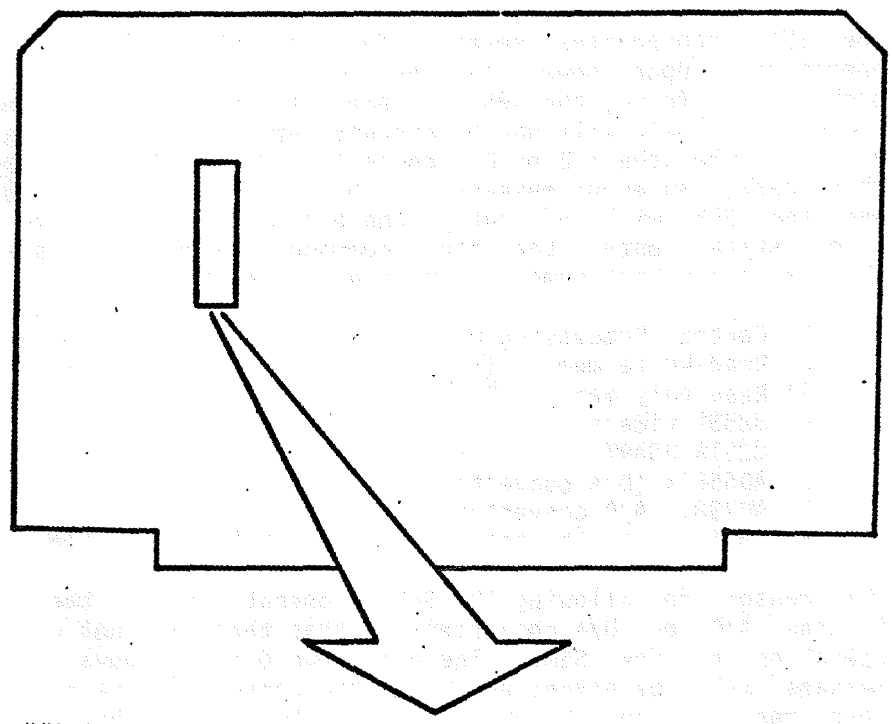
Communication Mode Selection

The SEM has two modes of communication. The first is terminal mode in which the SEM communicates with a terminal and an operator gives the SEM commands.

The second mode of operation is called line mode, in which the SEM communicates with a supervisory computer called the Dynamic Test Controller (DTC) and communicates via the DDCMP (Digital Data Communication Message Protocol) format. Information on the DDCMP formats can be found in Mine Wide Test of the WVU Monitoring Concept .[1] The mode of operation is wire-wrap selectable on the SEM CPU board. Figure 3.1.19 shows the proper location of the pins for wire wrap selection.

SYSTEM START-UP

The SEM is a relatively simple system to start up; just turn on the power. Hardware will take care of giving the reset to the processor. If the SEM is connected to the DTC, then the operator only needs to ensure that no fault lights on the front panel of the SEM are lit. If the SEM is to be connected to a terminal there will be a message printed at the time the diagnostics are finished and the system is ready to accept commands. If anything other than this happens, refer to the section in this manual on troubleshooting.



JUMPER CONFIG.	MODE FUNCTION
P1 - P2	MONITOR
P1 - P3	NONMONITOR
P4 - P5	ANALOG
P4 - P6	DIGITAL

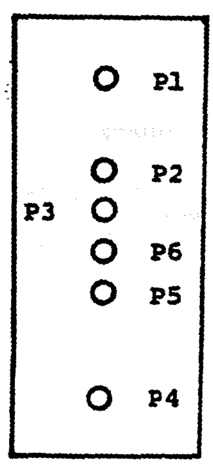


Fig. 3.1.19 Mode Selection

SYSTEM DIAGNOSTICS

The SEM incorporates several features that may halt normal SEM operation. Upon power on the SEM runs self-diagnostics and if a problem is found, the SEM will execute a halt instruction. The only time that a halt will not be executed upon the diagnostics finding an error is when the A/D or D/A converters do not function properly. In this case, an error message will be printed after the header message and the SEM will be put in the error state (to remove it from the error state, enter the abort command). Shown here is a summary of the components that comprise the diagnostics tests:

1) Central Processing Unit (CPU);
2) Read-Write memory (RAM);
3) Read only memory (ROM);
4) 8253A timers;
5) 8251A USART;
6) AD558's (D/A converters);
7) AD7581 (A/D converter);
8) implicit to the test is that the 8255A's function properly.

The reason for allowing the SEM to operate even if there is an error in the A/D or D/A converters is that these are not critical to the operation of the SEM. There are two possible ways that this error message will be given; one is if the converters are actually bad and need replaced and the other is if either one of the converters needs calibrated.

SEM COMMANDS

The Keyboard and Terminal Operations

Terminals vary greatly and, thus, this description of the special function keys that the SEM will respond to will be very general. You should remember that the terminal is not a typewriter but actually transmits information to the SEM. In normal operation it will appear as if you are actually typing on the screen; however, this is not the case. When you type a character it is sent to the SEM and the SEM echos the character back to the terminal. The echoing occurs so fast that it appears as if you are typing on your terminal.

Control Characters

When typing commands at the terminal, there are several keys that can be used to correct typing errors (which will be covered in this section). A control character is sent by holding the control key on the keyboard down while pressing the key with the appropriate letter on it.

DELETE vs. BACKSPACE vs. CNTRL-H vs. RUBOUT: You should avoid use of any key to delete characters other than the RUBOUT key. Other keys may appear as though they delete the text, but they do not. The RUBOUT key can be used as many times as needed to correct spelling mistakes in the command line.

CONTROL-U: This command is given to delete the entire command line. The actual line that was typed will remain on the screen, but the SEM will ignore it and give the system prompt again. This command is given by pressing the CONTROL key and the U at the same time.

CONTROL-C: A CONTROL-C will abort the current output to the terminal. It is used when the DUMP command has been given and you want to abort it. The SEM will abort the DUMP command's action and give the system prompt.

CONTROL-S and CONTROL-Q: Issuing a CONTROL-S will stop any listing to the terminal. This feature is especially useful when doing a memory dump and you wish to look at specific locations in memory. The listing will be temporarily halted at the terminal until a CONTROL-Q is issued. CONTROL-Q will restart the listing at exactly the place where it was suspended by the CONTROL-S.

Upper Case Versus Lower Case

The SEM will only respond to commands issued to it in upper case. On most keyboards there is a key marked CAPS LOCK. When this key is pressed, the terminal will only transmit upper case letters; all other characters typed on the keyboard will be transmitted normally. Note that it is not a shift lock. For example:

> With the CAPS LOCK pressed typing a w will transmit an upper case W. However, with the CAPS LOCK pressed typing a 2 will transmit a 2 and not an @. To transmit the @ with the CAPS LOCK pressed still requires you to type SHIFT 2.

Operating Commands

The commands and command descriptions that follow are used when the SEM operator is entering the commands from a terminal (terminal mode). These commands tell the SEM what it is to do.

The only data that the SEM is capable of giving when operated in this fashion is a triangular wave on each analog output that has a peak value of 5.0 volts and a frequency given by the equation:

$$f=1/(160*scalefactor*tickrate/1000) \text{ Hz}$$

and the digital data given at each digital port is given by:

00000001 Rotated Left one position every time the analog signal changes slope.

Command Abbreviations

To enter the following commands, the entire word need not be given. The only requirement is that there be enough letters entered to distinguish one command from another. The following table shows, in capital letters, what part of the word is necessary for the command to be implemented.

Abort	SHow
Dump	STATus
Restart	STARt
SEt	

When a command has a parameter associated with it, the parameters need only be specified by enough letters to make them unique also. For example TICK (for the set and show commands) can be given by TIC.

ABORT COMMAND: This command aborts the current test or moves the SEM from any state to the state it was in before the START command was used (i.e. WAIT$START). This command and the RESET command are the only ones that can recover the SEM from an error. Upon use of the ABORT command, the parameters that can be set by the SET command are returned to their default values.

DUMP COMMAND: The DUMP command dumps the contents of Read-Write Memory (RAM). The CONTROL-S, CONTROL-Q, and CONTROL-C entries are used in conjunction with this command.

RESET COMMAND: This command forces the SEM to begin program execution at location zero. It has the same effect as pressing the reset button on the front panel of the SEM or turning power off then on again. Remember that the diagnostics are run and the contents of RAM are written with zeros.

SET COMMAND: This command is used to set the parameters for the current test. The parameters that can be set are:

1. TICK-sets the tick rate used for the test;
2. CHECK-puts the SEM in a mode to check data;
3. TYPE-configures the SEM to output data, check data, or do both;

The syntax of the particular set commands are:

1. SET TICK=value [MILLISECONDS, SECONDS, MINUTES] where the value for milliseconds can be between 1 and 1000, the value for seconds can be between 1 and 120, and the value for minutes can be between 1 and 10;

2. SET CHECK [SELF, PARTNER, BOTH, NONE] where SELF tells the SEM to check its own data against the data sent by the DTC, PARTNER tells the SEM to check its partner SEM's data against the data sent by the DTC, BOTH tells it to check its own data and the data from the partner SEM, and NONE tells the SEM not to check any data at all. This command is used in conjunction with the SET TYPE command. This command is only used in line mode of operation. Several set up configurations are not allowed and the SEM will respond after the START command is issued if the set up is not valid.

3. SET TYPE [OUTPUT, CHECK, BOTH] where OUTPUT tells the SEM it is only to output data and not check any data, CHECK tells the SEM it is only used to check data and not output any data, BOTH tells the SEM it is to output data and check data. This command is used in conjunction with the SET CHECK command. This command is only used in line mode operation. Several set up configurations are not valid and the SEM will respond after the START command is issued if the set up is wrong.

Invalid Set Ups: The SEM will not allow itself to be set up incorrectly. The combinations of the SET TYPE and SET CHECK commands are as follows. If the TYPE is OUTPUT then CHECK cannot be SELF, PARTNER, or BOTH; it must be NONE. If the TYPE is CHECK or BOTH then CHECK cannot be NONE; it must be SELF, PARTNER, or BOTH. If any of these invalid set ups are entered, the SEM will not start when the START command is issued but will print an error message and display the invalid set up options.

SHOW COMMAND: This command is used to display on the terminal the current parameter asked for. The parameters that can be used with this command are:

1. TICK-displays the current tick setting;
2. CHECK-displays the current check mode;
3. TYPE-displays the current SEM type.

STATUS COMMAND: This command will display the current state of the SEM and will also display the current interrupt mask. If the SEM is in the error state, then the state it was in when the error occurred is displayed along with the current state and the tick count when the error occurred. To calculate the actual tick count, the first number displayed should be converted from hex to decimal and multiplied by ten thousand (10,000) and the second number should be converted from hex to decimal and added to the results obtained from the multiplication.

START COMMAND: This command starts a test with the current parameters given by the SET command or, if no parameters are given, the default values are used. The default parameters for the test are:

TICK=1000 milliseconds
TYPE=OUTPUT
CHECK=NONE

SEM ERRORS

The errors that the SEM can encounter are placed into two classifications: fatal errors are those that have no error messages and halt the processor; nonfatal errors are those that move the SEM into the error state and stop the current test. These two error types will be examined in this chapter.

Fatal Errors

Those errors which are classified as being fatal are:

1) CPU fault;
2) RAM fault;
3) ROM CRC error:
4) 8253A timer fault;
5) 8251A USART fault;
6) software wild-execution;
7) infinite software looping.

These types of errors require a technician to locate the cause of the error and correct the problem. They are usually fixed by component replacement.

Nonfatal Errors

The errors that are classified as being nonfatal identify themselves by printing an error number on the operator's terminal. The error message will appear as:

ERROR xx

where xx is the error number.

The error numbers represent the following errors.

1 = unexpected START command received
2 = SEM missed a TICK interrupt
3 = software time-out
4 = wild execution
5 = unexpected SYNC interrupt received
6 = target system data exhausted early
7 = data buffer dequeued early
8 = unexpected TICK interrupt received
9 = A/D or D/A conversion error

10 = sequence data error on self check
11 = sequence data error on partner check
12 = individual data error on self check
13 = individual data error on partner check
14 = attempt to get check buffer from empty list
15 = attempt to return check buffer to a full list
16 = reserved-undefined

OEEH = unavailable for definition-reserved for system use

TROUBLESHOOTING

This section will deal with common problems that may be encountered. These problems will not be in-depth problems, but will be typical forgetful problems such as forgetting communication cables and so forth. These problems may seem trivial to some, but are a cause of great frustration when they occur unexpectedly.

This section is in outline form and will have the problem that could be encountered followed by possible solutions indented from the margin.

Troubleshooters Guide (Operators)

1) SEM power is on but SEM's do not respond:

a) power supply not plugged in;
b) no power cable to SEM;
c) bad power cable to SEM;
d) blown fuse in SEM;
e) no SEM CPU board in box;
f) diagnostics found an error (in this case a fault light should be on to indicate this);

2) SEM powers up, no fault lights on, no response:

a) missing communication cable from terminal to SEM;
b) bad communication cable from terminal to SEM;
c) ribbon cable from box to SEM CPU board not connected;
d) bad connection inside SEM box;
e) terminal in wrong mode i.e. terminal in local mode rather than remote (on line) mode;
f) terminal not turned on;
g) "power on reset" circuit problem;
h) SEM missing CPU board;

3) SEM powers up, no fault light, no analog data:

a) SEM missing I/O board;
b) missing coaxial cables from SEM to target system;
c) bad coaxial cables used;
d) faulty indicators, sensors, meters or device attached to target system end of coaxial cable;
e) A/D or D/A converter failure during use (rerun diagnostics using RESET command);
f) coaxial cable connected to wrong connectors on SEM back panel;

4) SEM powers up, no fault lights on, no digital data:

a) SEM missing I/O board;
b) missing ribbon cable from SEM to target system;
c) bad ribbon cable used;
d) faulty relay on SEM I/O board;
e) faulty indicator, sensor, meter, or device attached to target system end of ribbon cable;
f) ribbon cable not plugged in on SEM I/O board;
g) ribbon cable plugged into wrong connector on SEM I/O board;

5) SEM powers up, fault light on:

a) send to technician;

6) SEM powers up, no fault lights, garbage printed at the terminal:

   a) terminal baud set to wrong value-set baud to 1200;

3.1.3 SEM Operational Configurations

The SEM can be put into many operational configurations. These configurations depend upon the TYPE and CHECK parameters of the SET command. The operational configurations that are recommended for use are presented in Figure 3.1.20a-d. Figures 3.1.20a and 3.1.20b show the possible SEM configurations when all 21 SEM's need to be used for a test. If only ten SEM's are needed for a test, then the SEM's can be used in pairs. In the configurations shown in Figures 3.1.20c and 3.1.20d, the partner SEM is used to check data and not to output data. The set check and set type parameters for the configuraions are shown for each SEM.

Other configurations for SEM pairing are possible, but due to software limitations they are useless becaue each SEM contains two DDCMP station addresses. One station address is for data and the other is for commands. One SEM cannot receive another SEM's data unless the SEM software is modified to add another DDCMP station to each SEM. This new station must have its own station address. Since this change has not been made, if an SEM is going to be used to check data (such as SEM B in Figures 3.1.20c and 3.1.20d), it must be sent the same data that is sent to another SEM (SEM A in Figures 3.1.20c and 3.1.20d). This means that two SEM's contain the same data, which prohibits the SEM used for data checking from being used to output data since one SEM is already sending that data to the SUT.

3.1.4 The SEM as a State Machine

The SEM software is an implementation of the state machine described by the state diagram in Figure 3.1.21. The SEM begins its operation on power up by executing diagnostics and is in the diagnostics state. Successful completion of the diagnostics will move the SEM into the initialization state. In the initialization state, the programmable integrated circuits used in the SEM are programmed and all test variables are set to their pretest values. In this state the SEM determines if it is to operate in terminal mode or line mode. In terminal mode, the SEM is operated through a terminal by an operator and will output a triangular waveform on the analog outputs and rotate a bit across the digital outputs. This mode is used mostly for debugging hardware. The SEM is in line mode when it is used to accept buffers of data from the DTC. The mode of operation is selected by a wire wrap option on the SEM CPU board that either ties the SID (Serial Input) line on the 8085 high or low. A logical '1' on the SID line indicates terminal mode and a logical '0' indicates line mode. The mode is checked in the procedure

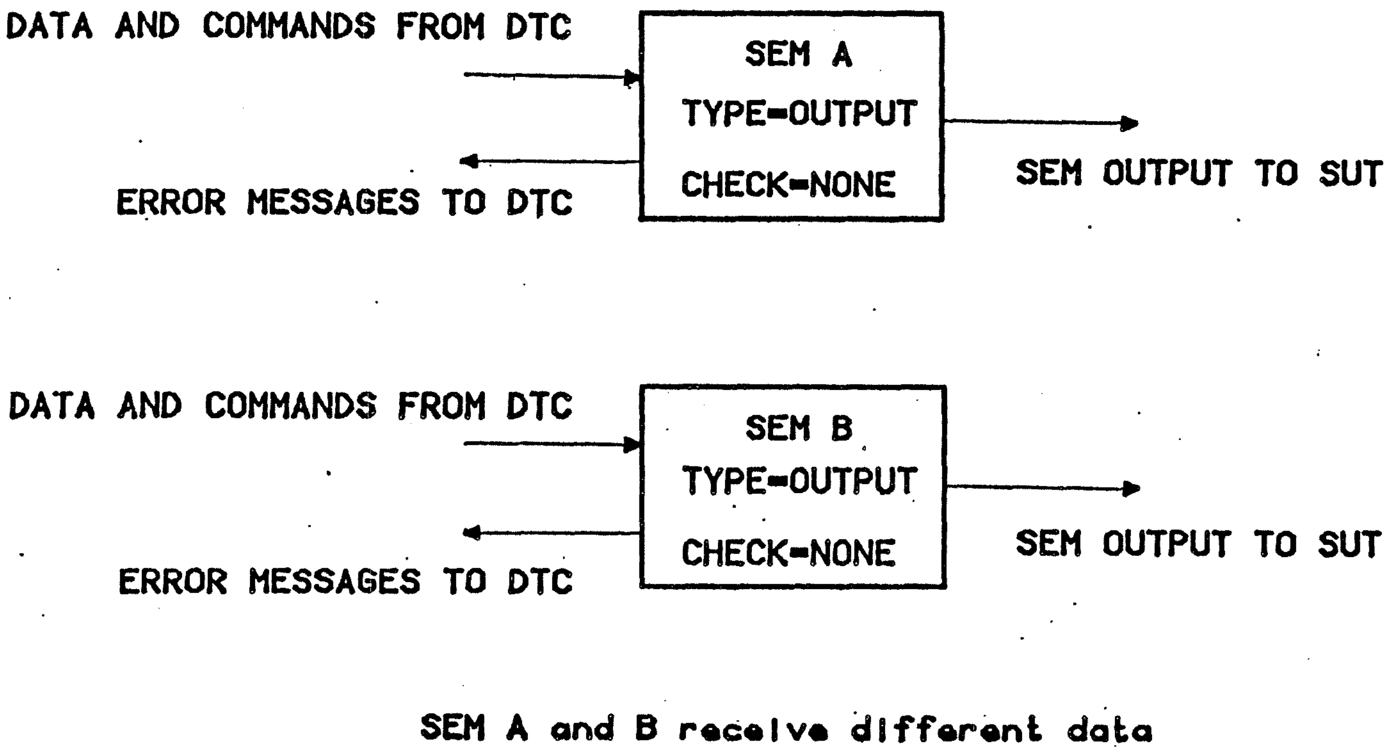
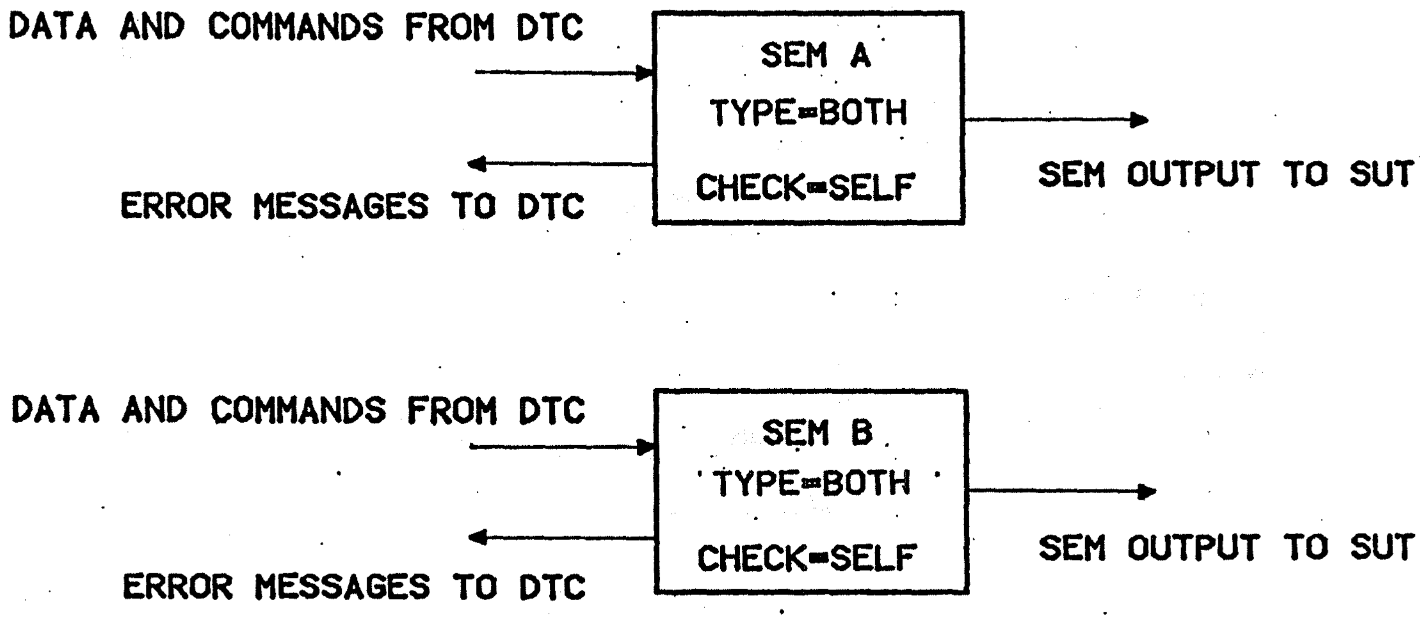
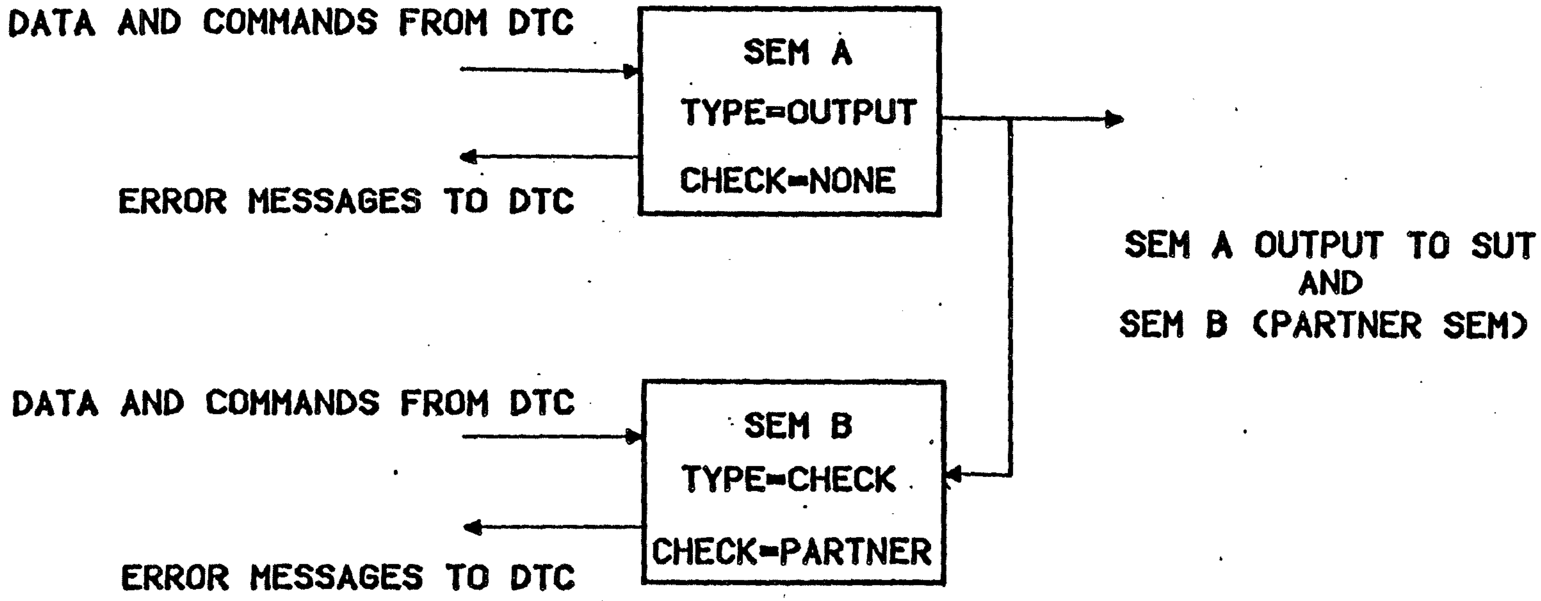


Fig. 3.1.20a SEM Operational Configuration



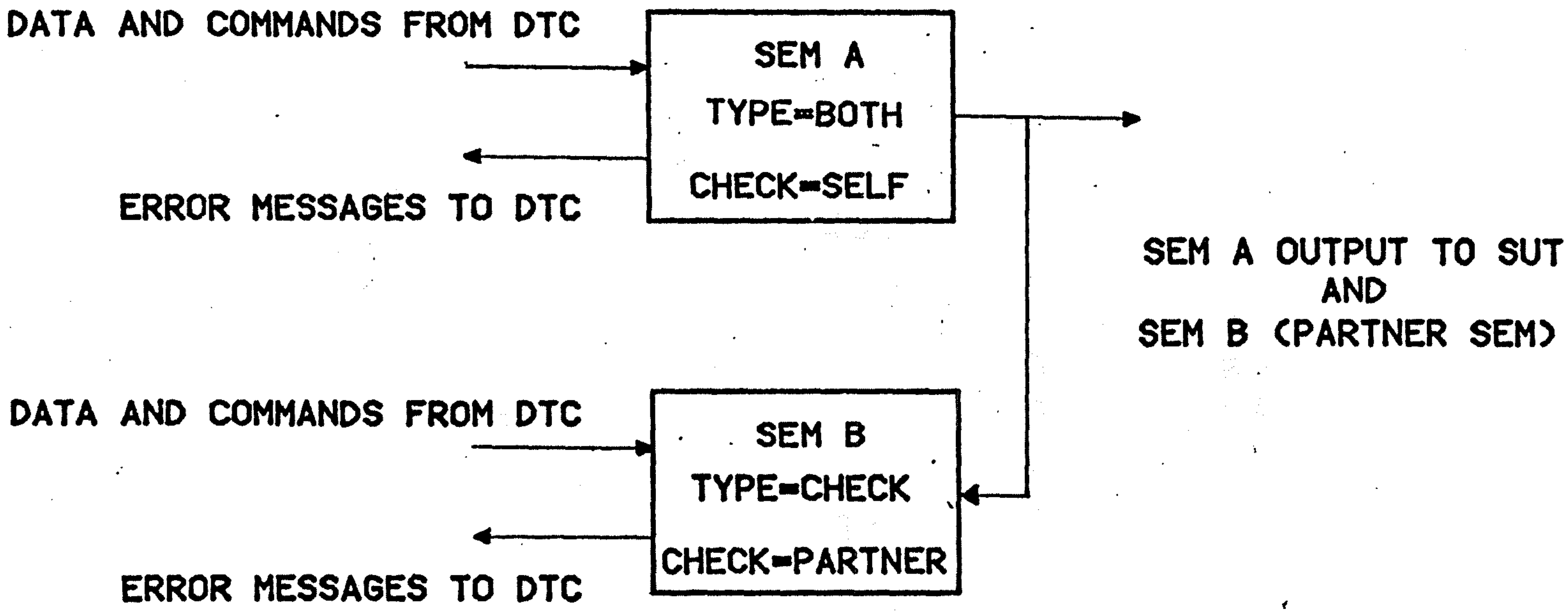
SEM A and B receive different data

Fig. 3.1.20b SEM Operational Configuration



SEM A and B receive the same data

Fig. 3.1.20c SEM Operational Configuration



SEM A and B receive the same data

Fig. 3.1.20d SEM Operational Configuration

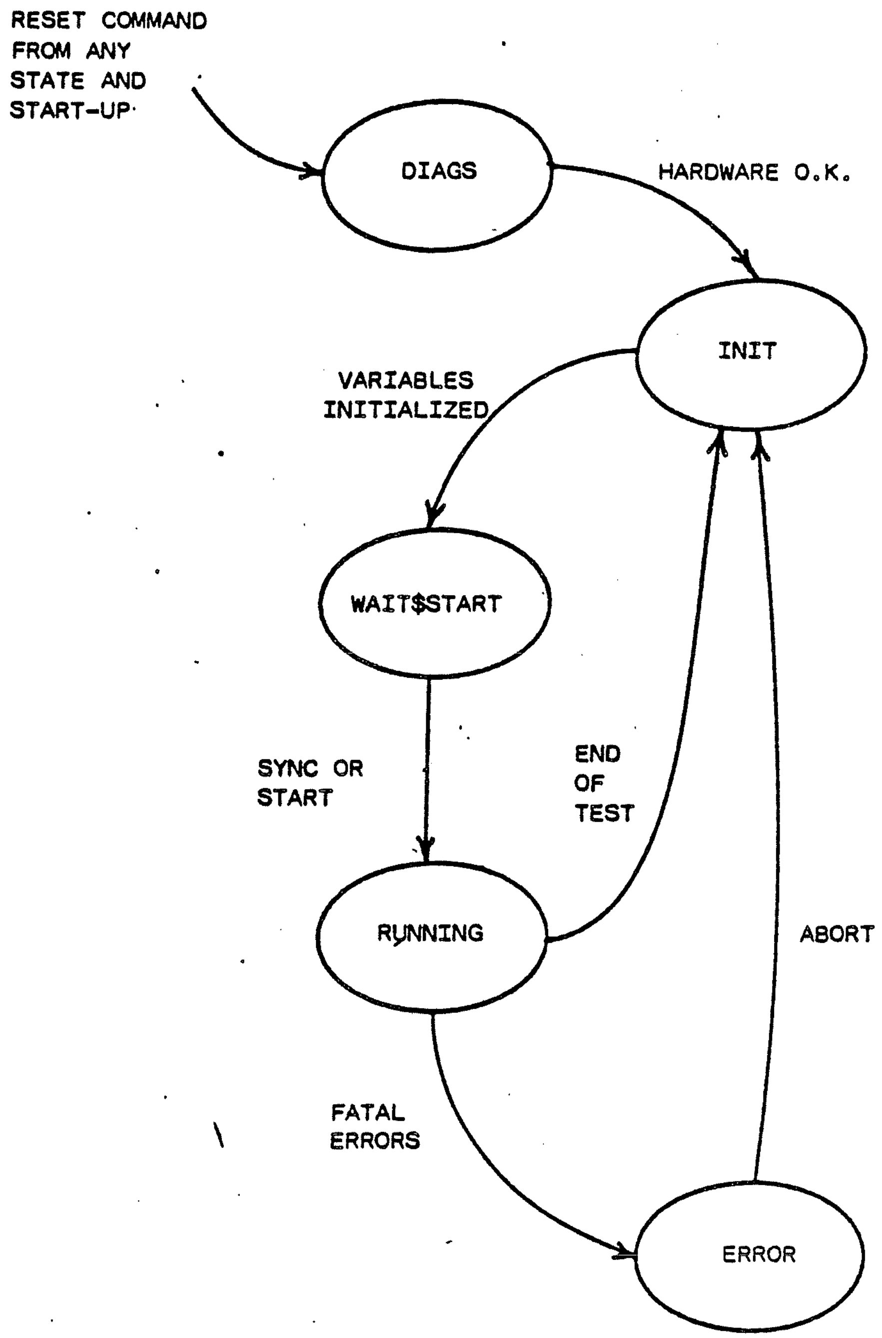


Fig. 3.1.21 SEM State Diagram

SETUPAND$CHECK$MODE. Upon completion of the hardware setup the SEM moves into the WAIT$START state. In this state the SEM waits for a start command or an RST 7.5 interrupt (SYNC) to begin SEM operation and data output. When the SEM receives the RST 7.5 interrupt or a start command, it moves into the running state. In the running state the SEM sends data to the target system and, if the SEM is set up to do data checking, inputs check data and performs the data checking. The SEM will remain in the running state until it encounters a fatal error, a hard error, a RESET command, or an ABORT command. If a fatal or hard error is encountered, the SEM is put into the error state and the current SEM will be terminated. A RESET or ABORT command will put the SEM in the initialization state. The RESET command forces SEM execution back to the beginning of the software and reinitializes the SEM just as if the reset line on the CPU was used or the SEM power was turned off and then on again. If no errors are encountered and there are no commands sent for the SEM to abort the current operation, the SEM will remain in the running state until it encounters a buffer on the receive queue (STATION(2).RECVQ), which contains fewer than six bytes of data. This is interpreted by the software as a buffer with zero point data blocks and the SEM is placed in the initialization state and an end of test message is issued.

3.2 COMMUNICATIONS

3.2.1 HARDWARE

Hardware involved in communications consists of a serial interface (DL-11), a parallel interface (DRV-11), and a Communication Interface Box, which is auxillary to the PDP-11. The serial interface is used by DDCMP for communication of serial data to and from the SEM's. The Communication Interface Box contains an interval timer, RS-232C transmission line driver and receiver, and a high current driver for the serial transmission line to the SEM's. The high current driver increases the fanout of the RS-232C driver. The circuit diagram for the RS-232C transmission line driver, receiver, and high current driver is shown in Figure 3.2.1. The RS-232C driver and receiver are connected to the serial interface of the PDP-11.

The interval timer, whose circuit diagram is shown in Figure 3.2.2, is connected to the parallel interface of the PDP-11. It consists of a crystal controlled frequency generator and dividers, which generate a frequency of 500 Hertz. This clock signal is connected to the input port of the parallel interface of the PDP-11 and causes an interrupt every two milliseconds.

The synchronizing signal (SYNC) that starts all the SEM's simultaneously is generated by the output port of the parallel interface. This signal goes through the cable from the parallel interface, to the Communication Interface Box, and on to the SEM's in the four wire cable that also contains the serial transmission lines.

3.2.2 SOFTWARE

The Data Link Control Software (DDCMP)

The DDCMP Ancillary Control Processor (ACP) is an implementation of Digital Data Communication Message Protocol (DDCMP), which is a data link control protocol specified by Digital Equipment Corporation. Primarily, a data link control protocol is responsible for the integrity of data exchanged between processors on a data link. In this case, DDCMP manages communications between the DTC and SEM's over a full-duplex, multi-point data link. A Cyclic Redundancy Check, CRC-16 is used for error detection. Retransmission of data is the method of error recovery.

The data link control software consists of a device driver, ACP, control program, and real-time display program. The device driver acts as an interface between the user and the ACP. ACP's are special tasks recognized by RSX-11M as capable of handling Queued I/O requests (QIO) resulting from tasks issuing the QIO$ directive. The control program, ZZC, allows users to control the operation of the ACP by commands entered from terminals. Progress of DDCMP can be displayed by using, ZZD, the real-time display program.

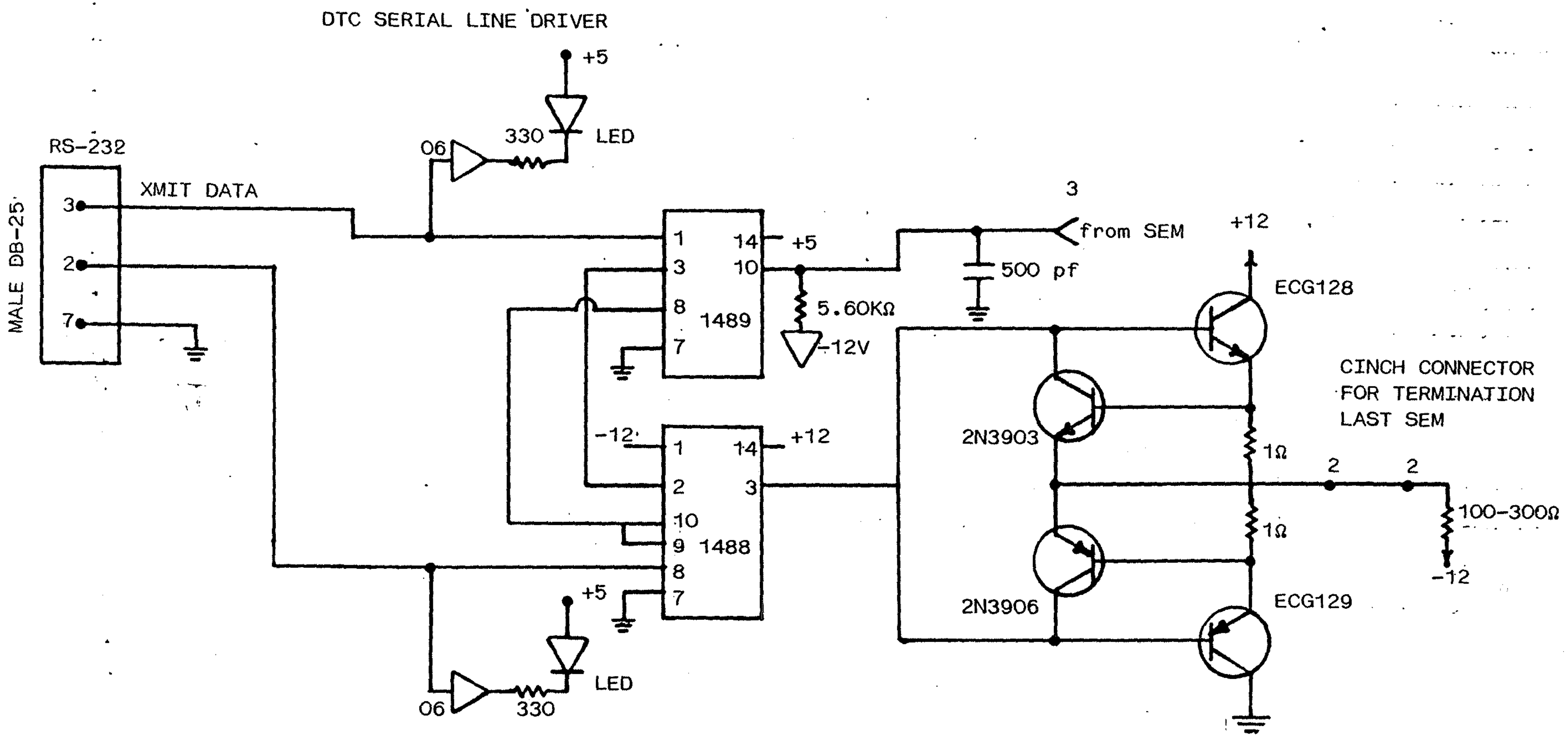


Fig. 3.2.1 Circuit Diagram for Serial Transmission Line Interface between the PDP-11 and SEM's

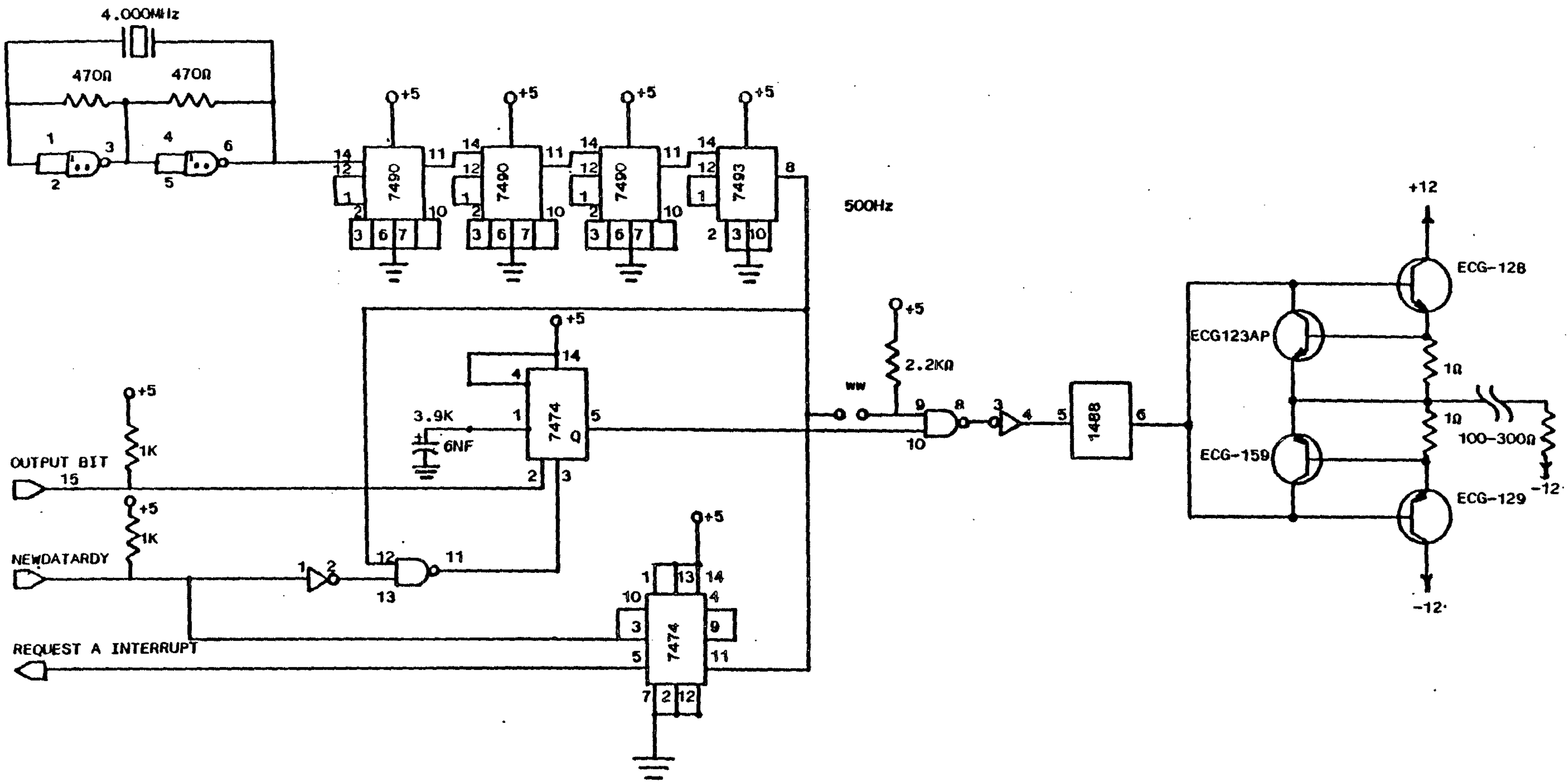


Fig. 3.2.2 Circuit Diagram of the Interval Time Contained in the Communication Interface Box

How DDCMP Works

User tasks issue QIO directives for the ZZ: device. The RSX-11M executive verifies parameters and forms an I/O packet. The ZZ: device driver gets the I/O packet, does some preliminary checking to see that the valid function is requested, and passes the I/O packet to the ACP. Upon receiving the I/O packet, the ACP is unstopped and the requested function is performed. Finally, when the function is performed, the I/O packet is returned to the executive, which completes the QIO request for the issuing task.

I/O functions performed by the DDCMP ACP are Attach (IO.ATT), Detach (IO.DET), Read Logical Block (IO.RLB), and Write Logical Block (IO.WLB). Attach causes DDCMP to initiate communications with a station by going through the startup sequence of exchanging Start and Start Acknowledge messages. Detach stops message exchange with a station and puts that station in the halted state of DDCMP. A Read Logical Block request is queued by the ACP until a data message is received from the station. The received data is passed to the issuing task when a complete data message is received without any errors detected. Write Logical Block causes the ACP to queue the request until a data message is ready for transmission, in which case, the request is handled on a first-in, first-out basis. The request is not completed by the ACP until an acknowledgement from the receiving station is received.

Usage and Operation

First, the DDCMP ACP must be installed, then the ZZ: device driver can be loaded. The following commands do this:

```
>INSTALL $DDCMP
>LOAD ZZ:/HIGH
```

Following is a brief description of programs associated with DDCMP.

DDCMP - an ACP for the Data Link Control Layer which is responsible for the integrity of data exchanged between processors.

ZZDRV - a device driver for the device ZZ:. ZZ: is not a real device, but rather, ZZDRV acts as an interface between user tasks that issue QIO's to ZZ: and the ACP.

ZZCON - a command processor which controls/reports certain aspects of the ACP.
To invoke use: >ZZC (see HELP DDCMP ZZC for more)

ZZRMD - a task which gives a real time display of control blocks used by the ACP.
To invoke use: >ZZD SHOW 1 (see HELP DDCMP ZZD for more)

Following is a table of valid commands for ZZCON, the program for controlling operation of the DDCMP ACP.

Usage	<Result>
ZZC LOG	<enables logging to system console CO:>
ZZC NOLOG	<disables logging>
ZZC SHOW 1	<gives snapshot display of station control block for ZZn:>
ZZC START 1	<starts message exchange with station 'n'>
ZZC HALT 1	<stops message exchange and flushes outstanding I/O>

NOTE: Station and SEM numbers are not the same.
Station refers to the DDCMP station and corresponds to the unit number for ZZ:
Station for command channel = (SEM * 2) - 1
Station for data channel = (SEM * 2)

The following are valid commands for ZZRMD, the program that displays the status of the DDCMP ACP operations in real-time.

Usage	<Result>
ZZD SHOW 1	<gives real time display of control block for station 'n'> works only on VT-100.
<ESC>	change station
<CTRL-Z>	exit
<CTRL-C>	exit
<SPACE>	refresh display

The Alarm Time Tagging Program

The Alarm Time Tagging Program captures alarm messages from the target system, tags the alarm message with the current time, and stores the time-stamped alarm message in a file. Alarm messages from the target system are in the form of asynchronous serial data, which are sent to a terminal such as a CRT console or printer. The alarm is tagged with the current time when the entire alarm message has been captured. Current time is the duration of time from the start of the test as determined from the interval timer in the interface box. This interval timer has a frequency of 500 Hertz and causes an interrupt every two milliseconds. This interrupt is serviced by the ZT: device driver. Alarm messages and the current test duration time for each alarm are stored in a file for processing at a later time by the analysis phase of the test.

How the Alarm Time Tagging Program Works

Characters that are sent to the target system terminal for alarm display are intercepted and received by a serial interface on the PDP-11 (DL-11). Each received character causes an interrupt that is serviced by the interrupt service routine of the alarm time tagging program. This interrupt service routine stores each character in a buffer until the end of the alarm message is recognized. Then the buffer is sent to the main program, which stores the buffer as a record in a file. The only function that the main program performs is accepting buffers from the interrupt service routine and storing these buffers in the file.

Each buffer has a fixed maximum length determined at assembly time. A circular array of eight buffer pointers keeps track of the current state of each buffer. There are two indices for the circular array. One index is for storing buffers from the interrupt service routine and the other index is for retrieval of buffers by the main program. The input index is incremented modulo eight when a buffer is filled with one alarm message and the buffer pointer is placed in the circular array. The output index is incremented modulo eight when the main program has stored the buffer as a record in the file and the buffer pointer is removed from the circular array. This method was employed because it is fast and mutual exclusion of the interrupt service routine is not required.

The interrupt service routine recognizes alarm messages from the target system by using deterministic finite state automata. A string of characters delimits the beginning of an alarm message and another string delimits the end of the alarm. The state machine recognizes the string delimiting the beginning of the alarm and commences storing characters in a buffer. The interrupt service routine continues storing characters until the state machine recognizes the string delimiting the end of the alarm message; whereupon, the buffer is passed to the main program.

Usage of the Alarm Time Tagging Program

The alarm time tagging program uses command line input to determine the file name for the file where alarms are to be stored. The syntax for the command line input is:

```
>output file specification[/switch]
```

where the output file specification is any valid Files-11 file name. The switch is optional and the only valid switch is /AP which means append to the most recent version of the file. Normally, a new file is created and given a higher version number when a file is opened. However, when the /AP switch is used the newest existing file is opened and appended to. If no output file specification is given, then TI: is opened for output. For example, to run the alarm time tagging program the user would
type in:

```
>ALARM SYSTEMA.DAT
```

In this case the file will be created and be used to store alarm messages from the target system. On the other hand, if the user wants to append alarms to an existing file, the following command might be used:

```
>ALARM SYSTEMA.DAT/AP
```

The alarm time tagging program is terminated by an abort task request issued by a command line interpreter or by another task by using the ABRT$ RSX-11M directive. This can be accomplished with the following command:

```
>ABORT/TASK ALARM
```

When this is done, the alarm time tagging program closes the file and disconnects from the interrupt.

In order to collect system responses, it is necessary to receive the system's alarm reporting messages. Because of the different protocols and message formats involved, it is also necessary to customize the alarm program for a specific system to be tested. To modify the alarm tagging program, familiarity with PDP-11 Asslemly Language is requred, along with a knowledge of RSX-11M task building.

How to Modify the Alarm Time Tagging Program

If another port is to be used for capturing alarms it is not necessary to reassemble any programs. First, determine the interrupt vector and control status register addresses for the port to be used. Next, edit the command file for task building the task, "ALARM.CMD." The global definitions (GBLDEF) must be changed to the values desired for $VEC and $DVCSR. $VEC is the address of the interrupt vector and $DVCSR is the address of the control status register. Finally, run the task builder using the new command file:

```
>TKB @ALARM
```

The state machine for recognizing regular expressions is table driven. The table defining the state machine is a separate MACRO source program so that the state machine can be easily modified. In Figure 3.2.3 there is a generic state diagram. There are nodes connected by directed arcs. The nodes represent states and arcs represent state transitions. Corresponding to each state transition there is an input that causes the transition and an action that accompanies the transition. The following is a generic program definition of a state with transitions away from the state:

```
label:
        .WORD   input, nextstate, action
                .
                .
                .
        .WORD   -1, anystate, anyaction
```

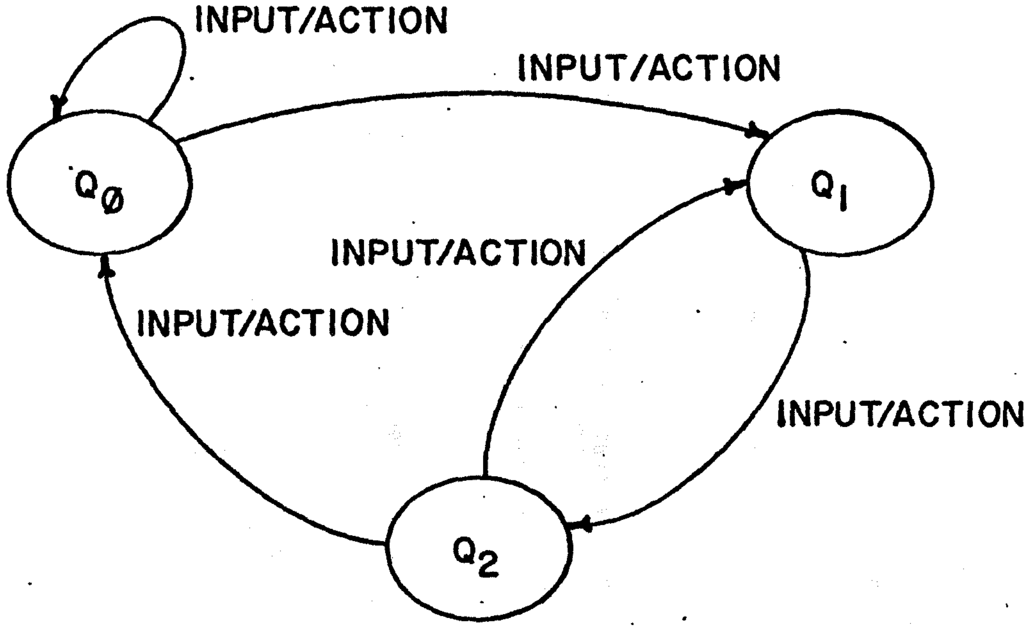


Fig. 3.2.3 A Generic State Diagram

"Label" is any valid label in MACRO-11 and is used to reference that state. "Input" is any non-negative 16-bit quantity that is to cause a state transition. "Nextstate" is a label for the state that the state machine will be in after the input occurs. "Action" is a label for the subroutine that will be called by the state machine when the input occurs. The state definition must end with a definition for an input having a negative value, the default case, which will occur when an input occurs that does not match any of the previous input definitions.

The following is an example of a state table definition for the state machine to recognize the string "ABC."

```
STATE:: .WORD   ZERO
ZERO:.  .WORD   'A,ONE,NOACT
        .WORD   -1,ZERO,NOACT
ONE:    .WORD   'B,TWO,NOACT
        .WORD   -1,ZERO,NOACT
TWO:    .WORD   'C,ZERO,FOUND
        .WORD   -1,ZERO,NOACT
;
; Action routine called by the state machine when no action is to occur.
;
NOACT:  RTS     PC
;
; Action routine called by the state machine when the input is the
; character "C" and the state is "TWO."
;
FOUND:  PRINT   <RECOGNIZED THE STRING "ABC">
        RTS     PC
        .END
```

Additionally, there are two predefined action routines, PUTC and DONE. PUTC is called when the user wants the input saved in a buffer; DONE is called when the user wants the buffer stored in the file.

The alarm time tagging program is easily modified to recognize alarms for different target systems. First, create the MACRO-11 source program for the state definitions; next, assemble the program. Then edit the command file,"ALARM.CMD", for task building so that the state definition module is linked with the remainder of the program. Finally, task build the alarm time tagging program.

```
>EDIT STATES.MAC
>MACRO STATES
>TKB @ALARM
```

Task Builder Command Files

Following is the command file, "ALARM.CMD", for the task building of the alarm time tagging program.

```
ALARM/PR,ALARM/-SP=STATES,TIMETAG,GCML,FDBTA,LB:[1,54]RSX11M.STB/SS
/
PRI=130
GBLDEF=$VEC:230
GBLDEF=$DVCSR:176630
//
```

<u>The ZT: Device Driver</u>

The ZT: device driver handles the interval timer contained in the interface box, keeps track of the current test duration time, and sends the signal for starting all the SEM's. The interval timer is a 500 Hertz clock that generates an interrupt every two milliseconds (tick) through a parallel interface (DRV-11) on the PDP-11. The current test duration time is in ticks, seconds, minutes, hours, and days. Ticks are from 0 to 499, seconds from 0 to 59, minutes from 0 to 59, hours from 0 to 59, and days from 0 to 30. A Unit Control Block (UCB) for ZT: contains the current test duration time. The signal for starting all the SEM's (SYNC) is transmitted from the interface box when the interval timer is started.

A QIO$ directive with the write logical block I/O function causes the ZT: device driver to start the interval timer, zero the current test duration time, and send the SYNC signal. On the other hand, a QIO$ directive with the read logical block I/O function causes the ZT: device driver to stop the interval timer and reset the SYNC signal. The current test duration time is left intact until a write logical block is performed.

The ZT: device driver is installed, loaded, and ready for usage by using the following commands:

```
>LOAD ZT:/HIGH
```

3.3 DYNAMIC TEST CONTROLLER (DTC) USER MANUAL

3.3.1 INTRODUCTION

This document is intended only to supplement (not replace) the LSI-11/23 hardware and software manuals provided by the vendors and included with the test fixture. The DTC User Manual will summarize the hardware required to run the test fixture, and detail the specialized hardware built by WVU. In addition, hardware options selected are outlined. For specific descriptions of Digital Equipment Corporation hardware, refer to the appropriate hardware reference manual. The RSX-11M Operating System, which forms the basis of the DTC software, is described extensively in the vendor supplied system's documentation. Therefore, this manual will describe the software written by WVU, with explanations of special features of RSX incorporated into the DTC software package.

3.3.2 HARDWARE

The hardware consists primarily of an LSI-11/23 CPU with the floating-point option (KEF11-AA) and two RL02 10 Megabyte removable disk packs. At least 128K words of memory is recommended. Possibly, the system could be built around a smaller mass storage device, but the RL02 disks allow full on-line "help" and extensive archival of test results. A minimum of four serial lines (RS-232) are required; one for the operator console, one for the communications interface, one for the graphics terminal and plotter, and one for use in monitoring the target system's console messages. A single parallel interface is also required to control the system test synchronization circuit (included in the communications interface). The serial line control register addresses and interrupt vectors are set at operating system generation time. Aside from the console, serial line assignments may vary, but the communications task, DDCMP, and the response recording task are assigned to a specific serial line during task building. WVU used standard locations for all I/O device registers and vectors. Two LSI-11/23 systems were used in the development of the test fixture. An LSI-11/23-PLUS system was used for multi-user software development and is referred to as the "BIG SYSTEM". The other system, used as a test system for debugging is referred to as the "SMALL SYSTEM". Either system can operate as the test fixture DTC.

```
               BIG SYSTEM
            Card Installation
        A        B        C        D
      I-------+-------+-------+-------I
    1 I          KDF11-B CPU          I
      I-------+-------+-------+-------I
    2 I       MSV11-PL 512KB MEM      I
      I-------+-------+-------+-------I
    3 I       MSV11-PL 512KB MEM      I
      I-------+-------+-------+-------I
    4 I MSV11-LK 256KB       BLANK    I
      I-------+-------+-------+-------I
    5 I DLV11-J SIO          BLANK    I
      I-------+-------+-------+-------I
    6 I DLV11-J SIO          BLANK    I
      I-------+-------+-------+-------I
    7 I DRV11 PIO            BLANK    I
      I-------+-------+-------+-------I
    8 I       RLV12 DISK CTRLR        I
      I-------+-------+-------+-------I
    9 I             SPARE             I
      I-------+-------+-------+-------I
```

Processor Type: 11/23-PLUS   Memory Size: 640KW

Options: Floating Point Processor (KEF11-AA)
Extended Instruction Set
Extended (22 bit) Addressing

Device Configuration:

Device	Vector	CSR	Unit	Type	Comment
DLA	160	174400			Slot 8
			0	RL02	Volume RSXM32 System Disk
			1	RL02	Volume USER04 User Disk
			2	RL02	Not Used
YLA	60	177560	1		TT0 Console (CPU Card) 4800 Baud
YLB	300	176500	0		TT1 (Slot 5) 4800 Baud
YLC	310	176510	1		TT2 (Slot 5) 4800 Baud
YLD	320	176520	2		TT3 (Slot 5) 4800 Baud
YLE	330	176530	3		TT4 (Slot 5) 4800 Baud
YLF	340	176540	2		TT5 (CPU Card) 4800 Baud
YLG	200	176600	0		TT6 (Slot 6) 1200 Baud
YLH	210	176610	1		TT7 (Slot 6) 1200 Baud
YLI	220	176620	2		TT10 (Slot 6) 300 Baud
YLJ	230	176630	3		TT11 (Slot 6) 9600 Baud

Memory Module	17772100	Addr. 0-256KW (Slot 2)
Memory Module	17772102	Addr. 256-512KW (Slot 3)
Memory Module	17772104	Addr. 512-640KW (Slot 4)

SMALL SYSTEM
Card Installation

```
          A       B       C       D
        I-------+-------+-------+-------+
      1 I            KDF11 CPU          I
        I-------+-------+-------+-------I
      2 I MSV11-DD 64KB        BLANK    I
        I-------+-------+-------+-------I
      3 I MSV11-DD 64KB        BLANK    I
        I-------+-------+-------+-------I
      4 I MSV11-LF 128KB       BLANK    I
        I-------+-------+-------+-------I
      5 I DLV11-J SIO          BLANK    I
        I-------+-------+-------+-------I
      6 I DRV11 PIO            BLANK    I
        I-------+-------+-------+-------I
      7 I RXV21 DISK           BLANK    I
        I-------+-------+-------+-------I
      8 I        RLV11 DISK CTRLR       I
        I-------+-------+-------+-------I
      9 I              SPARE            I
        I-------+-------+-------+-------I
```

Processor Type: 11/23   Memory Size: 128KW

Options: Floating Point Processor (KEF11-AA)
          Extended Instruction Set (EIS)

Device Configuration:

Device	Vector	CSR	Unit	Type	Comment
DLA	160	174400			Disk Controller (Slot 8)
			0	RL02	Volume:RSXM32 System Disk
			1	RL02	Volume:USER01 User disk
DYA	264	177170			Floppy Disk Controller (Slot 7)
			0	RX02	
			1	RX02	
YLA	60	177560	3		TT0 Console (Slot 5) 4800 Baud
YLB	300	176500	0		TT1 (Slot 5) 1200 Baud
YLC	310	176510	1		TT2 (Slot 5) 1200 Baud
YLD	320	176520	2		TT3 (Slot 5) 9600 Baud
Memory Module		17772100			Addr. 0-64KW (Slot 4)

3.3.3 SOFTWARE

The DTC software package consists of separate computer programs or "tasks", that interact through "executive calls", common data structures, and file exchange. The tasks perform three basic DTC functions: test set-up and data generation, communications and test control, and data analysis. The RSX11-M operating system manages task memory requirements, device (I/O) interfacing, and inter-task communications.[8] The remainder of this document will describe the DTC tasks and the library of system interface procedures written to interface them to the operating system.

<u>COMMON DATA STRUCTURE</u>

A common data structure definition was used by the Pascal compiler to allow tasks to access a memory-resident common area. The structure defines the data types using the user-defined type characteristic of the Pascal Language.[7] Because of its extensive use, the annotated source code for the data structure is shown below:

```
§semcfgtyp.h standard constants and type descriptions†
        maxsems = 21; §number of SEMs supported†
        maxpoints = 420; §84 analog, 336 digital†
        sembufsize = 86; §SEM internal data buffer†
        cmdbufsize = 32; §size of SEM command buffer†
        resbufsize = 200; §size of SEM response buffer†
    type
        byte = 0..255; §defines an 8-bit variable quantity†
        typarameter = (av,ch4,co2,co,temp,o2,volt,unkn);
            §typically analog mine sensors measure air-velocity (av),
            methane (ch4), carbon dioxide (co2), carbon monoxide (co),
            temperature (temp), oxygen (o2), voltage (volt), or an
            unknown type sensor (unkn)†
        tysensor = record §describes a single analog sensor†
            param: typarameter; §what the sensor is measuring†
            scale: real; §full scale value†
            alarm: real; §alarm limit†
          end;
        tyanalog = record §assign an analog sensor to model mine location†
            model: integer; §branch or junction number†
            kind: tysensor §describe the sensor†
          end;
        tydigital = record §2 bytes make up a 16 bit digital point†
            low,high: byte
          end;
        tyuse = (busy,free); §describes buffer use†
        tystat = (halted,running,offline); §describes condition of SEM
            or DTC†
        tyiosb = record §I/O status structure used by op-sys†
            retcode,xfercnt: integer §I/O status code,data transfer count†
          end;
```

```
tycmdbuf = packed array[1..cmdbufsize] of char;
    §ASCII buffer for storage of messages TO SEMS†
tyresbuf = packed array[1..resbufsize] of char;
    §ASCII buffer for storage of messages FROM SEMS†
tydatbuf = record §data transfer buffer status record†
    use: tyuse; §busy or free buffer†
    tick: integer; §test data sequence number†
    stat: tyiosb; §I/O status for testing data transfer†
    bufno: integer §number of data buffer†
  end;
tycheck = (none,self,partner,all);
    §SEMS can do self checking of data, check on a partner SEM,
    neither (none), or both (all)†
tytype = (out,check,both);
    §SEMS can be output only, check only, or both†
tyunit = (tmilli,tsecs,tmins);
    §SEM time units recognized for data output interval†
tystate = (inactive,initial,diagnos,waitstart,run,error);
    §SEM internal states, only 'waitstart', 'running', and
    'error' states are accessable by DTC†
tychannel = (cmd,dat,res,dump);
    §SEM communications actually assigns two DDCMP remote
    addresses to each SEM.  At addr n, commands and responses
    (in ASCII) are exchanged, while at addr n+1, data is
    transfered (in 8-bit binary).
tytimedate = record §format for real time†
    year,month,day: integer;
    hour,minute,second: integer;
    tick,tickrate: integer
  end;
tysem = record §describes an SEM†
    id:      1..maxsems; §the SEM identification number†
    stat:    tystat; §SEM status†
    channel: array[cmd..dump] of tyuse; §communications status†
    tickrate: integer; §SEM data output interval†
    tickunit: tyunit; §SEM data output interval units†
    check: tycheck; §mode of use of sem†
    semtype: tytype; §another mode descriptor†
    state: tystate; §SEM internal state†
    lasterror,errseq: integer; §error counters†
    analog: array[1..4] of tyanalog; §describes the 4 D/A outputs†
    digital: tydigital; §describes 16 relay outputs†
    buff: array[0..1] of tydatbuf; §buffer status blocks†
    cmdbuff: record §buffer for commands to the SEM†
            use: tyuse;
            stat: tyiosb;
            data: tycmdbuf
          end;
    resbuff: record §buffer of responses from the SEM†
            use: tyuse;
            stat: tyiosb;
```

```
                data: tyresbuf
              end
        end;
      tytest = (setup,exercise,fire,disaster);
          §some possible test scenarios and conditionst
      tyline = packed array[1..81] of char; §generic line buffert
      fname = packed array[1..20] of char; §file name buffert
      tytick = record §cumulative data 'ticks' in testt
          high,low: integer
        end;
      tydtc = record §decribes internal states etc. of dtct
          test: tytest;
          stat: tystat;
          targetname: packed array[1..10] of char;
          dataname,respname: fname;
          sems: array[1..maxsems] of boolean;
          start: tytimedate;
          tick: tytick;
          tickrate: integer; §data output intervalt
          tickunit: tyunit; §interval unitt
          comflag: boolean; §communications status flagt
          logflag: boolean; §test log flagt
          errorcount: integer
        end;
      typoint = (sema,semd); §analog or digitalt
      tytarget = record §mapping of target points to sem outputst
          semno: integer; §sem numbert
          alarm: boolean; §alarm conditiont
          kind: typoint; §type of outputt
          pointno: integer §sem output numbert
        end;
§end semcfgtyp.ht
```

SEMCFG: SEM Configuration Data. This module contains the data structure for the Sensor EMulator (SEM) used by the Pascal programs. The module is written in Pascal with extensive use of user defined data types for ease of understanding. This module is compiled first using the 'list' feature of Whitesmith's Compiler to provide a Macro-Assembler source listing.

To Compile: @LB:[1,1]PASCAL SEMCFG /LIST/NOLINK

The file SEMCFG.MAC (Macro source) must be edited to remove C$text references and PSECTS directives. These are necessary only when executable code is being generated. The compiler presumes all programs are executable, but the SEMCFG module is task built without the usual task header and installed as a memory resident data common in the partition SEMCFG. See RSX-11M Task Builder Manual [3] for information on task building resident data commons. After editing, SEMCFG.MAC is assembled.

To Assemble: MACRO/LIST:SEMCFG SEMCFG

Before this can be task built, the partition 'SEMCFG' must exist. This partition must be large enough to contain the SEMCFG task. The size of SEMCFG can be determined from the listing generated by the macro-assembler. The current size is 12,352 words.

To create this partition: SET PARTITION/GEN/TOP:-602

This creates an unassigned memory block 602 (octal) by 64 bytes 12,352 words long at the top of the partition GEN. The BASE of the partition SEMCFG will be at the TOP of GEN, which can be found using:

```
SHOW PAR:GEN
```

Base and size are added to find the top of GEN. The partition SEMCFG is then created using:

```
SET PARTITION:SEMCFG/COMMON/BASE:(GEN TOP)/SIZE:602
```

SEMCFG.TSK can now be taskbuilt as the task builder can compute the phyiscal memory address from the partition charactersitics.

To Task build:
```
SEMCFG/PI/-HD/CO,SEMCFG/-SP,SEMCFG=DL1:[1,11]SEMCFG
LB:[1,1]PLIB/LB,LB:[1,1]CLIB/LB,LB:[1,1]PLIB/LB
/
PAR=SEMCFG
STACK=0
//
```

This builds a task SEMCFG without a task header, specifying position independent code and shared common characteristics. A symbol table file SEMCFG.STB is created. This file is used by the task builder to resolve references to the common data by other tasks that will access the data block. A stacksize of 0 is specified since the task is not executable code and requires no stack.

After task building, the task SEMCFG.TSK is installed in the partition SEMCFG using:

```
INSTALL/PARTITION:SEMCFG SEMCFG
```

Once this is completed, programs that access the data block SEMCFG can be created. In order to task build these programs, the UIC under which they are built must contain both the SEMCFG.TSK and SEMCFG.STB files.

```
var
    sem:    array[1..maxsems] of tysem;
    .
```

The variable name 'sem' is a global variable that can be imported by other Pascal tasks by including it in the program declaration. See Whitesmith's Pascal Programmer's Manual [9] for description of the use of external globals. The declaration of 'SEM' should be identical for all programs so that the data structure is consistent.

TEST SET-UP AND COMMUNICATIONS

Test set-up is accomplished through the sequential execution of programs. An initialization program is used to set the common data structure to the specific conditions required to start a test. Once initialization is complete, DDCMP communications between the DTC and the SEM's is started. This communications is continued over multiple tests, unless a new configuration of SEM's is desired. With communications running, the test fixture is ready to conduct the individual tests. For each test, the alarm tagging program, which records the target system's response, must be run before the test is begun.

INITIAL: Program to initialize the common data structure. The initialization program is a Pascal program that sets the memory data common area to the default values for the target system to be tested. The program is easily modified for each target system configuration. For each target system input, a corresponding SEM output is assigned. After the source code for the initialization program is set up for the test configuration, the program is compiled and linked to the common data area.

SOURCE FILE: INITIAL.PAS

TO COMPILE: PAS INITIAL Listing generated with the LIP Utility.

TO LINK: TKB @INITIAL.BLD

COMDRV: Program to interface test software and DDCMP. This program functions as the interface between the DDCMP driver software and the SEMCFG message buffers. The program is written in the Pascal programming language. COMDRV contains procedures for initialization of the SEMCFG communications data area, timestamping message transfers and logging messages to a log file 'COMDRV.LOG', logging errors to the file and a main driver procedure for polling of SEM buffer and communications status.

The COMDRV main program opens the file COMDRV.LOG and writes a timestamped startup message into the file. Initialization also assigns the initial condition free to the two DDCMP channels allocated for each SEM and used by COMDRV. The communication status COMSTAT[n] is set to 'offline'. Following intialization, the main program begins round robin selection of SEM's and calls the driver procedure to execute communications functions. This selection continues until the 'done' variable is true. The done variable can be set true only after shutdown of DDCMP communications results in NO active channels, which cannot occur until one or more DDCMP remote stations (SEM's) have been started. After 'done' becomes true, the main program writes a timestamped completion message into the log and closes the log file before exiting. Most of the processing is done in the procedure DRIVER, which is described in the next section.

COMDRV: procedure DRIVER. This procedure monitors the state of each SEM, its buffers, and its allocated channels. The state of the SEM and its communications state (stored in variable COMSTAT) are used to determine when to startup and shutdown communications with that SEM, as shown:

SEM.STAT	COMSTAT	ACTION
offline	offline	Initial Condition...no action.
halted	offline	Startup DDCMP with this SEM, and set SEM.STAT & COMSTAT = running.
halted	running	Shutdown DDCMP with this SEM, set SEM.STAT & COMSTAT = offline, and test for 'done'.
running	XXX	Test buffers and channels and initiate I/O and set completion flags.

The startup process calls external function PSEMSTART, which attaches the task to the proper device. A timestamped log entry is made to indicate DDCMP startup with the particular SEM and the logical channel assignment is saved.

The shutdown process calls external procedure DETACH with the saved logical channel assignment to stop DDCMP communications with the SEM. A timestamped log entry is made that indicates DDCMP shutdown. A test is made of all communication states to see if any SEM channels are still active. If not, then 'done' is set and COMDRV will exit.

The main business of the DRIVER procedure is buffer and channel allocation and testing. Each buffer contained in the SEMCFG memory common is a 'use' indicator of either 'free' or 'busy'. Each SEM channel allocated by COMDRV also has a use of 'free' or 'busy'. DRIVER treats these indicators as 'states' and takes appropriate action, depending on the channel.

The command channel processing uses external procedure SEMWRITE to initiate a DDCMP transfer to the SEM. The command channel state table and actions are:

BUFFER	CHANNEL	DRIVER ACTION
FREE	FREE	None. (initial state)
FREE	BUSY	None. (unreachable state)
BUSY	FREE	Initiate DDCMP and mark channel 'busy'.
BUSY	BUSY	Test IOSB for channel done, if done then mark buffer & channel 'free' and log the transfer. If not done, no action.

The response channel processing uses external procedure SEMREAD to initiate a DDCMP transfer from the SEM. The command channel state table and actions are:

BUFFER	CHANNEL	DRIVER ACTION
FREE	FREE	None. (initial state)
FREE	BUSY	None. (unreachable state)
BUSY	FREE	Initiate DDCMP and mark channel 'busy'.
BUSY	BUSY	Test IOSB for channel done, if done then mark buffer & channel 'free' and log the transfer. If not done, no action.

Note that in each case, it is necessary for some other task to act during the initial state and change the state of 'BUFFER' to busy, so that DRIVER will initiate DDCMP tranfers. Initiation of DDCMP transfer varies by channel; however, the main action is to zero the Input/Output Status Block (IOSB) and compute buffer length, then call the external procedure.

COMDRV: Summary

NAME COMDRV - program to interface DDCMP and the SEM buffers.

FUNCTION COMDRV is a 'state' driven program that initializes the SEM configuration (SEMCFG) memory resident common, initiates DDCMP message transfers between the tasks accessing the SEM buffers and the SEMs and logs all communications actions and errors.

SOURCE FILES VOLUME USER04 UIC=[1,10]COMDRV.PAS
VOLUME USER04 UIC=[1,10]COMDRV2.PAS
VOLUME USER04 UIC=[1,10]COMDRV3.C

COMPILER INSTRUCTIONS

```
        PAS COMDRV
        PAS COMDRV2
        Listings generated by the LIP utility program.
        CCO COMDRV3
        Listing generated by the LIC utility program.
```

TASK BUILD INSTRUCTIONS

```
        SET DEF [1,11]
        COPY [1,10]COMDRV.OBJ [1,11]
        COPY [1,10]COMDRV2.OBJ [1,11]
        COPY [1,10]COMDRV3.OBJ [1,11]
        TKB @COMDRV.BLD
                where COMDRV.BLD is:
```

```
        DL1:[1,12]COMDRV/CP/FP/RO,  DL1:[1,13]COMDRV/-SP=LB:[1,1]CHDR
DL1:[1,11]COMDRV,DL1:[1,11]COMDRV2,DL1:[1,11]COMDRV3
        DL1:[1,11]DTCUTILS/LB
LB:[1,1]PLIB/LB,LB:[1,1]CLIB/LB,LB:[1,1]PLIB/LB
        /
        PAR=GEN
        LIBR=FCSRES:RO
        RESCOM=SEMCFG/RW
        UNITS=50
        STACK=3000
        //
```

INTERNAL GLOBALS: None.

EXTERNAL GLOBALS

SEM-memory resident common in SEMCFG (read/write access)
Calls PCREATE & PCLOSE external procedures
Calls PGETTIME,PDETACH,PDELAY from DTCUTILS/LB

FILES & DEVICES

File COMDRV.LOG - log file for communication events.
NLO: The system NULL device.
ZZ1: through ZZ40: DDCMP drivers.

EXAMPLE USE

ALARM: Program to collect target system responses.

DATA GENERATION and TEST CONTROL

The test data is generated by a program named "DATGEN". This program provides a menu for the operator, allowing him to select a data algorithm. Based on experience with the early tests, it was determined that only two data generator algorithms were required to allow meaningful data analyis. A "pulse" test algorithm provides for simultaneous alarming of ALL target system inputs (analog and digital). The operator can select a pulse width of 1 "tick" (or SEM output interval) or a pulse width of 86 ticks (standard SEM data buffer size). By varying the "tick" rate, and choosing between pulse width options, a "one-shot" burst of alarms can be generated. A second algorithm, called the "square wave" test, provides an alternating alarm/reset condition on a SINGLE target system input. An alarm duration of 1 tick or 43 ticks (single cycle per buffer) is available to allow a full range of test conditions. However, multiple buffers of data can be generated, allowing this test to continue for as long as desired. Since only one SEM is required to do this test, it is recommended that the communications driver (COMDRV) be started specifically for this test with only a single SEM. The third data algorithm, is based on the results of an analysis of the WVU model mine conducted by the Greuer/Edwards ventilation program. Unfortunately, this program could not be run "on-line" using the DTC processor due to excessive data storage requirements, but it was run on a VAX computer and the results were used to generate a simulated fire condition.

DATGEN: Program to load SEM's with test data.

SEMCMD: Program to process commands from the DTC operator to the SEM's. This program allows the operator to communicate with the SEM's from his terminal. SEMCMD assigns the pseudo device NLO to the operator's terminal so that all messages from the SEM's are displayed on that terminal. The operator is prompted for input (commands to be sent to SEM's) and destination (a specific SEM or ALL). When the operator sends the "GO" command, the SEM's report their statuses. SEMCMD checks the SEM tick values, contained in each status report, to verify that they are consistent.

SEMCMD utiliizes the terminal independent screen formatting procedures written for the DTC. Using these procedures the SEM messages are displayed in reverse video (black characters on white background) to distinguish them from SEMCMD messages. Only one terminal in the multi-user system can run SEMCMD at a time, since the NLO device an only be assigned to one terminal.

RTD: Program to monitor test progress (Real-Time Display). This program provides a display of the SEM status. The display is continuously updated on terminal types VT100, VT52 (also ADDS Viewpoint), and user-defined terminal model 128 (Tektronix 4025). All other terminal types are treated as LA-36 Decwriters and the display is updated in response to operator input. The program has read-only access to the memory resident common data block, SEMCFG. The status information in this data area is displayed; therefore, it does not necessarily reflect the current state of an SEM. Changes in SEM status must be made by other tasks in order for the common data to be accurate.

The RTD program source is contained in three files which are linked after compilation along with the necessary library routines. File "RTDISPLAY.PAS" contains the main program, while "RTDISPLA2.PAS" is an unnamed Pascal program that handles screen formatting, and SEM selection. The file "RTDISPLA3.C" contains the C Language procedures necessary to interface RTD to the operating system and data common areas. An example of the RTD program as it works with the LA-36 Decwriter is shown below.

<u>EXAMPLE USE OF RTD</u>

The program is invoked by typing 'RTD<CR>'. The command line interpreter automatically searches for the installed 'RTD' program and runs it. On a printing terminal the use instructions are printed; on video terminals the screen is cleared. The program then prompts for the SEM ID (number of the SEM from 1 thru 20). After the

SEM ID is selected, the real-time display is written to the terminal. The real-time display is made up of four (4) main segments: the heading, the SEM status flags, the sensor assignments, and the data buffer contents. The first line of the display has the heading (WVU-USBM, etc.) along with the date and time. When used with screen terminals, the date and time are updated once each second.

The SEM status line contains the SEM ID, its current status, the TICK setting and TICK scale, and the status of communication buffers. When used with video terminals, these status variables are updated each second. The sensor assignments show the SEM output, its assignment in the model mine, SENSOR type and units, the full scale value of its output (RANGE), and the current alarm limits defined for that output. The range and limits are shown in the proper units for the output as currently defined. These values are updated each second on screen type terminals.

The operator may select a new SEM for display by typing the 'ESC' key and then responding to the program prompt. Typing a 'space' will result in the entire display page being rewritten, which is the only way to update the whole display on Decwriter type terminals. Finally, typing a control-Z will terminate the program.

STOP: Program to close down a test.

ERROR RECOVERY

The error recovery built into the DTC software is concerned primarily with communications and remote (SEM) errors. The communications protocol (DDCMP) provides for retransmission of messages when an error is detected. Since a CRC error detection scheme is part of DDCMP, most errors are detectable. DDCMP error recovery is transparent to the DTC software; however, the result of too many DDCMP errors is an error count threshold exceeded condition. This condition returns an error status to the DTC software, which then attempts to restart the channel and notifies the operator. Communications events are logged in a file called "COMDRV.LOG" for diagnostic purposes and for a permanent record of the test.

Remote SEM error conditions are processed by the DTC software. If the error is "fatal" (no recovery possible in real time), then the SEM is shut down and the operator is notified. It is up to the operator to invalidate the test data for that particular SEM. If the error is "non-fatal," the operator is notified, but the software attempts recovery by resetting the SEM state.

Other error detection and processing is performed by the operating system and the command line interpreters. These errors are normal ones associated with operator input or task status and the operator is notified with a message describing the error condition.

DATA ANALYSIS (see section 3.4)

FILE STRUCTURE

The design of the DTC software will make use of the RSX-11M multi-tasking capability, allowing elements of the software to be independent programs (tasks) executing on a common data base and file structure. The file structures fall into four (4) categories: task image, static data, dynamic data, and log files. The global data used by the software will include the SEM configuration and test (DTC) configuration data areas. For ease of use, these data areas will be memory resident commons with each task having access privileges as required.

TASK IMAGE FILES: The executable programs that make up the DTC software are compiled and task built into task images. These task images are stored on disk and loaded by the system when required. The DTC will also require some of the system supplied tasks. System tasks will be saved in the system directory UIC=[1,54] on the system disk. Applications tasks will be saved in UIC=[1,12].

STATIC DATA FILES: Some of the data will be fixed and will not change regardless of the test configuration. For example, the model mine physical structure is fixed and will be kept in a static data file. Access to static data files will be 'read-only' so that software cannot accidently corrupt these files. The file STARTUP.CMD in UIC=[1,2] contains the system startup command sequence, which is executed by the system when it is 'booted'. This command file will install the necessary tasks, define peripherals, query the operator for time and date information, set up the memory common partitions, and perform system initialization.

DOCUMENTATION FILES (HELP FILES): These files will be included with the DTC software package.

DYNAMIC DATA FILES: The dynamic data files include the target system response files, input data files, and test configuration files. These files can be superceded by new versions and may differ from test to test.

LOG FILES: Log files of all communications transfers and error conditions encountered can be kept. These are useful for analyzing test actions.

3.4 DATA ANALYSIS SOFTWARE PACKAGE USER'S MANUAL

3.4.1 INTENDED AUDIENCE

This manual is intended to be used by the personnel who wish to create and retrieve statistical data from the Mine Monitoring Test Facility. A knowledge of simple computer data entry is necessary and a knowledge of data files would be helpful.

Of the six programs documented in this manual, five are 90% menu driven. The remaining program only requires the user to answer one question. One program requires the user to perform pattern recognition and to determine pattern uniqueness of lines containing alphanumeric characters before using it. The manual explains in detail how to select and enter data, how to interpret each program's output, and how to effectively use the Tektronix graphics terminal and plotter. It contains no software source code.

3.4.2 INTRODUCTION

The Contents Of The Package

The Data Analysis Software Package for the WVU-USBM Mine Monitoring Test Facility contains six programs.[10]##The programs are:

1. CREATEMAP - An editing program designed for creating the full SEM to System Under Test (SUT) conversion map (Full Map). Requires pattern recognition techniques of lines containing alphanumeric characters.

2. DATACOND - Performs the data analysis and condensation on each test, creates the graphing data, and stores the data into a single condensed file. Requires about 90 seconds per test data file.

3. DATAGRAPH - Graphs the graphing data contained in the condensed data file. Requires the Tektronix 4025 terminal.

4. PRINTSTAT - Prints the test and graphing data contained in the condensed data file.

5. PRINTMAP- Prints the Full Map file.

6. PRINTTEST - Prints the time stamped test data files and the contents of the system file.

The package allows the user to generate and print statistical data on both individual tests and test groupings for a mine monitoring system. It graphs the data contained in the test groupings and allows the user to view the test data files created during the actual testing as well as the files created by the CREATEMAP and DATACOND programs.

3.4.3 SOFTWARE PACKAGE OVERVIEW

Basic Operations

During the testing of a mine monitoring system (known as the system under test or SUT), the Mine Monitoring Test Facility (MMTF) generates and sends data to the SUT. The generated data represents alarm and reset signals to the SUT. The SUT responds to the data by printing signal status reports on a printer or a video display terminal. The MMTF time stamps and saves the responses from the SUT in test data files - one test data file for every test performed. After the tests have been completed, the Data Anaylsis Software Package (DASP) regenerates the SUT input data, examines the time stamped data files, and evaluates the performance of the SUT. The results are given in both statistical and graphical form.

File Structure And Nomenclature

The MMTF software creates a system file for every mine monitoring system tested. The system file contains the names and locations of the test data files associated with that mine montoring system. The name of the system file is controlled and created by the user of the Dynamic Test Controller (DTC) software. The system file name is the "key" to running the Data Analysis Software Package. All DASP programs, except PRINTTEST, require the user to know the system file name in order to retreive the proper data.

The names of the test data files are created by the DTC software and are automatically saved in the system file. The DTC and DASP users have no control over the naming of these files, but the names can be obtained by using the DASP's PRINTTEST program.

The Mapping File (Full Map)

The MMTF contains 21 Sensor Emulators (SEM) each containing 16 digital output points and 4 analog output points which totals to 336 digital output points and 84 analog output points. Some or all of these points are connected to the input points of the SUT. A mapping file created by the DASP user, relates the MMTF points to the corresponding SUT points. The name of mapping file is the Full Map. With the Full Map, the DASP knows which SUT points were connected to the MMTF. It also knows which and when SUT points should respond to the input signals and what data contained in the test data files is useful. By combining the Full Map file the test data files, and the SUT input data, statistics showing response time delays and missed alarms of the SUT can be obtained.

The Heart Of The Package

The heart of the Data Analysis Software Package is the DATACOND program. DATACOND performs all of the statistical analysis and saves the results in a file. All of the other programs merely provide the means to create and view input to the DATACOND program or view its output.

The most difficult part of the data analysis procedure is creating the Full Map. The Full Map requires the user to recognize patterns in the output data of a SUT. Since the data should be alphanumeric characters the patterns shouldn't be difficult to recognize but it will require some forethought. (The pattern recognition and selection is discussed in-depth in the section entitled THE CREATEMAP PROGRAM.)

Procedure For Running The Package

The package can be used anytime after the tests have been performed on the SUT with the exception of the PRINTTEST program. The PRINTTEST program is independent of the other five programs and can be used at anytime during and after testing a system. The remaining five programs must be executed in the following order:

```
LEVEL 1              CREATEMAP
LEVEL 2          DATACOND - PRINTMAP
LEVEL 3    DATAGRAPH - PRINTSTAT - PRINTMAP
```

The CREATEMAP program must be the first program used of the five shown. It generates the Full Map file which is used as input to the DATACOND program. Once the Full Map file is created, the DATACOND program can be used to generate statistics and the PRINTMAP program can be used to print the Full Map file. The PRINTMAP program is not a necessary part of the statistical analysis, only a helpful utility program. The DATACOND program though is necessary and must be executed to precede to level 3. NOTE: If the Full Map file is ever changed the DATACOND program must be reexecuted to generate the proper statistics.

The DATACOND program creates a file containing the statistical data for the DATAGRAPH and PRINTSTAT programs. Once the DATACOND program has finished executing the DATAGRAPH and PRINTSTAT programs can be executed. The DATAGRAPH program will graph the results of the tests and the PRINTSTAT program will print the statistics of each test. The PRINTMAP program can still be used in level 3 for printing the Full Map file.

To log into the computer system type "HELLO TEST/MINE". All of the DASP programs are located in this account.

3.4.4 THE CREATEMAP PROGRAM

Introduction

The CREATEMAP program creates the Full Map file used by the DATACOND program. It requires a lot of data entry, which quickly becomes tedious, and some forethought on the data to be entered. Those reasons make it the most difficult step of the data analysis.

The purpose behind creating the Full Map is to allow the DATACOND program to look through the test data files, decide what lines contain alarm and reset reports, and associate each alarm and reset report to a particular point on the Mine Monitoring Test Facility (MMTF). [The test data files contain the ASCII alarm and reset reports produced by the System Under Test (SUT) during each test.] To understand the organization of the data to be entered requires a good knowledge of the Full Map mapping scheme.

The Mapping Scheme

For every point on the MMTF (420 points total) there exists 3 ten-character ASCII strings in the Full Map file. The strings are called the Point Name the Alarm Name, and the Reset Name. DATACOND uses these strings to scan the time stamped test data files for alarm and reset reports.

DATACOND first scans a line of a file for a Point Name. If one of the Point Names is contained in the line, DATACOND scans the line again for the corresponding Alarm Name. If the Alarm Name is contained in the line, DATACOND considers the line an alarm report. If the Alarm Name is not found, then DATACOND scans the line again for the corresponding Reset Name. If the Reset Name is contained in the line, DATACOND considers the line a reset report. If neither the Alarm Name or Reset Name are contained in the line, DATACOND continues on scanning the line for other possible Point Names. If no MMTF points match the line, the line is ignored.

The Point Name is generally a number that corresponds to a particular point on the SUT such as "0235". The Alarm Name is generally a word like "ON", "ALARM", or "HIGH" that signifies the point is in the alarm state. The Reset Name is generally a word like "OFF", "RESET", or "LOW" that signifies the point is in the reset state.

How To Choose The Point, Alarm, And Reset Names

The combination of the Point Name and the Alarm Name for a particular point must uniquely specify that point's alarm reports and nothing else. Similarly, the combination of the Point Name and the Reset Name for a particular point must uniquely specify that point's reset reports and nothing else. It does NOT mean that all Point Names must be unique from each other. It does NOT mean that all Alarm Names must be unique from each other. Nor does it mean that all Reset Names must be unique from each other. Only the combinations of the Point Names and Alarm Names and the combinations of the Point Names and Reset Names must all be unique.

Determining the uniqueness of the combinations requires the user to know all of the possible reports that are contained in the test data files. All systems under test uniquely identify each point and specify each point's status condition so that the operator of the system can correctly identify trouble areas in the mine. The next few pages contain two examples of Point, Alarm, and Reset Name selections for two different mine monitoring systems under test.

Example 1: SYSTEM B

SYSTEM B contained both analog and digital inputs. The following lists show sample lines from a SYSTEM B test data file:

SYSTEM B digital status reports:

```
16:34:28.10  0356 ON
16:34:28.10  0357 ON
16:34:28.10  0440 ON
16:34:28.10  0441 ON
16:38:00.50  0356 OFF
16:38:00.50  0357 OFF
16:38:00.50  0440 OFF
16:38:00.50  0441 OFF
```

SYSTEM B analog status reports:

```
16:34:28.10  0000 ALARM
16:34:28.10  0001 ALARM
16:34:28.10  0002 ALARM
16:34:28.10  0003 ALARM
16:38:00.50  0000 RESET
16:38:00.50  0001 RESET
16:38:00.50  0002 RESET
16:38:00.50  0003 RESET
```

The first column containing the number strings "16:34:28.10" and "16:38:00.50" is the internal time kept by SYSTEM B. The time signifies when the status change was detected by SYSTEM B. It is useless for determining the Point, Alarm, and Reset Names since the time values are unpredictable and differ from one test data file to another.

The second column contains the number strings "0356", "0357", "0440", and "0441" for the digital points and "0000", "0001", "0002", and "0003" for the analog points. These number strings are SYSTEM B's point numbers identifying which point changed status. The numbers are unique for every point on the system and follow a four digit octal numbering pattern. The point number strings differ from the time number strings because they lack colons and periods.

The analog point numbers ranged from "0000" thru "0237". The digital point numbers ranged from "0240" thru "0737". The point numbers were chosen as the Point Names making all Point Names for SYSTEM B unique. (See supplement 3.4A for a listing of the SYSTEM B Full Map. Note that not all of SYSTEM B's points were used.)

The third column contains the character strings "ON", "OFF", "ALARM", and "RESET". These strings signify the status of the points. "ON" signifies a digital point alarming and "OFF" signifies a digital point reseting. Similarly, "ALARM" signifies an analog point alarming and "RESET" signifies an analog point reseting. These character strings were chosen as Alarm Names and Reset Names for the Full Map file. "ON" was used for the digital Alarm Names "OFF" was used for the digital Reset Names, "ALARM" was used for the analog Alarm Names and "RESET" was used for the analog Reset Names. (See supplement 3.4A.)

The character strings chosen are not the shortest strings possible to use. The strings "ON", "OF", "AL", and "RE" could have been used in place of the existing ones.

Other messages produced by SYSTEM B are:

```
              JAN 00 0000              PAGE  48
*00:08:43.05   DIRECTROL SCAN FAULT
*00:10:33.75   DIRECTROL SCAN FAULT (RESET)
```

These messages do not contain any of the previously defined combinations of Point, Alarm, and Reset Names and are ignored by the DATACOND program.

Recapping Example 1: SYSTEM B's output lent itself nicely to the Full Map scheme. The four-digit point numbers became the Point Names for the MMTF. The point number's uniqueness gave each point on the MMTF a unique "Point Name - Alarm Name" and "Point Name - Reset Name" combination. The Point Names also stayed within the 10 character limit imposed by the DASP on all Names. The Alarm and Reset Names were also trivial and stayed within the 10 character limit.

A case to note, in SYSTEM B the Point Name came before the Alarm and Reset Names, which will probably be the case for most SUTs. The DATACOND program, though, is not restricted to this ordering. The DATACOND program scans the entire line for the Point Name. If the Point Name exists, it rescans the entire line for the Alarm Name or Reset Name. Therefore, the ordering of the Point, Alarm, and Reset Names is unrecognized by the program.

Example 2: SYSTEM D

SYSTEM D contained 1 analog and 64 digital inputs. The following lists show sample lines from a SYSTEM D test data file:

SYSTEM D digital status reports:

18:05:40	04/12/84	Postion	6	Set	M7(ON)	- Spare
18:05:40	04/12/84	Postion	6	Set	M8(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M1(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M2(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M3(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M4(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M5(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M6(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M7(ON)	- Spare
18:05:40	04/12/84	Postion	7	Set	M8(ON)	- Spare
18:06:22	04/12/84	Postion	6	Set	M7(OFF)	- Spare
18:06:22	04/12/84	Postion	6	Set	M8(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M1(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M2(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M3(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M4(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M5(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M6(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M7(OFF)	- Spare
18:06:22	04/12/84	Postion	7	Set	M8(OFF)	- Spare

SYSTEM D analog status reports:

18:05:40	04/12/84	Postion	1	Change	Analog - HI Set Point
18:06:22	04/12/84	Postion	1	Change	Analog Within Range

The first column containing the number strings "18:05:40" and "18:06:22" is the internal time kept by SYSTEM D. The time signifys when the status change was detected by SYSTEM D. It is useless for determining the Point, Alarm, and Reset Names since the time values are unpredictable and differ from one test data file to the next.

The second column contains the date of the test "04_/12_/84" as entered by the operator of SYSTEM D. It, too, is useless for determining the Point, Alarm, and Reset Names because it doesn't change from point to point and should be ignored.

The third column contains SYSTEM D's transponder number denoted by the character strings "Position 6" and "Position 7". On the system tested, there were 8 transponders ;abeled "Position 1" thru "Position 8". Each transponder contained 8 digital input points. One transponder, "Position 1", contained one analog input point.

The "Position 1" thru "Positon 8" strings were used as Point Names. The Point Names are not unique for each point but are unique for each transponder. Since the character strings are longer than 10 characters, they were shortened to "tion 1" thru "tion 8".

The Point Name strings chosen are not the smallest possible strings that could have been used. The smallest strings would have been similar to "1". The strings were left long for easy "human" recognition!

The fourth column contains the character strings "Set" and "Change". These strings are useless for Alarm and Reset Names because they never change from report to report and should be ignored. (They do signify the difference between the digital points and the analog points but so do the items in column 5.)

The fifth column contains the status of the points and a character or two defining what point it is within a transponder. For the digital points, the strings "M1(ON)", "M2(ON)", "M3(ON)", "M4(ON)", "M5(ON)", "M6(ON)", "M7(ON)", and "M8(ON)" were used as Alarm Names and "M1(OFF)", "M2(OFF)", "M3(OFF)", "M4(OFF)", "M5(OFF)", "M6(OFF)", "M7(OFF)", and "M8(OFF)" were used as the Reset Names. For the analog point, the string "HI Set" was used as the Alarm Name and "Within" was used as the Reset Name. (See supplement 3.4B for a listing of the SYSTEM D Full Map.)

The Alarm and Reset Name strings chosen are not the smallest of possible choices either. The digital strings could have been similar to "1(ON" and "1(OF" for point 1 of any of the 8 transponders. The analog strings could have been "H" and "W".

Other messages of concern produced by SYSTEM D are:

Time Date Label Alarm Cause

These messages do not contain any of the previously defined combinations of Point, Alarm, and Reset Names and are ignored by the DATACOND program.

Recapping Example 2: The Point, Alarm, and Reset Names did not follow the same uniqueness pattern as did SYSTEM B's but were still fairly simple to pick out. The Point Names for SYSTEM D only determined the uniqueness of the transponder, not of the individual points. The Alarm Names determined the individual points of an transponder and the alarm status but not the actual transponder. The Reset Names determined the individual points of an transponder and the reset status but not the actual transponder. The "Point Name - Alarm Name" combinations formed unique alarm reports for each point on the system. Similarly, the "Point Name - Reset Name" combinations formed unique reset reports for each point on the system. Again, the author will stress the fact that it's the uniqueness of the Name combinations that is the most important factor in choosing the Names. Also, all Names must be kept within the 10 character limit imposed by the program.

Other Items To Keep In Mind

Points on the MMTF that are not used during the testing of a mine monitoring system can be ignored by the DATACOND program. The section entitled The INSERT AND CHANGE POINT OPTIONS Menu discusses this function in more detail.

Each SEM uses 16 reed relays as controls for the 16 digital output points. The relays are mechanical in design and wear out over a period of time; therefore, the relays should be checked before and after a mine monitoring system is tested.

Check the relays before a system is tested and replace any defective ones. Check the relays after the system is tested, record the SEM number and point number of the defective relays, and replace any defective ones. The points containing the defective relays after the tests should not be included in the data analysis. If they are included, the data analysis will show that the SUT missed alarm signals which in reality it never received. Excluding the defective points will cause the data analysis to ignore them. The section entitled The INSERT AND CHANGE POINT Menu shows how to exclude (i.e. cause the DATACOND program to ignore) points when creating the Full Map file.

Using The CREATEMAP Program

The CREATEMAP program is a data entry program used for entering the Point, Alarm, and Reset Names. The program has the user associate the Names with a particular point on the Mine Monitoring Test Facility's Sensor Emulators. It contains several menus showing the user what options are available.

The remainder of this chapter assumes the user has already determined the Point, Alarm, and Reset Names and knows what Names correspond to each of the 420 output points on the MMTF. To run CREATEMAP type "RUN CREATEMAP".

The "System Filename"

At the beginning of the program, the first question asked is "What's the system file name?". (For the explanation of the system file name, see File Structure And Nomenclature.)##Type in the correct system file name to continue with the program.

If an incorrect file name is entered, the program will ask again for the file name. Up to five incorrect file names can be entered before the program automatically terminates. To exit the program from here, press <CTRL-Z>.

ThOPTIONS Menu

ı Full Map file was previously created for the system file name chıthe FILE OPTIONS menu will appear on the screen. This menu alıe user to create a completely new Full Map file update the cuı existing Full Map file or exit the program.

HIS MENU DOES NOT APPEAR ON THE SCREEN, DON'T BE ALARMED. It worear on the screen unless a Full Map file was previously created foıystem file.

hoose an option from the menu, press either 0, 1, or 2 and pree return key Pressing the return key without previously preany other keys produces the same result as pressing 0 and the rety Pressing <CTRL-Z> also produces the same result as pressing 0 a return key.

MEN FILE OPTIONS
0Exit program. (Or press <CTRL-Z>)
1Create a new mapping file.
2Jpdate the existing mapping file.

item 0 causes the program to exit and returns the user back to RSX.

item 1 erases all of the Point, Alarm, and Reset Names giving theı completely blank Full Map file.

item 2 reads the latest Full Map file into memory.

The IG OPTIONS Menu

:DITING OPTIONS menu is the real starting point of the CREA program and is therefore labeled LEVEL 1. The user can't retuıthe FILE OPTIONS menu or change the system file name They were preliminary queries before starting the actual editing.

here five menu items are open to the user. These options give the two exiting functions and three editing functions. To choose an oɾfrom the menu press either 1, 2, 3, 4, or 5 followed by the retur

MENU EDITING OPTIONS
1 :xit program and save new data.
2 :xit program without saving data.
3 List current point data for a SEM.
4 Insert o Change the point data of a SEM.
5 Insert o Change the default data of a SEM.

Menu item 1 causes the program to save the Full Map data currently in the memory of the computer into a new data file. The new data file becomes the Full Map file which will be used by the DATACOND program the next time DATACOND is executed using the same system file name. The old Full Map file, if one did actually exist, still exists but is ignored by all DASP programs.

Menu item 2 causes the program to throw away all editing that may have been performed while using the CREATEMAP program. No new Full Map file will be created. If no Full Map file existed, then there will still be no Full Map file. If one did exist, then it still exists as the Full Map file. The option is for users who mistakenly entered the wrong system file name or who only wish to practice using the CREATEMAP program and wish to exit without disturbing anything.

Menu item 3 allows the user to list out the current Point, Alarm, and Reset Names for each point of a particular SEM. A second menu, the LIST OPTIONS menu will appear asking the user to select a SEM.

Menu item 4 allows the user to insert and/or change the current Point, Alarm, and Reset Names of each point of a particular SEM. A second menu the INSERT AND CHANGE SEM OPTIONS menu, will appear asking the user to select a SEM followed by a third menu, the INSERT AND CHANGE POINT OPTIONS menu asking the user to select a point on the SEM.

Menu item 5 allows the user to insert and/or change the current default Point, Alarm, and Reset Names of a particular SEM. The default Names allow the user to quickly define repetitious Point, Alarm, and Reset Names within a SEM without having to enter them in for each point. A second menu the INSERT AND CHANGE SEM OPTIONS menu, will appear asking the user to select a SEM.

The LIST OPTIONS Menu

The LIST OPTIONS menu is a LEVEL 2 menu. (It falls one below the EDITING OPTIONS menu.) It allows the user to select the SEM number, from the 21 possible SEM numbers, to be listed. The listing contains the Point, Alarm, and Reset Names of all 16 digital points and all 4 analog points of a SEM.

The listing places the characters contained in each Name in between a greater than symbol ">" and a less than symbol "<". It also converts ASCII non-printable characters into spaces. The greater than and less than symbols are used so that the user can tell how many characters are actually in the Name and if any spaces are contained in the Name. The non-printable characters are converted to spaces to show that something does exist in those locations even though it can't be normally printed.

To select an option, enter a number from 0 to 21 and press the return key. Pressing the return key without pressing any other key will select menu item 0. Pressing <CTRL-Z> will return the user to the previous menu.

SEM #	LIST OPTIONS
0	List the data for all 21 SEM's.
1 thru 21	List the data for that SEM.
<CTRL-Z>	Return to previous menu.

Menu item 0 lists all the Point, Alarm, and Reset Names for all SEM's. It starts with SEM 1 and proceeds in order through SEM 21. At the end of each SEM listing the program automatically stops and waits for the user to press a <CTRL-Z>. The stop gives the user time to view the data before it scrolls off the screen. There's no way provided to stop the listing and return to the LIST OPTIONS menu (i.e the user must wait until all the data for all 21 SEM's has been displayed before returning to the LIST OPTIONS menu).

Menu items 1 thru 21 list all the Point, Alarm, and Reset Names for SEM's 1 thru 21. The user can choose the proper SEM number and the program will list the data for just that SEM. At the end of the SEM listing the program automatically stops and waits for the user to press a <CTRL-Z>. The stop gives the user time to view the data before it scrolls off the screen.

Menu item <CTRL-Z> returns the user back to the EDITING OPTIONS menu. All the user does is press <CTRL-Z>.

The INSERT AND CHANGE SEM OPTIONS Menu

The INSERT AND CHANGE SEM OPTIONS menu is a LEVEL 2 menu. (It falls one below the EDITING OPTIONS menu) It allows the user to select the SEM number, from the 21 possible SEM numbers to be edited. After choosing the proper SEM number the INSERT AND CHANGE POINT OPTIONS menu will ask the user which point on the SEM to edit.

To select an option enter a number from 1 to 21 and press the return key. Pressing <CTRL-Z> will return the user to the previous menu.

SEM #	INSERT AND CHANGE SEM OPTIONS
1 thru 21	Insert or Change the point data for that SEM.
<CTRL-Z>	Return to previous menu.

Menu items 1 thru 21 correspond to SEM's 1 thru 21. The user can choose the desired SEM number for the inserts and/or changes.

Menu item <CTRL-Z> returns the user back to the EDITING OPTIONS menu. All the user does is press <CTRL-Z>.

The INSERT AND CHANGE POINT OPTIONS Menu

The INSERT AND CHANGE POINTS OPTIONS menu is a LEVEL 3 menu. (It falls two below the EDITING OPTIONS menu.) It allows the user to select the point number, from the 20 possible points per SEM, to be edited. Point numbers 1 thru 16 correspond to the 16 digital points of the SEM. Point numbers 17 thru 20 correspond to the 4 analog points of the SEM.

After choosing the desired SEM point the CREATEMAP program will list out the current Point, Alarm, and Reset Names for that point. After listing out the current data, it will ask for the new Point Name then ask for the Alarm Name followed by the Reset Name. For each of the three questions there are three possible answers:

1. A new character or character string can be entered. Remember to end the character string by pressing the return key

2. The return key can be pressed without typing in any other characters, thus erasing any characters in the Name. If the Point Name is erased, the DATACOND program will ignore the SEM point when performing the data analysis.

3. A <CTRL-Z> can be pressed. The <CTRL-Z> will leave the data for that Name untouched and move the user down to the next question.

If the Point Name of a particular point contains no characters (i.e. is erased) the point is ignored by the DATACOND program even if there are characters contained in either or both of the Alarm and Reset Names. If the Point Name contains characters, then the point is used in the DATACOND program regardless of what is contained in the Alarm and Reset Names.

To select an option enter a number from 0 to 20 and press the return key Pressing the return key without pressing any other key will select menu item 0. Pressing <CTRL-Z> will return the user to the previous menu.

POINT #	INSERT AND CHANGE POINT OPTIONS
0	Insert or Change the data for all 20 points.
1 thru 20	Insert or Change the data for that point.
<CTRL-Z>	Return to previous menu.

Menu item 0 proceeds through all 20 points of the SEM before returning to the INSERT AND CHANGE POINT OPTIONS menu. The user can define all three Names for all 20 points before being asked to choose a new point.

Menu items 1 thru 20 correspond to points 1 thru 20 of the SEM. The user can choose the desired point number for the inserts and/or changes.

Menu item <CTRL-Z> returns the user back to the INSERT AND CHANGE SEM OPTIONS menu. All the user does is press <CTRL-Z>.

The INSERT AND CHANGE SEM DEFAULT OPTIONS Menu

The INSERT AND CHANGE SEM DEFAULT OPTIONS menu is a LEVEL 2 menu. (It falls one below the EDITING OPTIONS menu.) It allows the user to select the SEM number, from the 21 possible SEM numbers, to be edited.

Each SEM contains two default Name sets: one set for the 16 digital points and one set for the 4 analog points. The default Names speed the data entry process by allowing repetitive data to be entered only once per SEM.

After choosing the desired SEM number, the CREATEMAP program will list out the current digital default Point, Alarm, and Reset Names for the SEM. After listing out the current digital defaults, it will ask for the new Default Point Name then ask for the Default Alarm Name followed by the Default Reset Name. After entering in the digital default values the program will list out the current analog default Point, Alarm, and Reset Names. It will ask the same three questions for the analog Names as it did for the digital Names.

For each set of three questions there are three possible answers:

1. A new character or character string can be entered. Remember to end the character string by pressing the return key

2. The return key can be pressed without typing in any other characters, thus erasing any characters in the Name. If the Point Name is erased, the DATACOND program will ignore the SEM point when performing the data analysis.

3. A <CTRL-Z> can be pressed. The <CTRL-Z> will leave the data for that Name untouched and move the user down to the next question.

Example:

SYSTEM B uses "ON" for all digital Alarm Names and "OFF" for all digital Reset Names. By setting the digital Default Alarm Name to "ON" and the digital Default Reset Name to "OFF", the user is saved from typing "ON" and "OFF" 16 times each per SEM.

To select an option enter a number from 1 to 21 and press the return key. Pressing <CTRL-Z> will return the user to the previous menu.

SEM #	INSERT AND CHANGE SEM DEFAULT OPTIONS
1 thru 21	Insert or Change the point data for that SEM.
<CTRL-Z>	Return to previous menu.

Menu items 1 thru 21 correspond to SEM's 1 thru 21. The user can choose the desired SEM number for the default inserts and_/or changes.

Menu item <CTRL-Z> returns the user back to the EDITING OPTIONS menu. All the user does is press <CTRL-Z>.

3.4.5 THE DATACOND PROGRAM

Introduction

The DATACOND program performs the data analysis and data condensation, creates the graphing data, and stores the data in a single file. It is the heart of the entire Data Analysis Software Package (DASP). It is also the simplest program to use of the entire Package.

Using The DATACOND Program

The program requires almost no user interaction. The only item it requires is the system file name. (For the explanation of the system file name, see File Structure And Nomenclature.)##The user simply types in the correct system file name and the program will do the rest.

If an incorrect file name is entered, the program will ask again for the file name. Up to five incorrect file names can be entered before the program automatically terminates. To use DATACOND type "RUN DATACOND". DATACOND first performs the data analysis on the test data files saving the results in a separate file. It then uses the results to create the graphing data. The graphing data is added to the end of the separate file and the program terminates.

The program will run for approximately 30 minutes depending on the number of tests performed and the number of points used on the mine monitoring system. It requires about 2 minutes per Pulse Test test data file and about 15 seconds per Square Wave Test test data file.

3.4.6 THE DATAGRAPH PROGRAM

Introduction

DATAGRAPH provides four different graphs from the test results:

1. Percentage Of Missed Alarms versus SEM Alarms/Second.

2. Percentage Of Missed Alarms versus SEM Alarm Length

3. Average Alarm Time Delay versus SEM Alarms/Second.

4. Average Alarm Time Delay versus SEM Alarm Length.

The first and third graphs are derived from the Square Wave Tests. The second and fourth graphs are derived from the Pulse Tests. The data for the graphs comes from the data file produced by the DATACOND program.

The DATAGRAPH program requires the Tektronix 4025 Computer Display Terminal and 4662 Interactive Digital Plotter in order to obtain the proper performance results. The graphs can be viewed either on the screen of the terminal or on the plotter. The plotter and program give the user the freedom of locating the graphs anywhere on the plotter's surface the freedom to choose the sizes of the graphs, and the freedom to choose the pen colors of the graph axis and graph data.

Using The DATAGRAPH Program

The DATAGRAPH program is the second most difficult program to use. The difficulty is really not in using the program but in using the plotter. The users should have the TEK Operator's Manual for the 4662 Interactive Digital Plotter (With Option 31) handy while reading this chapter. To use DATAGRAPH type "RUN DATAGRAPH".

The "System Filename"

At the beginning of the program the first question asked is "What's the system file name?". (For the explanation of the system file name, see File Structure And Nomenclature.) Type in the correct system file name to continue with the program.

If an incorrect file name is entered, the program will ask again for the file name. Up to five incorrect file names can be entered before the program automatically terminates. To exit the program from here, press <CTRL-Z>.

The GRAPHS Menu

The GRAPHS menu is the starting point of the DATAGRAPH program. It contains five options for the user to choose from. The first option allows the user to exit the program. The other four options draw one of the four graphs discussed in the introduction. To choose an option from the menu, press either 0, 1, 2, 3, or 4 and press the return key. Pressing the return key without previously pressing any other key selects menu item 0. Pressing <CTRL-Z> also produces the same result as pressing 0 and the return key

MENU #	GRAPHS
0	Exit Program. (Or press <CTRL-Z>)
1	Percent Missed Alarms .vs. SEM Alarms/Second
2	Percent Missed Alarms vs. SEM Alarm Length
3	Average Alarm Time Delay vs. SEM Alarms/Second
4	Average Alarm Time Delay vs. SEM Alarm Length

Menu item 0 causes the program to exit and returns the user back to RSX-11M.

Menu item 1 tells the DATAGRAPH program to graph the Percent Missed Alarms versus SEM Alarms/Second graph and moves the user to the GRAPHING DEVICES menu.

Menu item 2 tells the DATAGRAPH program to graph the Percent Missed Alarms versus SEM Alarm Length graph and moves the user to the GRAPHING DEVICES menu.

Menu item 3 tells the DATAGRAPH program to graph the Average Alarm Time Delay versus SEM Alarms/Second graph and moves the user to the GRAPHING DEVICES menu.

Menu item 4 tells the DATAGRAPH program to graph the Average Alarm Time Delay versus SEM Alarms/Second graph and moves the user to the GRAPHING DEVICES menu.

The GRAPHING DEVICES Menu

The GRAPHING DEVICES menu gives the user the option of having the graph drawn on the Tektronix terminal or the Tektronix plotter. If the terminal is chosen, the graph chosen from the GRAPHS menu will be drawn on the screen and the program will return to the GRAPHS menu.

If the plotter is chosen, the program will ask what pen number to use when drawing the axis of the graph. At this point the user should place a sheet of paper on the plotting surface and set the lower-left and upper-right boundries on the plotter (see the Operator's Manual for the 4662 Interactive Digital Plotter section 2 FRONT PANEL SWITCHES AND INDICATORS, heading LOAD SWITCH and heading PAGE FORMATTING). If no pens are in the rotary pen turret, then pens should be inserted too (see the above mentioned Operator's Manual heading PROCEDURE FOR OPTION 31 EQUIPPED PLOTTERS). After answering the pen number question, the program will draw the axis and then ask what pen number to use when drawing the data of the graph. After answering the question the program will draw the data and return to the GRAPHS menu.

The pen numbers correspond to the numbers located on the rotary pen turret. If a valid pen number is chosen but there is no pen in the pen holder the program will still execute as if there was a pen. The program has no way of knowing if a pen exists in the pen holder or not.

Some things to keep in mind when using the plotter:

1. The labels on the graph's axis will reach to the outer limits of the boundries set on the plotter (i.e. the title of the graph will reach the top boundry, the x-axis label will reach the bottom boundry, the y-axis label will reach the left boundry, and the largest number of the x-axis will reach the right boundry).

2. Remove and recap the pens after using them so they won't dry out.

3. Test the pens on a piece of paper before placing them in the plotter.

To choose an option from the menu press either 0, 1, or 2 and press the return key. Pressing the return key without previously pressing 0, 1, or 2 produces the same result as pressing 0 and the return key Pressing <CTRL-Z> also produces the same result as pressing 0 and the return key.

MENU #	GRAPHING DEVICES
0	Exit Program. (Or press <CTRL-Z>)
1	Draw The Graph On The Terminal.
2	Draw The Graph On The Plotter.

Menu item 0 causes the program to exit and returns the user back to RSX-11M.

Menu item 1 causes the graph to be drawn on the terminal.

Menu item 2 causes the graph to be drawn on the plotter.

3.4.7 THE PRINTSTAT PROGRAM

Introduction

The PRINTSTAT program prints the statistics for each test plus prints the values of the data points for each of the four graphs. The test results are in the same order that the tests were performed.

Using The PRINTSTAT Program

The program requires almost no user interaction. The only item it requires is the system file name (For the explanation of the system file name, see File Structure And Nomenclature.) The user simply types in the correct system file name and the program will do the rest.

If an incorrect file name is entered, the program will ask again for the file name. Up to five incorrect file names can be entered before the program automatically terminates To use the PRINTSTAT program type "RUN PRINTSTAT".

Explanation Of The Statistics

The statistics are broken up into two sections. The first section contains the statistics for each test data file. The second section contains the data generated for the four graphs drawn by DATAGRAPH.

Test Statistics

Each test data file has a list of statistics associated with it. The list gives information on the type and length of the test plus the statistical results of the data analysis. There are two different types of lists due to the two different types of MMTF tests. The lists vary slightly and are given below:

TEST 1	
Test Type	Pulse
Data Filename	DL02:[001,011]S000TP000.DAT
Passed Fire Code	F
Passed Timing Aspect:	T
Total Buffers/SEM:	1
Total Points Checked:	348
Total Alarms Issued:	348
Total Valid Alarms:	309
Total Missed Alarms:	39
Total Extra Alarms:	0
Percent Missed Alarms:	11.21%
Total Time Delay:	0.673E+02 Seconds
Average Time Delay	0.218E+00 Seconds
SEM Alarm Length:	0.210E+00 Seconds
Tick Length:	0.500E-02 Seconds

TEST 2	
Test Type:	Square Wave
Data Filename	DL02:[001,011]S000TS000.DAT
Passed Fire Code	F
Passed Timing Aspect:	T
Total Buffers/SEM:	1
Total Points Checked:	1
Total Alarms Issued:	43
Total Valid Alarms:	39
Total Missed Alarms:	4
Total Extra Alarms:	0
Percent Missed Alarms:	9.30%
Total Time Delay	0.289E+03 Seconds
Average Time Delay:	0.741E+01 Seconds
SEM Alarms/Second:	0.500E+00 Seconds
Tick Length:	0.100E+01 Seconds

Only two differences appear between the two lists. The first difference appears in the Test Type entry. The second difference appears in the next to last line of each list - the Pulse Tests have an entry labeled SEM Alarm Length and the Square Tests have an entry labeled SEM Alarms/Second.

The following 16 items define the lines contained in both lists:

1. TESTXXXX - the test number where XXXX is a number between 1 and 2000. The number represents the test data file's relative position in the system file. The first test data file's name in the file is considered as TEST 1. The second name is considered as TEST 2 and so on. The positions in the system file also represent the order in which the tests were performed.

2. Test Type - defines the output data of the MMTF. The DASP can currently analyze two different output types - the square wave tests and the pulse tests. If the particular test data file has either an incorrect name or incorrect internal parameters the test type will read "Bad Test File Name or Bad First Record In Test File".

3. Data Filename - the name of the test data file as contained in the system file. The name is normally unimportant to the user but may be important to the system operator.

4. Passed Fire Code - tells the user if the test passed the rules of the National Fire Code of 1981 [4] The answer depends on both the timing aspect of the Fire Code and the number of missed alarms. If the SUT fails the timing aspect of the Fire Code or misses any alarm signals from the MMTF, the SUT fails the National Fire Code. A "T" (True) is considered as passing the Fire Code and an "F" (False) is considered as failing the Fire Code.

5. Passed Timing Aspect - tells the user if the SUT passed the timing aspect of the Fire Code. The Fire Code requires the SUT to respond to alarm signals within a certain amount of time. When one or more alarms don't respond within the proper time period, the SUT fails the timing aspect. A "T" (True) is considered as passing the timing aspect and an "F" (False) is considered as failing the timing aspect.

6. Total Buffers/SEM - the number of 86 tick buffers sent to each SEM during the test. This data is for the more experienced users who are interested in the amount of data sent to the SEM's.

7. Total Points Checked - the number of points used when doing the data analysis on the test data file. For all Square Wave Tests the value is 1 (only one point is used on the entire MMTF for Square Wave Tests). For the Pulse Tests the value should be equal to the number of Point Names in the Full Map file.

8. Total Alarms Issued - the cumulative number of alarm signals sent to the SUT by all SEM's combined. For square wave tests the value should always be greater than one and generally a multiple of 43. For pulse tests the value should be equal to the number of Point Names used in the Full Map File

9. Total Valid Alarms - the total number of valid alarm reports received from the SUT. A valid alarm report is a report which doesn't come before an alarm signal is sent by the MMTF, isn't a repeated alarm report, and is expected by the DATACOND program. The summation of the total missed alarms and total valid alarms should equal the number of issued alarms.

10. Total Missed Alarms - the total number of missed alarm signals. When a valid alarm report can't be found for a particular alarm signal the alarm is considered a missed alarm.

11. Total Extra Alarms - the total number of extra alarm reports. Extra alarms are comprised of alarm reports received before an alarm signal was sent by the MMTF to the SUT and of repetitive alarm reports. An alarm report is considered repetitive when one alarm report has already been associated with a particular alarm signal and a second alarm report is detected. The extra alarm total is independent of both the valid alarm total and the missed alarm total.

12. Percent Missed Alarms - the percentage of missed alarms. The value is found by comparing the number of total missed alarms to the number of total issued alarms.

13. Total Time Delay - For pulse tests: the summation of the time delays between each valid alarm report plus the time delay from the alarm signal to the first valid alarm report. For square wave tests: the summation of the time delays between the alarm signals and their associated alarm reports.

14. Average Time Delay - the average time delay of the valid alarms. The program divides the total time delay by the total valid alarms to obtain this statistic.

15. SEM Alarm Length - the total length of time the pulse test sent an alarm signal to a single point. Only one pulse is sent to each point during the test; therefore, the Alarm Length also represents the pulse width of the alarm signal. The SEM Alarm Length value also represents the x-axis position of the test's data in Graphs 2 and 4.

16. SEM Alarms/Second - the frequency of the alarm signals during a square wave test. It also represents the x-axis position of the test's data in Graphs 1 and 3.

17. Tick Length - the length of a single tick of the test. There are 86 ticks in a buffer and one or more buffers in a test. A single tick can range from two milliseconds to several minutes and is therefore an arbitrary length of time defined at the time of the test.

Graph Data

The DATACOND program generates four different graphs from the test results and labels them Graph 1, 2, 3, and 4. The generated data for each graph is listed out following the listing of the test results. Each graph listing contains the graph number and title what tests were used to derive the graphing data, the total number of data points, the minimum and maximum x-axis and y-axis values and the actual x and y data points. A sample listing of the graphing data is given below:

Graph 1:	Percentage Of Missed Alarms .vs. SEM Alarms/Second
Derivation:	All Square Wave Tests
Total Data Points:	2
Maximum X-axis:	0.250E+01
Minimum X-axis:	0.000E+00
Maximum Y-axis:	0.100E+03
Minimum Y-axis:	0.000E+00
X Data Point 1:	0.100E+00
Y Data Point 1:	0.233E+01
X Data Point 2:	0.250E+00
Y Data Point 2:	0.116E+01

3.4.8 THE PRINTMAP PROGRAM

Introduction

The PRINTMAP program prints the Full Map on a terminal or a printer. It provides the user with a way of listing the Full Map file without using the CREATEMAP program. The listing will look similar to the CREATEMAP program's listing. The listing though won't stop after printing a complete SEM worth of data, but will continue listing through SEM 21.

Like the CREATEMAP program, PRINTMAP places the characters contained in each Name in between a greater than symbol ">" and a less than symbol "<". It also converts ASCII non-printable characters into spaces. The symbols are used so that the user can tell how many characters are actually in the Name and if any spaces are contained in the Name. The non-printable characters are converted into spaces to show that something does exist in those locations even though it can't be normally printed.

A sample output from the program is shown below:

SEM # 1	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0320<	>ON<	>OFF<
POINT # 2	DIGITAL	>0321<	>ON<	>OFF<

If the user is using a printer, two SEM's worth of data will be printed per page. If the user is using a terminal approximately four blank lines will separate each set of data.

Using The PRINTMAP Program

The program requires almost no user interaction. The only item it requires is the system file name. (For the explanation of the system file name, see File Structure And Nomenclature.) The user simply types in the correct system file name and the program will do the rest.

If an incorrect file name is entered, the program will ask again for the file name. Up to five incorrect file names can be entered before the program automatically terminates. To use the PRINTMAP program type "RUN PRINTMAP".

After entering the system file name the program lists the data contained in the Full Map file. Upon completing the listing the program asks again for a system file name.

3.4.9 THE PRINTTEST PROGRAM

Introduction

The PRINTTEST program prints the contents of the test data files and the contents of the system file. It allows the user to view the test data files for a particular mine monitoring system. The program can be used during and after the testing of a mine monitoring system (as long as the system file isn't being used by another program).

PRINTTEST provides two different methods of retrieving the test data files for viewing. The first method lets the user enter a test number to retrieve the file. The test numbers can be found when listing the system file and from the output of the PRINTSTAT program. The second method lets the user enter the test data file name to retrieve the file.

Using The PRINTTEST Program

The PRINTTEST program contains four menus. The following sections explain how each menu works. To use the program type "RUN PRINTTEST".

The FUNCTION OPTIONS Menu

The FUNCTION OPTIONS menu is the starting point of the PRINTTEST program. From this menu the user can choose to either print the data contained in the test data files or print the contents of a system file. To select an option press either 0, 1, or 2 and then press the return key. Pressing the return key without pressing any other key will select menu item 0.

MENU #	FUNCTION OPTIONS
0	Exit program. (Or press <CTRL-Z>)
1	Print a test data file.
2	Print the system file.

Menu item 0 causes the program to exit and returns the user back to RSX-11M.

Menu item 1 allows the user to print a test data file. A second menu, the RETRIEVAL METHODS menu, will appear asking the user to choose a file retrieval method.

Menu item 2 allows the user to print the contents of the system file. After selecting item 2, the program asks the user for the system file name. (For the explanation of the system file name see File Structure And Nomenclature.) Type in the correct system file name pressing the return key last. The contents of the system file will be printed and the program will return back to the FUNCTION OPTIONS menu.

The RETRIEVAL METHODS Menu

The RETRIEVAL METHODS menu allows the user to choose between two different file retrieval methods. The first method lets the user enter a test number to retrieve the file. The test numbers can be found when listing the system file and from the output of the PRINTSTAT program. The second method lets the user enter the test data file name to retrieve the file. The first method restricts the user to a particular system file, whereas, the second method does not.

To select one of the options press either 0, 1, 2, or 3 and then press the return key. Pressing the return key without pressing any other key will select menu item 0.

MENU #	RETRIEVAL METHODS
0	Return to the FUNCTION OPTIONS menu.
1	You can enter in a test number and let the program search for the proper file name.
2	You can specify the file name.
3	Exit program. (Or press <CTRL-Z>)

Menu item 0 returns the user to the FUNCTION OPTIONS menu.

Menu item 1 allows the user to use the test number retrieval method for printing a test data file. A third menu, the TEST DATA FILES menu, will appear giving the range of valid test numbers to be used.

Menu item 2 allows the user to specify the actual test data file name. The file names are given by both the PRINTSTAT program and by this program. After selecting item 2, the program will ask for the test data file name. Type in a correct file name pressing the return key last. If the name is correct, a third menu, the PRINTING STYLE menu, will appear. If the name is incorrect, the program will return to the RETRIEVAL METHODS menu.

Menu item 3 causes the program to exit and returns the user back to RSX-11M.

The TEST DATA FILES Menu

The TEST DATA FILES menu allows the user to enter the test number to retrieve a particular test data file. The menu contains the range of the possible test numbers to enter. Simply enter the test number and press return. If no test data files are contained in the system file, the program automatically returns to the RETRIEVAL METHODS menu.

The test numbers and corresponding test data file names can be obtained by listing the contents of the system file and by running the PRINTSTAT program. (NOTE: Pressing the return key without pressing any other key will select menu item 0.)

MENU #	TEST DATA FILES
0	Return to the RETRIEVAL METHODS menu.
1 thru XXX	Print the corresponding Test data file.
<CTRL-Z>	Exit program. (Or press <CTRL-Z>)

where XXX can be a number from 1 to 999.

Menu item 0 returns the user to the FUNCTION OPTIONS menu.

Menu items 1 thru XXX are the test numbers corresponding to the test data files. Entering a number from 1 to XXX causes the program to search and open the proper file contained in the system file. Once the file is opened, the fourth menu, the PRINTING STYLE menu will appear.

Menu item <CTRL-Z> causes the program to exit and returns the user back to RSX-11M.

The PRINTING STYLE Menu

The PRINTING STYLE menu gives the user the option of printing the time stamp along with each line of the file or just printing each line of the file. The time stamp shows the time the Dynamic Test Controller (DTC) received the line from the SUT relative to the start of the test.

The time stamp is broken into seven units - year, month, day hour, minute, second, tick. The seven units are printed one line before each line of the file. Years and months should always be zero because the DTC only keeps track up to days but provides for years and months. (For the curious, a second consists of 500 ticks · Here the ticks are a fixed length of time and are NOT the same ticks referred to in the Tick Length line of the output of the PRINTSTAT program.) To select one of the options press either 0, 1, 2, or 3 and then press the return key. Pressing the return key without pressing any other key will select menu item 0.

MENU #	PRINTING STYLE
0	Return to the previous menu.
1	Print the time stamp along with each line.
2	Ignore the time stamp and just print each line.
3	Exit program. (Or press <CTRL-Z>)

Menu item 0 returns the user back to the previous menu. This option allows the user to change his or her mind before listing a file that may take a few minutes to completely list out.

Menu item 1 will cause the time stamp to be printed along with each line of the test data file.

Menu item 2 will print each line of the test data file without printing the time stamp.

Menu item 3 causes the program to exit and returns the user back to RSX-11M.

3.4A THE SYSTEM B FULL MAP

SEM # 1	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0240<	>ON<	>OFF<
POINT # 2	DIGITAL	>0241<	>ON<	>OFF<
POINT # 3	DIGITAL	>0242<	>ON<	>OFF<
POINT # 4	DIGITAL	>0243<	>ON<	>OFF<
POINT # 5	DIGITAL	>0244<	>ON<	>OFF<
POINT # 6	DIGITAL	>0245<	>ON<	>OFF<
POINT # 7	DIGITAL	>0246<	>ON<	>OFF<
POINT # 8	DIGITAL	>0247<	>ON<	>OFF<
POINT # 9	DIGITAL	>0250<	>ON<	>OFF<
POINT #10	DIGITAL	>0251<	>ON<	>OFF<
POINT #11	DIGITAL	>0252<	>ON<	>OFF<
POINT #12	DIGITAL	>0253<	>ON<	>OFF<
POINT #13	DIGITAL	>0254<	>ON<	>OFF<
POINT #14	DIGITAL	>0255<	>ON<	>OFF<
POINT #15	DIGITAL	>0256<	>ON<	>OFF<
POINT #16	DIGITAL	>0257<	>ON<	>OFF<
POINT #17	ANALOG	>0000<	>ALARM<	>RESET<
POINT #18	ANALOG	>0001<	>ALARM<	>RESET<
POINT #19	ANALOG	>0002<	>ALARM<	>RESET<
POINT #20	ANALOG	>0003<	>ALARM<	>RESET<

SEM # 2	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0260<	>ON<	>OFF<
POINT # 2	DIGITAL	>0261<	>ON<	>OFF<
POINT # 3	DIGITAL	>0262<	>ON<	>OFF<
POINT # 4	DIGITAL	>0263<	>ON<	>OFF<
POINT # 5	DIGITAL	>0264<	>ON<	>OFF<
POINT # 6	DIGITAL	>0265<	>ON<	>OFF<
POINT # 7	DIGITAL	>0266<	>ON<	>OFF<
POINT # 8	DIGITAL	>0267<	>ON<	>OFF<
POINT # 9	DIGITAL	>0270<	>ON<	>OFF<
POINT #10	DIGITAL	>0271<	>ON<	>OFF<
POINT #11	DIGITAL	>0272<	>ON<	>OFF<
POINT #12	DIGITAL	>0273<	>ON<	>OFF<
POINT #13	DIGITAL	>0274<	>ON<	>OFF<
POINT #14	DIGITAL	>0275<	>ON<	>OFF<
POINT #15	DIGITAL	>0276<	>ON<	>OFF<
POINT #16	DIGITAL	>0277<	>ON<	>OFF<
POINT #17	ANALOG	>0010<	>ALARM<	>RESET<
POINT #18	ANALOG	>0011<	>ALARM<	>RESET<
POINT #19	ANALOG	>0012<	>ALARM<	>RESET<
POINT #20	ANALOG	>0013<	>ALARM<	>RESET<

SEM # 3	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0300<	>ON<	>OFF<
POINT # 2	DIGITAL	>0301<	>ON<	>OFF<
POINT # 3	DIGITAL	>0302<	>ON<	>OFF<
POINT # 4	DIGITAL	>0303<	>ON<	>OFF<
POINT # 5	DIGITAL	>0304<	>ON<	>OFF<
POINT # 6	DIGITAL	>0305<	>ON<	>OFF<
POINT # 7	DIGITAL	>0306<	>ON<	>OFF<
POINT # 8	DIGITAL	>0307<	>ON<	>OFF<
POINT # 9	DIGITAL	>0310<	>ON<	>OFF<
POINT #10	DIGITAL	>0311<	>ON<	>OFF<
POINT #11	DIGITAL	>0312<	>ON<	>OFF<
POINT #12	DIGITAL	>0313<	>ON<	>OFF<
POINT #13	DIGITAL	>0314<	>ON<	>OFF<
POINT #14	DIGITAL	>0315<	>ON<	>OFF<
POINT #15	DIGITAL	>0316<	>ON<	>OFF<
POINT #16	DIGITAL	>0317<	>ON<	>OFF<
POINT #17	ANALOG	>0020<	>ALARM<	>RESET<
POINT #18	ANALOG	>0021<	>ALARM<	>RESET<
POINT #19	ANALOG	>0022<	>ALARM<	>RESET<
POINT #20	ANALOG	>0023<	>ALARM<	>RESET<

SEM # 4	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0320<	>ON<	>OFF<
POINT # 2	DIGITAL	>0321<	>ON<	>OFF<
POINT # 3	DIGITAL	>0322<	>ON<	>OFF<
POINT # 4	DIGITAL	>0323<	>ON<	>OFF<
POINT # 5	DIGITAL	>0324<	>ON<	>OFF<
POINT # 6	DIGITAL	>0325<	>ON<	>OFF<
POINT # 7	DIGITAL	>0326<	>ON<	>OFF<
POINT # 8	DIGITAL	>0327<	>ON<	>OFF<
POINT # 9	DIGITAL	>0330<	>ON<	>OFF<
POINT #10	DIGITAL	>0331<	>ON<	>OFF<
POINT #11	DIGITAL	>0332<	>ON<	>OFF<
POINT #12	DIGITAL	>0333<	>ON<	>OFF<
POINT #13	DIGITAL	>0334<	>ON<	>OFF<
POINT #14	DIGITAL	>0335<	>ON<	>OFF<
POINT #15	DIGITAL	>0336<	>ON<	>OFF<
POINT #16	DIGITAL	>0337<	>ON<	>OFF<
POINT #17	ANALOG	>0030<	>ALARM<	>RESET<
POINT #18	ANALOG	>0031<	>ALARM<	>RESET<
POINT #19	ANALOG	>0032<	>ALARM<	>RESET<
POINT #20	ANALOG	>0033<	>ALARM<	>RESET<

SEM # 5	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0340<	>ON<	>OFF<
POINT # 2	DIGITAL	>0341<	>ON<	>OFF<
POINT # 3	DIGITAL	>0342<	>ON<	>OFF<
POINT # 4	DIGITAL	>0343<	>ON<	>OFF<
POINT # 5	DIGITAL	>0344<	>ON<	>OFF<
POINT # 6	DIGITAL	>0345<	>ON<	>OFF<
POINT # 7	DIGITAL	>0346<	>ON<	>OFF<
POINT # 8	DIGITAL	>0347<	>ON<	>OFF<
POINT # 9	DIGITAL	>0350<	>ON<	>OFF<
POINT #10	DIGITAL	>0351<	>ON<	>OFF<
POINT #11	DIGITAL	>0352<	>ON<	>OFF<
POINT #12	DIGITAL	>0353<	>ON<	>OFF<
POINT #13	DIGITAL	>0354<	>ON<	>OFF<
POINT #14	DIGITAL	>0355<	>ON<	>OFF<
POINT #15	DIGITAL	>0356<	>ON<	>OFF<
POINT #16	DIGITAL	>0357<	>ON<	>OFF<
POINT #17	ANALOG	>0040<	>ALARM<	>RESET<
POINT #18	ANALOG	>0041<	>ALARM<	>RESET<
POINT #19	ANALOG	>0042<	>ALARM<	>RESET<
POINT #20	ANALOG	>0043<	>ALARM<	>RESET<

SEM # 6	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0360<	>ON<	>OFF<
POINT # 2	DIGITAL	>0361<	>ON<	>OFF<
POINT # 3	DIGITAL	>0362<	>ON<	>OFF<
POINT # 4	DIGITAL	>0363<	>ON<	>OFF<
POINT # 5	DIGITAL	><	><	><
POINT # 6	DIGITAL	>0365<	>ON<	>OFF<
POINT # 7	DIGITAL	>0366<	>ON<	>OFF<
POINT # 8	DIGITAL	>0367<	>ON<	>OFF<
POINT # 9	DIGITAL	><	><	><
POINT #10	DIGITAL	>0371<	>ON<	>OFF<
POINT #11	DIGITAL	>0372<	>ON<	>OFF<
POINT #12	DIGITAL	>0373<	>ON<	>OFF<
POINT #13	DIGITAL	>0374<	>ON<	>OFF<
POINT #14	DIGITAL	>0375<	>ON<	>OFF<
POINT #15	DIGITAL	>0376<	>ON<	>OFF<
POINT #16	DIGITAL	>0377<	>ON<	>OFF<
POINT #17	ANALOG	>0050<	>ALARM<	>RESET<
POINT #18	ANALOG	>0051<	>ALARM<	>RESET<
POINT #19	ANALOG	>0052<	>ALARM<	>RESET<
POINT #20	ANALOG	>0053<	>ALARM<	>RESET<

SEM # 7	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0400<	>ON<	>OFF<
POINT # 2	DIGITAL	>0401<	>ON<	>OFF<
POINT # 3	DIGITAL	>0402<	>ON<	>OFF<
POINT # 4	DIGITAL	>0403<	>ON<	>OFF<
POINT # 5	DIGITAL	>0404<	>ON<	>OFF<
POINT # 6	DIGITAL	>0405<	>ON<	>OFF<
POINT # 7	DIGITAL	>0406<	>ON<	>OFF<
POINT # 8	DIGITAL	>0407<	>ON<	>OFF<
POINT # 9	DIGITAL	>0410<	>ON<	>OFF<
POINT #10	DIGITAL	>0411<	>ON<	>OFF<
POINT #11	DIGITAL	>0412<	>ON<	>OFF<
POINT #12	DIGITAL	>0413<	>ON<	>OFF<
POINT #13	DIGITAL	>0414<	>ON<	>OFF<
POINT #14	DIGITAL	>0415<	>ON<	>OFF<
POINT #15	DIGITAL	>0416<	>ON<	>OFF<
POINT #16	DIGITAL	>0417<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM # 8	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0420<	>ON<	>OFF<
POINT # 2	DIGITAL	>0421<	>ON<	>OFF<
POINT # 3	DIGITAL	>0422<	>ON<	>OFF<
POINT # 4	DIGITAL	>0423<	>ON<	>OFF<
POINT # 5	DIGITAL	>0424<	>ON<	>OFF<
POINT # 6	DIGITAL	>0425<	>ON<	>OFF<
POINT # 7	DIGITAL	><	><	><
POINT # 8	DIGITAL	><	><	><
POINT # 9	DIGITAL	>0430<	>ON<	>OFF<
POINT #10	DIGITAL	>0431<	>ON<	>OFF<
POINT #11	DIGITAL	>0432<	>ON<	>OFF<
POINT #12	DIGITAL	>0433<	>ON<	>OFF<
POINT #13	DIGITAL	>0434<	>ON<	>OFF<
POINT #14	DIGITAL	>0435<	>ON<	>OFF<
POINT #15	DIGITAL	>0436<	>ON<	>OFF<
POINT #16	DIGITAL	>0437<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM # 9	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0440<	>ON<	>OFF<
POINT # 2	DIGITAL	>0441<	>ON<	>OFF<
POINT # 3	DIGITAL	>0442<	>ON<	>OFF<
POINT # 4	DIGITAL	>0443<	>ON<	>OFF<
POINT # 5	DIGITAL	>0444<	>ON<	>OFF<
POINT # 6	DIGITAL	>0445<	>ON<	>OFF<
POINT # 7	DIGITAL	>0446<	>ON<	>OFF<
POINT # 8	DIGITAL	>0447<	>ON<	>OFF<
POINT # 9	DIGITAL	>0450<	>ON<	>OFF<
POINT #10	DIGITAL	>0451<	>ON<	>OFF<
POINT #11	DIGITAL	>0452<	>ON<	>OFF<
POINT #12	DIGITAL	>0453<	>ON<	>OFF<
POINT #13	DIGITAL	>0454<	>ON<	>OFF<
POINT #14	DIGITAL	>0455<	>ON<	>OFF<
POINT #15	DIGITAL	>0456<	>ON<	>OFF<
POINT #16	DIGITAL	>0457<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #10	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0460<	>ON<	>OFF<
POINT # 2	DIGITAL	>0461<	>ON<	>OFF<
POINT # 3	DIGITAL	>0462<	>ON<	>OFF<
POINT # 4	DIGITAL	>0463<	>ON<	>OFF<
POINT # 5	DIGITAL	>0464<	>ON<	>OFF<
POINT # 6	DIGITAL	>0465<	>ON<	>OFF<
POINT # 7	DIGITAL	>0466<	>ON<	>OFF<
POINT # 8	DIGITAL	>0467<	>ON<	>OFF<
POINT # 9	DIGITAL	>0470<	>ON<	>OFF<
POINT #10	DIGITAL	>0471<	>ON<	>OFF<
POINT #11	DIGITAL	>0472<	>ON<	>OFF<
POINT #12	DIGITAL	>0473<	>ON<	>OFF<
POINT #13	DIGITAL	><	><	><
POINT #14	DIGITAL	>0475<	>ON<	>OFF<
POINT #15	DIGITAL	>0476<	>ON<	>OFF<
POINT #16	DIGITAL	>0477<	>ON<	>OFF<
POINT #17	ANALOG	>0110<	>ALARM<	>RESET<
POINT #18	ANALOG	>0111<	>ALARM<	>RESET<
POINT #19	ANALOG	>0112<	>ALARM<	>RESET<
POINT #20	ANALOG	>0113<	>ALARM<	>RESET<

SEM #11	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0500<	>ON<	>OFF<
POINT # 2	DIGITAL	>0501<	>ON<	>OFF<
POINT # 3	DIGITAL	>0502<	>ON<	>OFF<
POINT # 4	DIGITAL	>0503<	>ON<	>OFF<
POINT # 5	DIGITAL	>0504<	>ON<	>OFF<
POINT # 6	DIGITAL	>0505<	>ON<	>OFF<
POINT # 7	DIGITAL	>0506<	>ON<	>OFF<
POINT # 8	DIGITAL	>0507<	>ON<	>OFF<
POINT # 9	DIGITAL	>0510<	>ON<	>OFF<
POINT #10	DIGITAL	>0511<	>ON<	>OFF<
POINT #11	DIGITAL	>0512<	>ON<	>OFF<
POINT #12	DIGITAL	>0513<	>ON<	>OFF<
POINT #13	DIGITAL	>0514<	>ON<	>OFF<
POINT #14	DIGITAL	>0515<	>ON<	>OFF<
POINT #15	DIGITAL	>0516<	>ON<	>OFF<
POINT #16	DIGITAL	>0517<	>ON<	>OFF<
POINT #17	ANALOG	>0120<	>ALARM<	>RESET<
POINT #18	ANALOG	>0121<	>ALARM<	>RESET<
POINT #19	ANALOG	>0122<	>ALARM<	>RESET<
POINT #20	ANALOG	>0123<	>ALARM<	>RESET<

SEM #12	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0520<	>ON<	>OFF<
POINT # 2	DIGITAL	>0521<	>ON<	>OFF<
POINT # 3	DIGITAL	>0522<	>ON<	>OFF<
POINT # 4	DIGITAL	>0523<	>ON<	>OFF<
POINT # 5	DIGITAL	>0524<	>ON<	>OFF<
POINT # 6	DIGITAL	>0525<	>ON<	>OFF<
POINT # 7	DIGITAL	><	><	><
POINT # 8	DIGITAL	>0527<	>ON<	>OFF<
POINT # 9	DIGITAL	>0530<	>ON<	>OFF<
POINT #10	DIGITAL	>0531<	>ON<	>OFF<
POINT #11	DIGITAL	>0532<	>ON<	>OFF<
POINT #12	DIGITAL	>0533<	>ON<	>OFF<
POINT #13	DIGITAL	>0534<	>ON<	>OFF<
POINT #14	DIGITAL	>0535<	>ON<	>OFF<
POINT #15	DIGITAL	>0536<	>ON<	>OFF<
POINT #16	DIGITAL	>0537<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #13	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0540<	>ON<	>OFF<
POINT # 2	DIGITAL	>0541<	>ON<	>OFF<
POINT # 3	DIGITAL	>0542<	>ON<	>OFF<
POINT # 4	DIGITAL	>0543<	>ON<	>OFF<
POINT # 5	DIGITAL	>0544<	>ON<	>OFF<
POINT # 6	DIGITAL	>0545<	>ON<	>OFF<
POINT # 7	DIGITAL	>0546<	>ON<	>OFF<
POINT # 8	DIGITAL	>0547<	>ON<	>OFF<
POINT # 9	DIGITAL	>0550<	>ON<	>OFF<
POINT #10	DIGITAL	>0551<	>ON<	>OFF<
POINT #11	DIGITAL	>0552<	>ON<	>OFF<
POINT #12	DIGITAL	>0553<	>ON<	>OFF<
POINT #13	DIGITAL	>0554<	>ON<	>OFF<
POINT #14	DIGITAL	>0555<	>ON<	>OFF<
POINT #15	DIGITAL	>0556<	>ON<	>OFF<
POINT #16	DIGITAL	>0557<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #14	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0560<	>ON<	>OFF<
POINT # 2	DIGITAL	>0561<	>ON<	>OFF<
POINT # 3	DIGITAL	>0562<	>ON<	>OFF<
POINT # 4	DIGITAL	>0563<	>ON<	>OFF<
POINT # 5	DIGITAL	><	><	><
POINT # 6	DIGITAL	>0565<	>ON<	>OFF<
POINT # 7	DIGITAL	>0566<	>ON<	>OFF<
POINT # 8	DIGITAL	>0567<	>ON<	>OFF<
POINT # 9	DIGITAL	>0570<	>ON<	>OFF<
POINT #10	DIGITAL	>0571<	>ON<	>OFF<
POINT #11	DIGITAL	>0572<	>ON<	>OFF<
POINT #12	DIGITAL	>0573<	>ON<	>OFF<
POINT #13	DIGITAL	>0574<	>ON<	>OFF<
POINT #14	DIGITAL	>0575<	>ON<	>OFF<
POINT #15	DIGITAL	>0576<	>ON<	>OFF<
POINT #16	DIGITAL	>0577<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #15	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0600<	>ON<	>OFF<
POINT # 2	DIGITAL	>0601<	>ON<	>OFF<
POINT # 3	DIGITAL	>0602<	>ON<	>OFF<
POINT # 4	DIGITAL	>0603<	>ON<	>OFF<
POINT # 5	DIGITAL	>0604<	>ON<	>OFF<
POINT # 6	DIGITAL	>0605<	>ON<	>OFF<
POINT # 7	DIGITAL	>0606<	>ON<	>OFF<
POINT # 8	DIGITAL	>0607<	>ON<	>OFF<
POINT # 9	DIGITAL	>0610<	>ON<	>OFF<
POINT #10	DIGITAL	>0611<	>ON<	>OFF<
POINT #11	DIGITAL	>0612<	>ON<	>OFF<
POINT #12	DIGITAL	>0613<	>ON<	>OFF<
POINT #13	DIGITAL	>0614<	>ON<	>OFF<
POINT #14	DIGITAL	>0615<	>ON<	>OFF<
POINT #15	DIGITAL	>0616<	>ON<	>OFF<
POINT #16	DIGITAL	>0617<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #16	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0620<	>ON<	>OFF<
POINT # 2	DIGITAL	>0621<	>ON<	>OFF<
POINT # 3	DIGITAL	>0622<	>ON<	>OFF<
POINT # 4	DIGITAL	>0623<	>ON<	>OFF<
POINT # 5	DIGITAL	>0624<	>ON<	>OFF<
POINT # 6	DIGITAL	>0625<	>ON<	>OFF<
POINT # 7	DIGITAL	>0626<	>ON<	>OFF<
POINT # 8	DIGITAL	>0627<	>ON<	>OFF<
POINT # 9	DIGITAL	>0630<	>ON<	>OFF<
POINT #10	DIGITAL	>0631<	>ON<	>OFF<
POINT #11	DIGITAL	>0632<	>ON<	>OFF<
POINT #12	DIGITAL	>0633<	>ON<	>OFF<
POINT #13	DIGITAL	>0634<	>ON<	>OFF<
POINT #14	DIGITAL	>0635<	>ON<	>OFF<
POINT #15	DIGITAL	>0636<	>ON<	>OFF<
POINT #16	DIGITAL	>0637<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #17	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0640<	>ON<	>OFF<
POINT # 2	DIGITAL	>0641<	>ON<	>OFF<
POINT # 3	DIGITAL	>0642<	>ON<	>OFF<
POINT # 4	DIGITAL	>0643<	>ON<	>OFF<
POINT # 5	DIGITAL	>0644<	>ON<	>OFF<
POINT # 6	DIGITAL	>0645<	>ON<	>OFF<
POINT # 7	DIGITAL	>0646<	>ON<	>OFF<
POINT # 8	DIGITAL	>0647<	>ON<	>OFF<
POINT # 9	DIGITAL	>0650<	>ON<	>OFF<
POINT #10	DIGITAL	>0651<	>ON<	>OFF<
POINT #11	DIGITAL	>0652<	>ON<	>OFF<
POINT #12	DIGITAL	>0653<	>ON<	>OFF<
POINT #13	DIGITAL	>0654<	>ON<	>OFF<
POINT #14	DIGITAL	>0655<	>ON<	>OFF<
POINT #15	DIGITAL	>0656<	>ON<	>OFF<
POINT #16	DIGITAL	>0657<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #18	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0660<	>ON<	>OFF<
POINT # 2	DIGITAL	>0661<	>ON<	>OFF<
POINT # 3	DIGITAL	>0662<	>ON<	>OFF<
POINT # 4	DIGITAL	>0663<	>ON<	>OFF<
POINT # 5	DIGITAL	>0664<	>ON<	>OFF<
POINT # 6	DIGITAL	>0665<	>ON<	>OFF<
POINT # 7	DIGITAL	><	><	><
POINT # 8	DIGITAL	>0667<	>ON<	>OFF<
POINT # 9	DIGITAL	>0670<	>ON<	>OFF<
POINT #10	DIGITAL	>0671<	>ON<	>OFF<
POINT #11	DIGITAL	>0672<	>ON<	>OFF<
POINT #12	DIGITAL	>0673<	>ON<	>OFF<
POINT #13	DIGITAL	>0674<	>ON<	>OFF<
POINT #14	DIGITAL	>0675<	>ON<	>OFF<
POINT #15	DIGITAL	>0676<	>ON<	>OFF<
POINT #16	DIGITAL	>0677<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #19	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0700<	>ON<	>OFF<
POINT # 2	DIGITAL	>0701<	>ON<	>OFF<
POINT # 3	DIGITAL	>0702<	>ON<	>OFF<
POINT # 4	DIGITAL	>0703<	>ON<	>OFF<
POINT # 5	DIGITAL	>0704<	>ON<	>OFF<
POINT # 6	DIGITAL	>0705<	>ON<	>OFF<
POINT # 7	DIGITAL	>0706<	>ON<	>OFF<
POINT # 8	DIGITAL	>0707<	>ON<	>OFF<
POINT # 9	DIGITAL	>0710<	>ON<	>OFF<
POINT #10	DIGITAL	>0711<	>ON<	>OFF<
POINT #11	DIGITAL	>0712<	>ON<	>OFF<
POINT #12	DIGITAL	>0713<	>ON<	>OFF<
POINT #13	DIGITAL	>0714<	>ON<	>OFF<
POINT #14	DIGITAL	>0715<	>ON<	>OFF<
POINT #15	DIGITAL	>0716<	>ON<	>OFF<
POINT #16	DIGITAL	>0717<	>ON<	>OFF<
POINT #17	ANALOG	>0140<	>ALARM<	>RESET<
POINT #18	ANALOG	>0141<	>ALARM<	>RESET<
POINT #19	ANALOG	>0142<	>ALARM<	>RESET<
POINT #20	ANALOG	>0143<	>ALARM<	>RESET<

SEM #20	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>0720<	>ON<	>OFF<
POINT # 2	DIGITAL	>0721<	>ON<	>OFF<
POINT # 3	DIGITAL	>0722<	>ON<	>OFF<
POINT # 4	DIGITAL	>0723<	>ON<	>OFF<
POINT # 5	DIGITAL	>0724<	>ON<	>OFF<
POINT # 6	DIGITAL	>0725<	>ON<	>OFF<
POINT # 7	DIGITAL	>0726<	>ON<	>OFF<
POINT # 8	DIGITAL	>0727<	>ON<	>OFF<
POINT # 9	DIGITAL	>0730<	>ON<	>OFF<
POINT #10	DIGITAL	>0731<	>ON<	>OFF<
POINT #11	DIGITAL	>0732<	>ON<	>OFF<
POINT #12	DIGITAL	>0733<	>ON<	>OFF<
POINT #13	DIGITAL	>0734<	>ON<	>OFF<
POINT #14	DIGITAL	>0735<	>ON<	>OFF<
POINT #15	DIGITAL	>0736<	>ON<	>OFF<
POINT #16	DIGITAL	>0737<	>ON<	>OFF<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM #21	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	><	><	><
POINT # 2	DIGITAL	><	><	><
POINT # 3	DIGITAL	><	><	><
POINT # 4	DIGITAL	><	><	><
POINT # 5	DIGITAL	><	><	><
POINT # 6	DIGITAL	><	><	><
POINT # 7	DIGITAL	><	><	><
POINT # 8	DIGITAL	><	><	><
POINT # 9	DIGITAL	><	><	><
POINT #10	DIGITAL	><	><	><
POINT #11	DIGITAL	><	><	><
POINT #12	DIGITAL	><	><	><
POINT #13	DIGITAL	><	><	><
POINT #14	DIGITAL	><	><	><
POINT #15	DIGITAL	><	><	><
POINT #16	DIGITAL	><	><	><
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

3.4B THE SYSTEM D FULL MAP

SEM # 3	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>tion 1<	>M1(ON)<	>M1(OFF)<
POINT # 2	DIGITAL	>tion 1<	>M2(ON)<	>M2(OFF)<
POINT # 3	DIGITAL	>tion 1<	>M3(ON)<	>M3(OFF)<
POINT # 4	DIGITAL	>tion 1<	>M4(ON)<	>M4(OFF)<
POINT # 5	DIGITAL	>tion 1<	>M5(ON)<	>M5(OFF)<
POINT # 6	DIGITAL	>tion 1<	>M6(ON)<	>M6(OFF)<
POINT # 7	DIGITAL	>tion 1<	>M7(ON)<	>M7(OFF)<
POINT # 8	DIGITAL	>tion 1<	>M8(ON)<	>M8(OFF)<
POINT # 9	DIGITAL	>tion 2<	>M1(ON)<	>M1(OFF)<
POINT #10	DIGITAL	>tion 2<	>M2(ON)<	>M2(OFF)<
POINT #11	DIGITAL	>tion 2<	>M3(ON)<	>M3(OFF)<
POINT #12	DIGITAL	>tion 2<	>M4(ON)<	>M4(OFF)<
POINT #13	DIGITAL	>tion 2<	>M5(ON)<	>M5(OFF)<
POINT #14	DIGITAL	>tion 2<	>M6(ON)<	>M6(OFF)<
POINT #15	DIGITAL	>tion 2<	>M7(ON)<	>M7(OFF)<
POINT #16	DIGITAL	>tion 2<	>M8(ON)<	>M8(OFF)<
POINT #17	ANALOG	>tion 1<	>HI Set<	>Within<
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM # 4	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>tion 3<	>M1(ON)<	>M1(OFF)<
POINT # 2	DIGITAL	>tion 3<	>M2(ON)<	>M2(OFF)<
POINT # 3	DIGITAL	>tion 3<	>M3(ON)<	>M3(OFF)<
POINT # 4	DIGITAL	>tion 3<	>M4(ON)<	>M4(OFF)<
POINT # 5	DIGITAL	>tion 3<	>M5(ON)<	>M5(OFF)<
POINT # 6	DIGITAL	>tion 3<	>M6(ON)<	>M6(OFF)<
POINT # 7	DIGITAL	>tion 3<	>M7(ON)<	>M7(OFF)<
POINT # 8	DIGITAL	>tion 3<	>M8(ON)<	>M8(OFF)<
POINT # 9	DIGITAL	>tion 4<	>M1(ON)<	>M1(OFF)<
POINT #10	DIGITAL	>tion 4<	>M2(ON)<	>M2(OFF)<
POINT #11	DIGITAL	>tion 4<	>M3(ON)<	>M3(OFF)<
POINT #12	DIGITAL	>tion 4<	>M4(ON)<	>M4(OFF)<
POINT #13	DIGITAL	>tion 4<	>M5(ON)<	>M5(OFF)<
POINT #14	DIGITAL	>tion 4<	>M6(ON)<	>M6(OFF)<
POINT #15	DIGITAL	>tion 4<	>M7(ON)<	>M7(OFF)<
POINT #16	DIGITAL	>tion 4<	>M8(ON)<	>M8(OFF)<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM # 5	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>tion 6<	>M1(ON)<	>M1(OFF)<
POINT # 2	DIGITAL	>tion 6<	>M2(ON)<	>M2(OFF)<
POINT # 3	DIGITAL	>tion 6<	>M3(ON)<	>M3(OFF)<
POINT # 4	DIGITAL	>tion 6<	>M4(ON)<	>M4(OFF)<
POINT # 5	DIGITAL	>tion 6<	>M5(ON)<	>M5(OFF)<
POINT # 6	DIGITAL	>tion 6<	>M6(ON)<	>M6(OFF)<
POINT # 7	DIGITAL	>tion 6<	>M7(ON)<	>M7(OFF)<
POINT # 8	DIGITAL	>tion 6<	>M8(ON)<	>M8(OFF)<
POINT # 9	DIGITAL	>tion 5<	>M1(ON)<	>M1(OFF)<
POINT #10	DIGITAL	>tion 5<	>M2(ON)<	>M2(OFF)<
POINT #11	DIGITAL	>tion 5<	>M3(ON)<	>M3(OFF)<
POINT #12	DIGITAL	>tion 5<	>M4(ON)<	>M4(OFF)<
POINT #13	DIGITAL	>tion 5<	>M5(ON)<	>M5(OFF)<
POINT #14	DIGITAL	>tion 5<	>M6(ON)<	>M6(OFF)<
POINT #15	DIGITAL	>tion 5<	>M7(ON)<	>M7(OFF)<
POINT #16	DIGITAL	>tion 5<	>M8(ON)<	>M8(OFF)<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

SEM # 6	TYPE	POINT NAME	ALARM NAME	RESET NAME
POINT # 1	DIGITAL	>tion 8<	>M1(ON)<	>M1(OFF)<
POINT # 2	DIGITAL	>tion 8<	>M2(ON)<	>M2(OFF)<
POINT # 3	DIGITAL	>tion 8<	>M3(ON)<	>M3(OFF)<
POINT # 4	DIGITAL	>tion 8<	>M4(ON)<	>M4(OFF)<
POINT # 5	DIGITAL	>tion 8<	>M5(ON)<	>M5(OFF)<
POINT # 6	DIGITAL	>tion 8<	>M6(ON)<	>M6(OFF)<
POINT # 7	DIGITAL	>tion 8<	>M7(ON)<	>M7(OFF)<
POINT # 8	DIGITAL	>tion 8<	>M8(ON)<	>M8(OFF)<
POINT # 9	DIGITAL	>tion 7<	>M1(ON)<	>M1(OFF)<
POINT #10	DIGITAL	>tion 7<	>M2(ON)<	>M2(OFF)<
POINT #11	DIGITAL	>tion 7<	>M3(ON)<	>M3(OFF)<
POINT #12	DIGITAL	>tion 7<	>M4(ON)<	>M4(OFF)<
POINT #13	DIGITAL	>tion 7<	>M5(ON)<	>M5(OFF)<
POINT #14	DIGITAL	>tion 7<	>M6(ON)<	>M6(OFF)<
POINT #15	DIGITAL	>tion 7<	>M7(ON)<	>M7(OFF)<
POINT #16	DIGITAL	>tion 7<	>M8(ON)<	>M8(OFF)<
POINT #17	ANALOG	><	><	><
POINT #18	ANALOG	><	><	><
POINT #19	ANALOG	><	><	><
POINT #20	ANALOG	><	><	><

3.4C WHAT IS MEANT BY "Press <CTRL-Z>"?

The symbol <CTRL-Z> is commonly used in all six DASP programs, but not all people know what it means. The <CTRL-Z> is really a single ASCII character just like the letter A or the number 1, but, it has a special meaning to both the RSX-11M operating system and the DASP FORTRAN programs. <CTRL-Z> means the End Of File (EOF) or, in other words, the end of input from the terminal.

The expression "Press <CTRL-Z>" means the following: press and hold down the "CTRL" key, press and release the "Z" key, then release the "CTRL" key. The characters "^Z" should appear on the screen

The "CTRL" key is used in the same fashion as the "SHIFT" key. When using the "SHIFT" key the user holds it down, presses and releases one of the other keys, then releases the "SHIFT" key.

3.4D THE SEM POINT NUMBER ASSOCIATION TABLE

Digital Outputs

Point No.	Type	Relay No.	DB-37 Pin No.	Wire Color
Point #1	Digital	RLY01	31	WHITE BLACK GRAY
Point #2	Digital	RLY02	7	BROWN
Point #3	Digital	RLY03	20	WHITE BLUE
Point #4	Digital	RLY04	28	WHITE BLACK BLUE
Point #5	Digital	RLY05	18	WHITE GREEN
Point #6	Digital	RLY06	4	GREEN
Point #7	Digital	RLY07	14	RED YELLOW
Point #8	Digital	RLY08	11	PINK
Point #9	Digital	RLY09	24	WHITE VIOLET
Point #10	Digital	RLY010	8	YELLOW
Point #11	Digital	RLY011	25	WHITE BLACK RED
Point #12	Digital	RLY012	3	WHITE
Point #13	Digital	RLY013	15	RED BLACK
Point #14	Digital	RLY014	19	WHITE YELLOW
Point #15	Digital	RLY015	29	WHITE BLACK BROWN
Point #16	Digital	RLY016	21	WHITE BROWN

Analog Outputs

Point No.	Type	Channel No.
Point #17	Analog	Channel 1
Point #18	Analog	Channel 2
Point #19	Analog	Channel 3
Point #20	Analog	Channel 4

3.5 Test Examples

3.5.1 System A

CONFIGURATION: Analog Points = 4
Digital Points = 0

EASE OF USE: Relatively simple in hookup & configuring the system for our tests because the small number of transponder points provided. Testing of transponder introduced few problems.

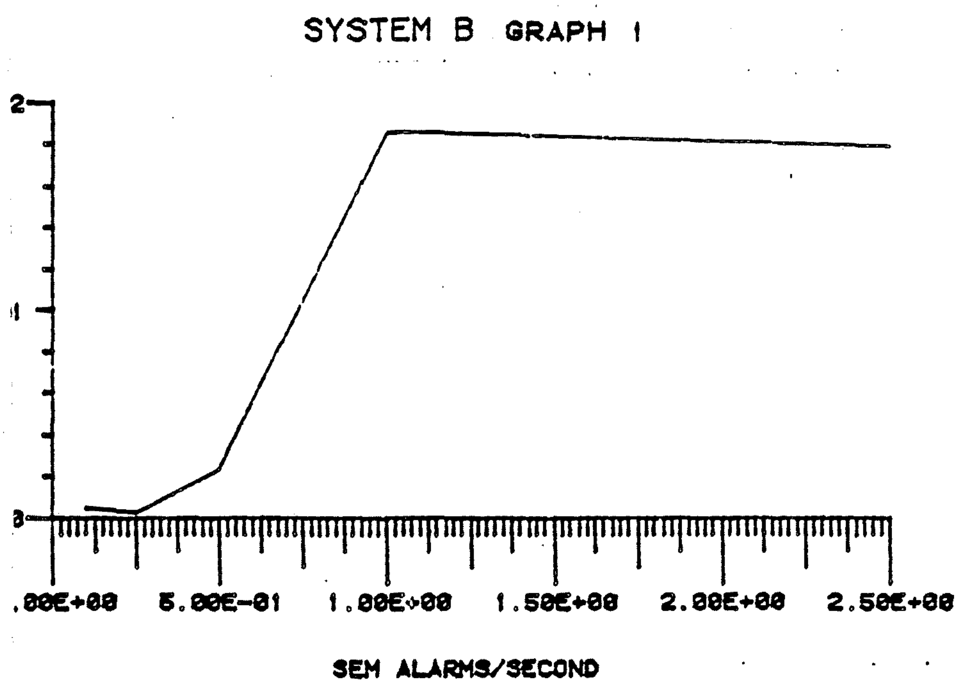
OPERATOR IMPRESSIONS: System performed well with no apparent problems. A larger system, however, would prove much more difficult to hookup and enter point data.

3.5.2 System B

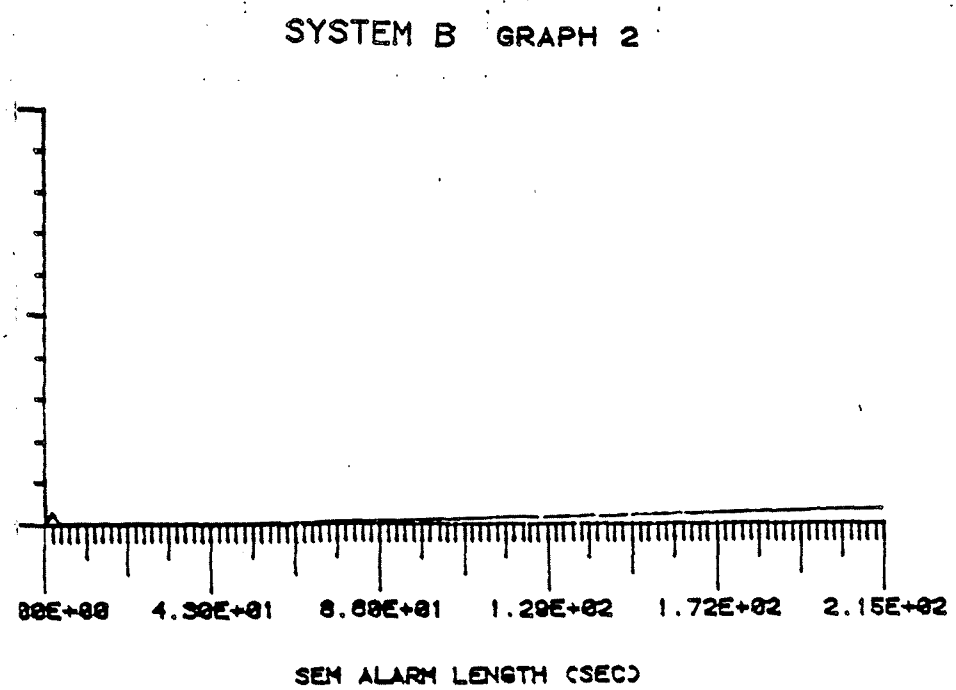
Figure 3.5.2a (Graph 1) shows the System B mine monitoring system as having a maximum alarm frequency of approximately 0.25Hz. Figure 3.5.2b (Graph 2) shows the system as having a scan time of about 1 second. Figure 3.5.2c (Graph 3) does not contain enough data points to specifically determine a harmonic frequency, but shows that one does exist. Therefore, the system uses a polling technique to scan its points. Figure 3.5.2d (Graph 4) shows that the system responds at a very quick and consistant rate of 220 milliseconds per alarm report.

Thirty-six analog points and 320 digital points were tested simultaneously on System B. This system appeared to prioritize the points, giving the digital points a higher priority than the analog points. When using pulse tests with pulse widths of 1 second or greater, the system generally printed all digital alarm reports first, then the analog alarm reports, followed by the digital and analog reset reports. For pulse widths between 420 milliseconds and 840 milliseconds, the system printed all of its digital alarm reports and most of its digital reset reports before printing its analog alarm and reset reports. For pulse widths between 16 milliseconds and 210 milliseconds, it printed most digital alarm and reset reports, but did not print any analog alarm or reset reports. (These results were seen by usng the PRINTTEST and PRINTSTAT programs).

System B was the first true system tested with the Mine Monitoring Test Fixture. A timing problem between the 21 SEM's was later found, which made the data accumulated during the tests invalid. A second problem in setting up the report time stamping program was found that caused the first report printed by System B to be ignored by the program in every test performed. In the author's opinion, however, the data is still fairly representative of the SUT.



5.2a Percentage of Missed Alarms vs Square Wave Frequency



3.5.2b Percentage of Missed Alarms vs Pulse Width

System B was not a full system such as normally sold; it was a special version that lacked a color graphics terminal.

CONFIGURATION:	Analog Points = 80
	Digital Points = 320
CONNECTED:	Analog Points = 36
	Digital Points = 320
EASE OF USE:	Hookup and operation of system was relatively simple;it was a turnkey system. Wiring points was time consuming but rather straightforward since racks were mostly modular mounted, which simplified wiring considerably.
OPERATOR IMPRESSION:	System was not a full-fledged system. Basic system responded well during test; no major problems encountered. When any hookup problems occurred, their representatives were helpful.

3.5.3 System C

CONFIGURATION:	Analog Points = 40
	Digital Points = 64
EASE OF USE:	Transponder hookup was time consuming but rather straightforward. Setup of points did not take too long. Locating bad transponders was a problem. Many transponders failed: 5 analog point; 19 digital points.

3.5.4 System D

The main problem with testing this system was interfacing its alarm output to the test fixture. The main alarm display was an internal display unit driven directly by the system processor. Therefore, it was necessary to use the parallal printer interface, through a parallel to serial converter rigged just for this test. The result was satisfactory but not ideal, as the interface introduced some errors into the recorded responses. Considerable time was spent "cleaning up" the response files.

Some operating procedure problems were identified, which the manufacturer intends to correct. Primarily, the monitoring software executes in a polled rather than event driven (i.e., interrupts) mode, causing monitoring to be suspended while operator commands were issued and when he printer was not ready. The result was a system that "hung" when operator input was incomplete.

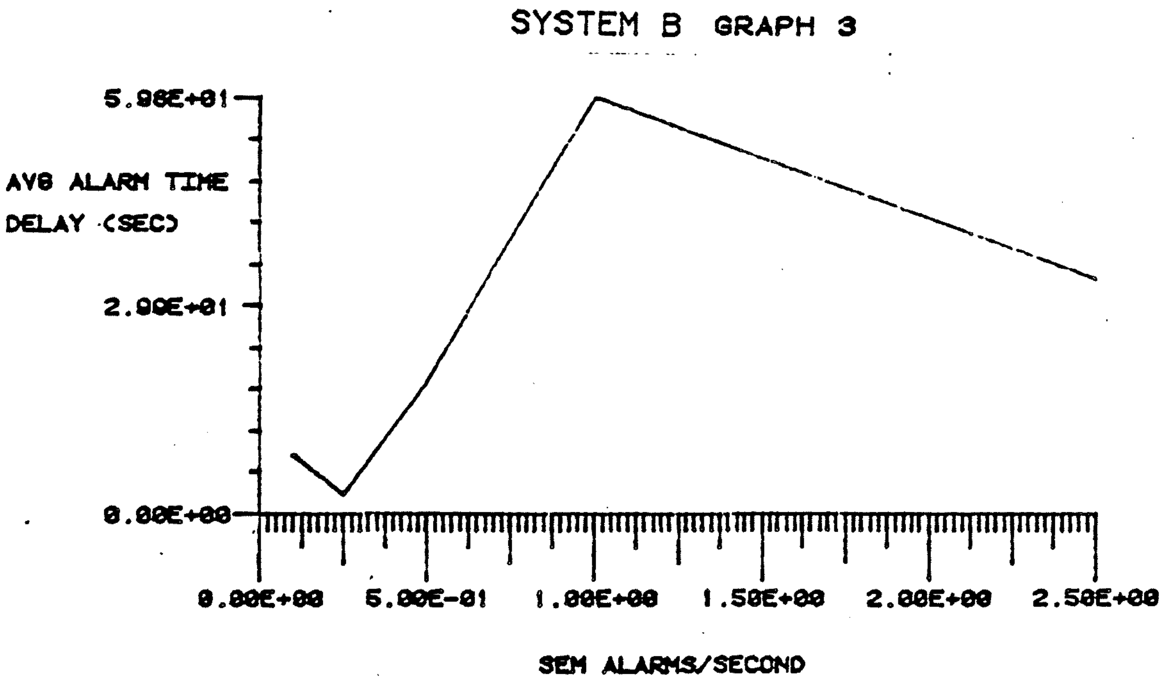


Fig. 3.5.2c Average Alarm Time Delay vs Square Wave Frequency

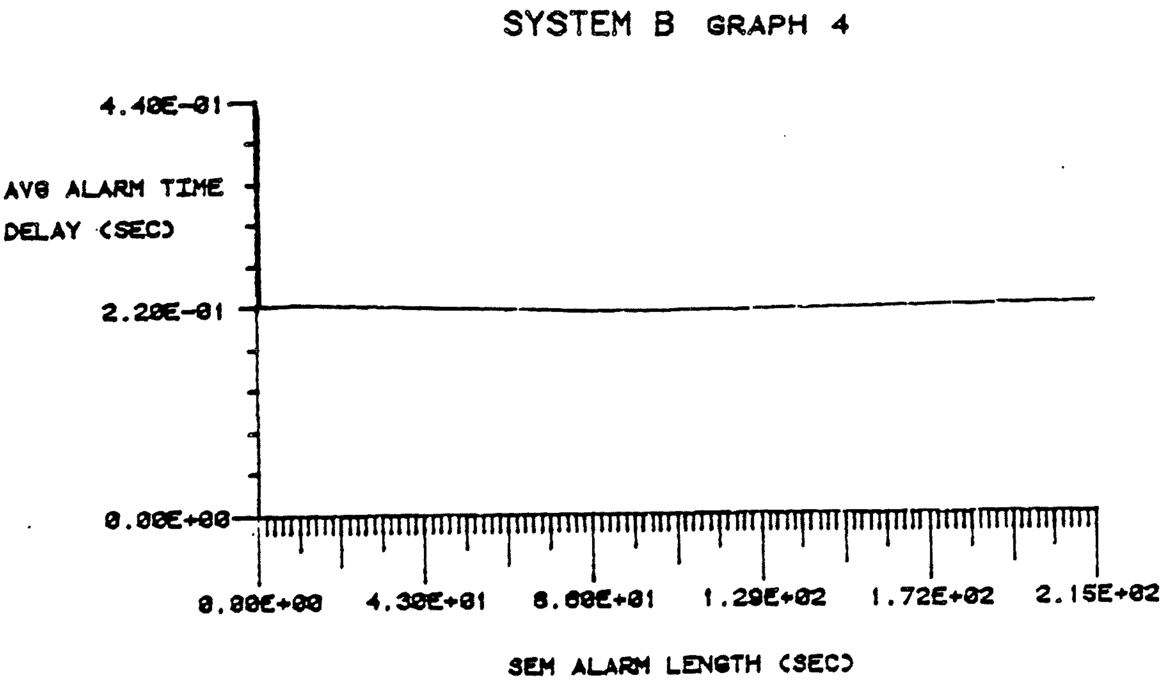


Fig. 3.5.2d Average Alarm Time Delay vs Pulse Width

Setting up the system for test was fairly simple, except for the printer interface. System softwre was menu driven with on-line help, which made it easy to set up once the operator became familiar with the command syntax. The documentation was limited to manuals describing the different components, but since the manufacturer installed the system for the test, this was not a problem. However, it is not possible for WVU to draw conclusions about the true ease of installation since this help was provided.

CONFIGURATION:	Analog Points = 1
	Digital Points = 64
EASE OF USE:	Representatives for System D hooked it up.
OPERATOR IMPRESSION:	Considerable time spent "cleaning up"; some operating procedure problems, causing system to "hang."

REFERENCES

1. R.S. Nutter, N.S. Smith, et al., Methodology Development for Mine Life Safety System Evaluation, USBM Contract No. J0100057, Final Report, June 6, 1982, West Virginia University (contact Ed Fries, USBM, for further information).

2. D.F. Tuthill, "A Data Link Protocol for the WVU Mine Monitoring System," MSEE Thesis, West Virginia University, January 1980.

3. RSX11-M Version 4.0 Volume 1, Digital Equipment Corporation, Maynard, Massachusetts, 1982

4. National Fire Protection Association, National Fire Codes 1981, Volume 7, Proprietary Protective Signaling Systems 72D, Chapter 4, Section 4-5.3.

5. Chia-Der Hsiao, "Microprocessor-Based Sensor Emulator," MSEE Thesis, West Virginia University, December 1982.

6. R.S. Nutter, W.R. Voltz, and M.D. Aldridge, Mine Wide Test of the WVU Monitoring Concept - Phase II Final Report: Modular Mine Monitoring and Control System, USBM Contract No. H0144114, February 1980, NTIS PB 80-227648.

7. K. Jensen and N. Wirth, Pascal User's Manual and Report, Springer-Verlag, New York, New York, 1979.

8. RSX11-M Version 4.0, Volume 4, Digital Equipment Corporation, Maynard, Massachusetts, 1982.

9. Whitesmiths Ltd, Portable 'C' Compiler; Installation Guide, C Interface Manual, C Interface Manual for PDP-11, C Programmers Manual, Pascal Programmer's Manual,

10. R.S. Chapman, "Data Analysis Software Package for the WVU-USBM Mine Monitoring Test Facility," MSEE Thesis, West Virginia University, August 1984.