ornl

ORNL/TM-13041

OAK RIDGE
NATIONAL
LABORATORY

MARTIN MARIETTA

The Automated Bicron Tester: Automated Electronic Instrument Diagnostic, Testing, and Alignment System with Records Generation

G. S. Rao
S. R. Maddox
G. W. Turner
R. I. Vandermolen

INSTRUMENTATION
I & C
CONTROLS DIVISION

MANAGED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL/TM-13041

Instrumentation and Controls Division

The Automated Bicron Tester

Automated Electronic Instrument Diagnostic, Testing, and Alignment System with Records Generation

G. S. Rao
S. R. Maddox
G. W. Turner
R. I. Vandermolen

Date Published — November 1995

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6005
managed by
LOCKHEED MARTIN ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-84OR21400

Contents

List of Figures

List of Tables

Abstract

The Bicron Surveyor MX is a portable radiation monitoring instrument used by the Office of Radiation Protection at Oak Ridge National Laboratory. This instrument must be calibrated in order to assure reliable operation. A manual calibration procedure was developed, but it was time consuming and repetitive. Therefore, an automated tester station that would allow the technicians to calibrate the instruments faster and more reliably was developed. With the automated tester station, calibration records and accountability could be generated and maintained automatically. This allows the technicians to concentrate on repairing defective units.

The Automated Bicron Tester consists of an operator interface, an analog board, and a digital controller board. The panel is the user interface that allows the technician to communicate with the tester. The analog board has an analog-to-digital converter (ADC) that converts the signals from the instrument into digital data that the tester can manipulate. The digital controller board contains the circuitry to perform the test and to communicate the results to the host personal computer (PC). The tester station is connected to the unit under test through a special test harness that attaches to a header on the Bicron. The tester sends pulse trains to the Bicron and measures the resulting meter output. This is done to determine if the unit is functioning properly.

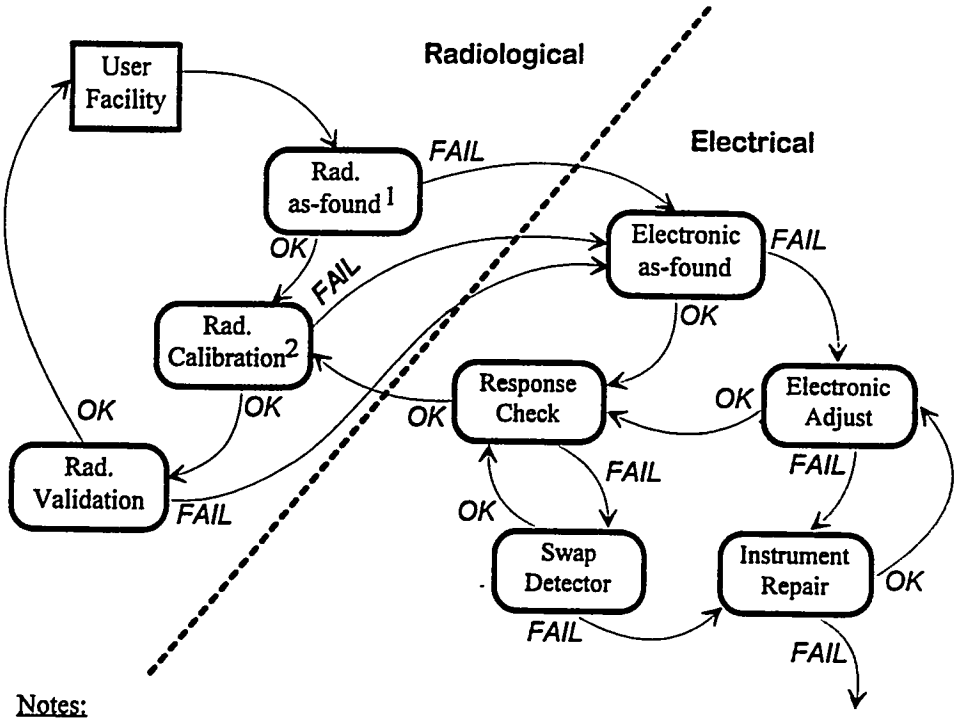
The testers are connected to the host PC through an RS-485 serial line. The host PC polls all the tester stations that are connected to it and collects data from those that have completed a calibration. It logs these data and stores the record in a format ready for export to the Maintenance, Accountability, Jobs, and Inventory Control (MAJIC) database. It also prints a report.

The programs for the Automated Bicron Tester and the host are written in the C language. On the host PC, the programs control the polling of the tester stations, data logging, and report generation. The firmware code on the tester executes the alignment and calibration procedure and communicates these data to the host.

1. Introduction

The Bicron Surveyor MX is a portable count rate instrument manufactured by the Bicron Corporation. It is used by the Office of Radiation Protection at Oak Ridge National Laboratory (ORNL) to monitor radiation levels. Different probes can be used with these devices to detect different types of radiation. The count rate is displayed on a meter on the instrument.

These instruments have to be calibrated quarterly to ensure reliable operation. The units undergo radiological and electronic calibrations as needed. A diagram of the Radiation and Standards Calibration Laboratory (RASCAL) procedure flow is given in Fig. 1.1, and another flowchart is given in HPIC-SOP-1. The manual electronic calibration procedure involved several steps and was repetitive, time consuming, and operator intensive. To automate the procedure and allow the instrument technicians to concentrate on the diagnosis and repair of defective units, the Automated Bicron Tester (ABT) was developed.



Notes:
1. Includes battery change.
2. May be a no-operation for some instrument types.

Fig. 1.1 RASCAL procedure flow.

The ABT is designed to

- minimize the time required to perform the calibration and thereby increase throughput,
- make the test and alignment procedure uniform,
- enhance the reliability of the procedure, and
- keep records of the alignment and maintain accountability.

The ABT has a bar code reader to collect the identification (ID) numbers of the technician, automated tester, and Bicron instrument. The tester works by sending probe signals (pulse trains) to the Bicron and measuring the meter output. It connects to the instrument through a special harness that is attached to the header pins that are visible when the Bicron is opened. The calibration procedure is stored in memory and is executed by a microcontroller. Throughout the procedure, the tester prompts the technician for information and verification on the progress of the calibration. The tester builds a record of the calibration and, upon completion, sends the information to the host PC. Several testers can be connected to one host through the RS-485 serial communication protocol. This allows several calibrations to be performed simultaneously with only one central database.

The host PC polls the tester stations and retrieves data packets whenever a station has completed a calibration. The host then creates a test results data file, stores the calibration data, and creates a log file. The records in the log file represent work orders and are in a format that can be sent to the MAJIC database. The host also prints a hard copy of the report and work order.

Detailed drawings of the ABT and its connections are available in the Q6350 set of illustrations. These illustrations are given in Sect. 6.6 and are referenced by number in the text. A basic outline of the operation of the ABT is given below. The rest of this report describes the construction of the tester, its software, and the calibration procedures.

Step 1: Be sure that the tester stations have been properly aligned and calibrated before operation.

Step 2: Make sure that the tester stations and the host PC equipment are properly connected.

Step 3: Run the MONITOR program on the host PC, which will continuously check the status of the tester stations, upload the completed data sets, and print the reports.

Step 4: Power up the tester station(s) and verify that the MONITOR program is communicating with them by viewing the host PC display.

Step 5: Scan your badge number to log in at the tester station and follow the on-screen instructions to perform the test procedure for an instrument.

Step 6: Confirm that the MONITOR program has received and stored the test results data set and printed the results report before proceeding with the next test.

2. Hardware Description

2.1 THEORY OF OPERATION

The ABT connects to the Bicron Surveyor MX Health Physics Instrument (HPI) and allows the operator to perform an electronic calibration of its circuits. The tester mimics the Bicron probe and sends a test signal to the Bicron HPI. The signal is a pulse train of a certain frequency that is proportional to the count rate. The Bicron HPI integrates this pulse train and sends the output voltage to its display meter. The tester measures this output voltage and uses it to calibrate the electronics of the instrument. The Bicron had to be modified to allow the tester to control it during the test. This modification is now done at the factory. The details of the modification are given in Sect. 2.7.

The tester implements the procedure for calibrating the Bicron Surveyor MX, and since the procedure is automated, there is little room for deviation. Skipped steps in the calibration are documented, and the variations in results between operators are minimized. The signal generated by the tester is deterministic, unlike natural radiation, which is random. The tester also saves all the pertinent information about the calibration in a packet that is later stored in a database on the host PC. Procedural compliance and archival records of the calibration are retained.

The ABT consists of three parts: the operator interface, the analog board, and the digital controller board. There is also a test harness that connects the tester to the Bicron HPI. An overall functional block diagram of the tester is given in drawing Q6350-16 in Sect. 6.6. The interconnection of the boards within the tester is shown in Q6350-12. These drawings provide an overview of the test station's construction and its connections to external devices.

2.2 OPERATOR INTERFACE

The operator interface is mounted on the front panel of the automated tester. A functional block diagram is given in drawing Q6350-15 in Sect. 6.6. The circuit diagram is part of the digital board's schematic drawing (Q6350-1). The front panel consists of a set of four programmable switches and an AND1013ST graphics display module (DSPY201). The module contains a controller board with a T6963T liquid crystal display (LCD) controller, which drives an LCD panel. The LCD panel displays the test being performed, information about the Bicron, and instructions for proceeding with the calibration. The LCD display and controller are connected to the digital board through connector P9.

The four push-button switches (S201–S204) are connected to the digital board through connector P10. The switch debounce circuits and buffers are on the digital board. The operator uses the switches to enter data and confirm proper operation during the calibration. The switches are generic and menu driven, which allows them to have different functions for different test sets.

2.3 ANALOG BOARD

The analog board serves as the link between the Bicron HPI and the digital controller board. It has an ADC to convert analog voltages from the HPI into digital data that can be processed by the microcontroller on the digital board. It also has a section that performs the antisaturation (antisat) test for probes with G-M

tubes. A functional block diagram of the board (Q6350-14) and the schematic (Q6350-2) are given in Sect. 6.6.

The analog board takes voltages from the Bicron via connector J1 and, using relays (U3–U5, U7, U8) to select the signal, routes it through buffers (U2, U10) and a low-pass filter (C13, C14). The buffers are used to isolate the Bicron from the automated tester and to drive the analog signal. The buffers are LF355 operational amplifiers. The filtered signal is passed to a sample-and-hold circuit (U6) that stores the analog signal value. The ADC (U1) then converts this analog voltage to a 12-bit digital number and passes it to the digital board via connector J5.

There is also a +5-V reference source (U9) on the board that is used in the self-test procedure. The reference is used to test the operation of the ADCs. Instead of the Bicron signal, the reference is routed to the ADC and read by the microcontroller. The voltage read by the ADCs should be within 10 mV of 5 V.

The pulse injection circuits on the digital board are connected to the analog board via connector J6. The signal can be looped back for verification or routed to the Bicron through connector J3 and the test harness. The relays control the routing of the signals. The control signal for the high-voltage board also comes through J6.

The high-voltage board is used in the antisat test. This test is performed only on instruments with G-M tubes. A Magnecraft high-voltage single-pole single-throw (SPST) reed relay (K101) driven by an npn transistor (Q1) is used to select between two resistances: 1000 and 2000 MΩ. The 1000-MΩ resistance is created by connecting two 2000-MΩ resistors in parallel. The tester checks the meter reading with these two loads to verify the operation of the antisaturation circuits.

2.4 DIGITAL CONTROLLER BOARD

The digital controller board carries out the computations required to perform the calibration and stores the information. This information is later transmitted to a computer, where a report is generated and an entry is created in a database. The tester keeps track of the numbers of the automated tester, the unit under test, and the operator. It also stores the as-found status of the unit and the as-left status after the calibration. Sect. 6.6 contains the functional block diagram (Q6350-13) and the schematic (Q6350-1) of the digital controller board.

The system has three main buses: the low half of the data bus (bits 0–7), the high half of the data bus (bits 8–15), and the control lines read, write, and address latch enable (RD, WRT, and ALE). These are used to control the various elements on the board and to route data between them. There are also several latches that are used to hold data for various components in the system.

The 80C321 microcontroller (U25) executes the calibration program that is stored in the erasable programmable read-only memory (EPROM) (U19) and builds the information packet in the static random access memory (SRAM) (U18). The microcontroller is composed of an 8051 central processing unit (CPU) with RAM, timers, and other built-in features.

The tester communicates with the host PC by sending data through the universal asynchronous receiver transmitter (UART) (U23) and then through an RS-485 interface (U10). The RS-485 standard is a slight variation of the RS-422 protocol that allows multidrop operation from one port. An independent UART (U22) and RS-232 interface (U12) are also available for communication. They are unused and free for expansion. Each tester has a unique address that is set on the digital board using dual inline package (DIP) switches (SW1).

The bar codes of the tester, unit under test, and operator's badge are read with a bar code reader. The bar code reader circuitry (U1–U3) handles the interface between the bar code reader and the microcontroller.

The HP1810 (U1) sends the data directly to the local serial channel (LSC) on the microcontroller and beeps on completion. The beeps are generated by the Mallory SNP2 annunciator (LS201) connected to J3.

The operator communicates with the tester by using the front panel. The front panel LCD receives its data from the digital board via connector J9. The display contrast is also set on the board (U13 and potentiometer R4). Input from the switches comes through J10 and is sent to debounce (U14) and buffer circuitry (U4A–U4D) before being routed to the interrupt controller.

The interrupt controller (U24) sends an interrupt to the microcontroller if activity occurs on its inputs. The ADC status line, front panel switches, bar code reader, and UART interrupts are connected to the interrupt controller.

The Bicron is controlled by the digital board while it is connected to the tester. The scale on the unit under test is selected with circuitry on the board (U7, U15A, Q1). The control signals are sent through J6 to the test harness. The pulse generator circuits (U5, U6C, U9, U11, U26E–F) are used to create a pulse train that is sent to the Bicron via J5 and the analog board. These pulses have specific, selected frequencies and are similar to those generated by the probes. The digital board controls the ADC and selects the Bicron instrument modes and the signals to test (U26A, U27A–B, U28A–F). The meter output of the Bicron is measured through the analog board. The ADC control lines go to the analog board via J4 and J5. Data are returned from the analog board via J4. The peripheral interface adapter (U21) multiplexes data among the ADC, the pulse generator circuits, and the data bus.

The device selector (U16) makes sure that there is no bus contention by selectively enabling different sections of the digital board. The output of the device selector is connected to the SRAM, peripheral interface adapter, RS-422 UART, LCD display, interrupt controller, scale selector, and RS-232 UART. The microcontroller selectively enables each device as needed.

2.5 TEST HARNESS

A special test harness had to be built to connect the automated tester to the Bicron HPI. Diagrams are given in Q6350-6, Q6350-7, and Q6350-10 in Sect. 6.6. The end of the harness connected to the tester has a 31-pin DD31S connector. Twenty-five lines of the cable are used to make the connector. The other end of the cable is attached to a custom-built adapter. The adapter has 25 spring-loaded pin contacts and a locking device. The adapter slides onto the circuit board of the Bicron HPI and locks into place. The pins are offset and spaced to make contact with the pins of the Bicron HPI header. This header connects the two printed circuit boards of the HPI and is exposed when the instrument's cover is removed. This establishes the electrical connection between the tester and the HPI. The harness is detached by disengaging the locking mechanism and sliding the adapter off.

2.6 PARTS LIST

The following subsections contain a list of the parts used to build the automated tester. The component label is included in parentheses. In addition to these parts, every integrated circuit on the digital board has a 0.1-μF bypass capacitor. Refer to the Q6350-series schematics for details on the circuitry and connections.

2.6.1 Operator Interface

AND1013ST: (DSPY201) Wm. J. Purdy Co. LCD panel with integrated T6963T controller . (Qty. 1)
Push-button: (S201–S204) push-button switch ..... (Qty. 4)
T6963T: LCD panel controller (part of AND1013ST package) ..... (Qty. 0)

2.6.2 Analog Board

1N4446: (D101) Diode ..... (Qty. 1)
2N3904: (Q1) Low-power high-frequency npn transistor ..... (Qty. 1)
AD574A: (U1) Analog Devices' 12-bit ADC ..... (Qty. 1)
AD584: (U9) Analog Devices' voltage reference ..... (Qty. 1)
DG126: (U3-U5, U7, U8) Intersil analog switch ..... (Qty. 5)
LF355: (U2, U10) Operational amplifier ..... (Qty. 2)
SMP-11: (U6) Analog Devices' sample and hold amplifier ..... (Qty. 1)
W102VX-50: (K101) Magnecraft high-voltage SPST reed relay (Newark #56F1031) ..... (Qty. 1)
Capacitors:
- 5000 pF ..... (Qty. 1)
- 0.001 µF ..... (Qty. 1)
- 0.01 µF ..... (Qty. 3)
- 0.1 µF ..... (Qty. 8)
- 1.0 µF ..... (Qty. 2)
- 10 µF ..... (Qty. 4)
- 4.7 µF ..... (Qty. 3)

Resistors (1% ¼ W):
- 62 Ω ..... (Qty. 1)
- 100 Ω ..... (Qty. 1)
- 1 kΩ ..... (Qty. 1)
- 1.5 kΩ ..... (Qty. 1)
- 2.4 kΩ ..... (Qty. 1)
- 4.99 kΩ ..... (Qty. 1)
- 10-kΩ potentiometers ..... (Qty. 4)
- 20 kΩ ..... (Qty. 2)
- 95.3 KΩ ..... (Qty. 2)
- 100-kΩ potentiometers ..... (Qty. 3)
- 1 MΩ ..... (Qty. 1)
- 2000 MΩ ..... (Qty. 2)

2.6.3 Digital Controller Board

1N4446: (D4) Diode ..... (Qty. 1)
2N3906: (Q1) Low-power high-frequency pnp transistor ..... (Qty. 1)
4049: (U4, U15, U26, U28) Hex inverters ..... (Qty. 4)
74HCT138: (U16) 8-line demultiplexer ..... (Qty. 1)
74HCT139: (U27) 4-line demultiplexer ..... (Qty. 1)

74HCT163: (U11) 4-bit up-counter ..... (Qty. 1)
74HCT174: (U9) Hex D flip-flop ..... (Qty. 1)
74HCT244: (U8) Buffer/bus logic drivers ..... (Qty. 1)
74HCT32: (U6, U20) Quad 2-input OR gates ..... (Qty. 2)
74HCT574: (U2, U7, U17) Octal D flip-flop ..... (Qty. 3)
80C321: (U25) AMD 8-bit microcontroller ..... (Qty. 1)
8259A: (U24) AMD interrupt controller ..... (Qty. 1)
82C52: (U22, U23) UART/serial controller ..... (Qty. 2)
82C55: (U21) Peripheral interface adapter ..... (Qty. 1)
87C257: (U19) EPROM ..... (Qty. 1)
DIP-4: (SW1) 4-line DIP switch ..... (Qty. 1)
DS8922A: (U10) National Semiconductor differential transceiver ..... (Qty. 1)
HM6116: (U3) Hitachi SRAM ..... (Qty. 1)
HP1810: (U1) Hewlett Packard bar code reader interface chip ..... (Qty. 1)
MAX233: (U12) Maxim bus transceiver ..... (Qty. 1)
MAX636: (U13) Maxim switch mode supply circuit ..... (Qty. 1)
MC14490: (U14) Motorola signal conditioner ..... (Qty. 1)
P6KE 7.5C: (D1-D3) Tranzorb bidirectional transient suppressors/zener diodes ..... (Qty. 3)
PXO-1000: (U5) Statek Corporation crystal-controlled oscillator ..... (Qty. 1)
SIP 10K: (RP1) Single inline package of 10-kΩ resistors ..... (Qty. 1)
SNP2: (LS201) Mallory annunciator ..... (Qty. 1)
TC5565: (U18) Toshiba SRAM ..... (Qty. 1)

Capacitors:
- 20 pF ..... (Qty. 4)
- 33 pF ..... (Qty. 4)
- 0.01 µF ..... (Qty. 1)
- 0.1 µF ..... (Qty. 1)
- 10 µF ..... (Qty. 2)
- 10 µF 20 V ..... (Qty. 3)
- 100 µF ..... (Qty. 1)

Crystals:
- 1.8 MHz ..... (Qty. 1)
- 1.8432 MHz ..... (Qty. 1)
- 11 MHz ..... (Qty. 1)
- 12 MHz ..... (Qty. 1)

Inductors:
- 330 µH ..... (Qty. 1)

Resistors (1% ¼ W):
- 1 kΩ ..... (Qty. 2)
- 10 kΩ ..... (Qty. 5)
- 10-kΩ potentiometers ..... (Qty. 1)

2.7 BICRON SURVEYOR MX HARDWARE MODIFICATION

For the testing station to remotely control the functions of the Bicron Surveyor MX, it is necessary to make a simple, one-time circuit modification to the Bicron instrument. This involves cutting a trace on the main circuit board and installing a slide switch to either make or break the connection. The effect is to bypass the "OFF" setting of the main rotary switch and allow remote selection of the other operating modes. This modification is now being done by the manufacturer, and the process is included here for completeness.

The steps of the modification procedure are outlined below. Please refer to the circuit drawings in the back of the Bicron Surveyor MX technical manual for the locations of the various components.

1. Turn the instrument main switch "OFF."
2. Remove the batteries.
3. Unsolder the jumper to the right of SW8.
4. Cut all traces around and between the outermost unused group of three pads at the jumper location (solder side).
5. Insert an SPDT (single-pole double-throw) slide switch (Alco #TSS11DG-1-PC) into that group of holes (component side).
6. Cut the trace running from pin 12 of the 24-pin connector to D16, near the cathode leg of D16 (solder side).
7. Attach a 30-gauge wire from the leg of D16 to the right-hand pin of the new switch (solder side).
8. Attach a 30-gauge wire from the other side of the cut trace to the middle pin of the new switch (solder side).
9. Reinstall the jumper in its original position, but on the solder (noncomponent) side of the board.
10. Replace the batteries.
11. With the new switch in the closed (operate) position, turn the main switch from "OFF" to "BAT" and then back to "OFF."

The meter should respond normally with the new switch in the CLOSED (operate) position but will not return to zero with the new switch in the OPEN (test) position. This check can be performed anytime to verify that the new switch is set properly. For more information on the Bicron Surveyor MX Portable Count Rate Meter, please refer to the technical manual on the instrument (Manual No. 1084901).

3. Test Station Setup

3.1 CONNECTING THE TESTER STATIONS TO THE HOST PC

The tester station has an RS-485 communication port (the terminal strip connector on the back) that must be connected to the RS-422 communication adapter on the host PC in order to transmit test results. The tester port is sometimes called an RS-422 port. RS-485 is the multidrop version of the RS-422 communication protocol. This means that several tester stations can be connected to one PC using a multidrop line.

The RS-422 adapter (Telebyte #262F) is designed to plug directly into the standard RS-232 serial port on the host PC. It requires an external 9-V supply, which is provided with the adapter. The switch labeled "DCE/DTE" should be set to the DCE position.

The RS-422 transmit and receive signals each require a twisted-wire pair (a total of four conductors). The shield for the cable should be grounded at the host end only. Note that the xmt/rcv signals are interchanged between the PC and the tester. The connections are shown in Fig. 3.1.

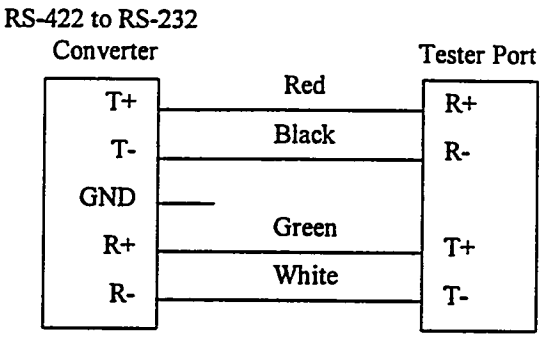


Fig. 3.1 Single RS-422 connection.

Multiple tester stations can be linked to the PC as shown in Fig. 3.2 by "daisy chaining" the connections from one station to the next.

Each station contains an internal address selector switch (SW1 on the digital board) that must be set to provide a unique address (1 through 7) for each station connected. SW1.3 is the most significant bit and SW1.1 is the least significant bit of the address. Address 1 is shown in Fig. 3.3. Table 3.1 lists all the addresses and their corresponding switch positions. The MODE switch (SW1.4) selects the tester station operating mode and should normally be in the OPEN (operate) position except during the tester alignment procedure.

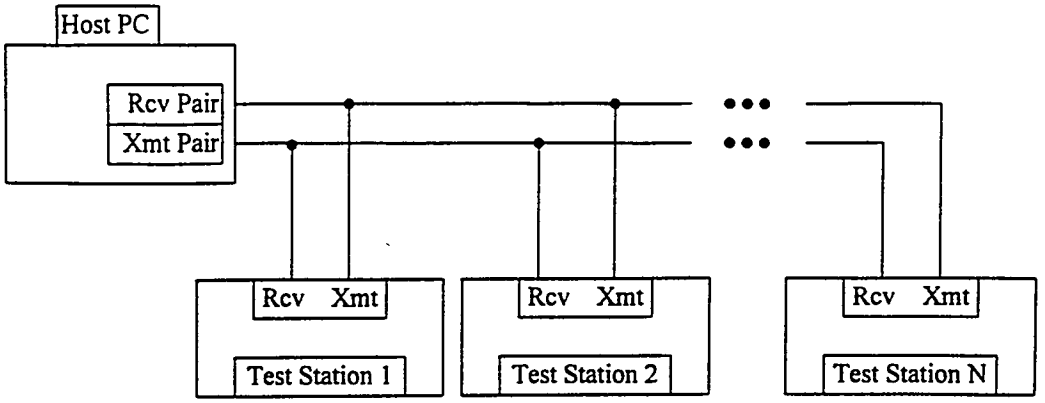


Fig. 3.2 Multidrop RS-485 connection.

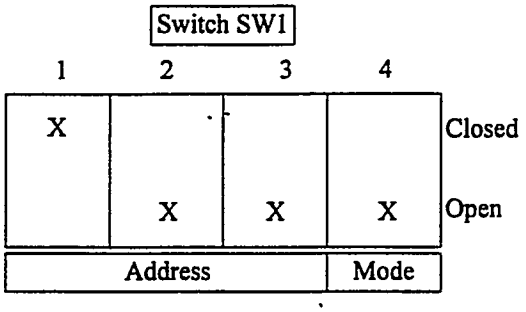


Fig. 3.3 Address selector switch settings.

Table 3.1 Switch selectable device addresses

Address	*Binary Number*	*SW1.1*	*SW1.2*	*SW1.3*
1	001	Closed	Open	Open
2	010	Open	Closed	Open
3	011	Closed	Closed	Open
4	100	Open	Open	Closed
5	101	Closed	Open	Closed
6	110	Open	Closed	Closed
7	111	Closed	Closed	Closed

3.2 CONNECTING THE TESTER STATION COMPONENTS

The tester station includes an ac-powered control and display unit with a large LCD panel and function keys, a test harness that provides the primary connection to the instrument to be tested, and a bar code reader wand. The test harness and the bar code wand should be connected to the tester station via the sockets on its back panel. The test harness has a 31-pin DD31S connector, and the bar code reader has a cylindrical Switchcraft 61HA5F 5-pin connector. The Bicron HPI should be opened, exposing the header pins. The U-shaped end of the test harness slides onto the Bicron circuit board, and the pins of the harness make contact with the Bicron header pins. The harness should lock into place.

A separate electrostatic reference voltmeter is used to obtain accurate measurements of high voltage from the instrument under test. The voltmeter should be connected to the probe connector (MHV coaxial) on the Bicron HPI. For the antisat test, the high-voltage connector of the automated tester should be connected to the HPI's probe connector. The connections are shown in Q6350-11 in Sect. 6.6.

4. Host PC Software Description

The PC programs are written in C and compiled with the Microsoft C compiler. There are four programs that run on the PC: MONITOR.EXE, REPORT.EXE, TESTCOMM.EXE, and TESTRPT.EXE. There are also three batch files. Since the programs use certain table files, you should be in the directory "C:\TESTER" when you run them. This is done by typing "cd \tester" at the DOS command prompt. A description of the required files is given in Sect. 4.4. A brief description of the PC programs is given in the installation instructions below. A more detailed description of these programs follows. Sections 6.2 and 6.3 are a reference with a list of all the definition files and functions in the PC portion of the software.

The programs use functions provided by commercial vendors. The "C Asynch Manager" tools by Blaise Computing, Inc., are used for asynchronous communication with the automated testers. The "Curses/PC" library by Aspen Scientific Corporation provides a window-based user interface for interaction with the user. Consult the manuals or the function reference in Sect. 6.3 for more information.

4.1 INSTALLATION

Installation of the PC software to support the Bicron Surveyor MX automated tester is accomplished by DOS batch commands. The software can be installed from either floppy drive a: or floppy drive b:.

Place the distribution diskette in either drive a: or drive b: . Change your active drive to the floppy drive you placed the distribution diskette into. Type either "a:<enter>" or "b:<enter>."

Next type "install<enter>." This will create directories "C:\TESTER" and "C:\TESTER\OLD" on your c: drive if they do not already exist. Next, the batch command copies any existing files to the backup directory, "C:\TESTER\OLD," and copies all the new files to "C:\TESTER." During the installation, the batch command will pause between installation steps to let you see what is being done. This is normal and you will be asked to "Press any key to continue . . ."

At the end of the installation, there is a message about whether this is the first time you have installed the software. If it is the first time you have installed the PC software, you will need to install tables needed by the tester program to generate reports and databases. To install the tables, type "tinstall<enter>." This copies new copies of the required tables onto your c: drive.

This completes initial software installation and leaves the following files and directories on the C: drive:

• OLD		backup directory
• HELP	BAT	lists OPSGUIDE.TXT to the screen
• TRESTORE	BAT	copies *.TBL from "OLD" directory
• TSAVE	BAT	copies *.TBL to "OLD" directory
• MONITOR	EXE	gathers data from autotesters during calibrations
• LIST	COM	allows scrolling of text on screen
• REPORT	EXE	generates hard-copy reports of calibrations
• TESTCOMM	EXE	tests communications between PC and autotesters
• TESTRPT	EXE	tests for proper tables and report generation
• README	TXT	README file with installation instructions
• OPSGUIDE	TXT	Operations Guide for the Automatic HPI Test System
• CALDATE	TBL	lists calibration dates for instruments by M#
• COMMENTS	TBL	canned comments for calibration and maintenance report

- SERNUM TBL Accumulated M#, service designator, and serial number list
- WORKREQ TBL Accumulated calibration records by work request number

4.2 MONITOR.EXE

The MONITOR program polls all of the tester stations that are connected to the PC's serial port via the RS-422 adapter, displays their current status, uploads test results data sets, and prints test results reports. To start the program, change to the "TESTER" directory and enter the command "MONITOR" from the DOS prompt.

The MONITOR program will then prompt you to enter the serial port number (default: 1) and whether or not you want the test results reports to be printed automatically (default: 'Y'). After that, it continually polls the tester stations and displays the current status of each:

- ----- : Station not responding (off-line)
- IDLE : Waiting for log in
- BUSY : Test procedure in progress
- READY : Test complete; ready to transmit data set
- DONE : Data set transmission complete

When any station indicates that it is "READY," the program requests the transmission of the current test results data set and stores it on the hard disk. If you have enabled the printer output, a formatted test results report will automatically be printed whenever a new data set is received. You should write in any comments or corrections necessary before copying and filing the report.

To exit from the MONITOR program, press the Escape key.

Note: The program should not be interrupted while it is uploading a data set or printing a report since the calibration data may be lost.

4.3 REPORT.EXE

The REPORT program provides a way to generate a printed results report from any data set previously saved by the MONITOR program. This is useful for printing reports at the end of the day, reprinting a report from a previous day, etc.

To start the program, enter the command "REPORT." The program will then display a list of the available subdirectories, and you may then select a directory by using the cursor keys (arrows, Home, End, PgUp, PgDn). Once you have selected a directory, press <enter>, and a list of the data files in that directory will appear. After selecting a data file name in a similar manner, press <enter> to print the formatted report. To get out of either selection list, press <esc>.

4.4 DATA FILES USED BY THE SOFTWARE

Both programs (MONITOR and REPORT) use the following files to look up data items that are not supplied in the test results data set. If these files are missing or incomplete, blanks will appear on the printed reports. A summary of the contents of the table files is given below.

- WORKREQ.TBL : Next work request number and data item definitions.
- COMMENTS.TBL : List of user's comments.
- SERNUM.TBL : Instrument ID, serial number, and service designator.
- CALDATE.TBL : Tester station and voltmeter ID numbers and calibration due dates.

The request number file (WORKREQ.TBL) is created using a text editor and contains the next available work request number (e.g., V50000). This file is updated by the program each time a new data set is uploaded from a tester station. In case this file is lost or damaged, it should be reloaded from a backup, and the work request number must be manually entered. This is to make sure that there is no overlap of work request numbers. The correct work request number is the next unused number and should be obtained from the MAJIC database.

The last two table files (SERNUM.TBL and CALDATE.TBL) should be extracted from the MAJIC database system and copied to the tester system PC at regular intervals in order to keep them up-to-date.

4.5 DATA FILES GENERATED BY THE SOFTWARE

The MONITOR program uses the current date to generate a subdirectory in which to save the test results data set (e.g.: 91-01-15). It uses the ID number of the instrument under test to generate a file name for the data set (e.g.: M012345x.DAT). In the case of multiple test runs for the same instrument on the same day, the last letter of the data set file names will be different (e.g.: A, B, C...).

The MONITOR program also maintains a "log file" for each date that the ABT is used (e.g.: 91-01-15.LOG). Each record in that file contains all of the data representing a single "work request," properly formatted for direct export to the MAJIC database system. These data must be periodically uploaded to the MAJIC database. To do this, follow these steps:

1. Rename the log files to be transferred (yy-mm-dd.LOG) to assume the form "yymmdd.DMI." The letters "yy", "mm," and "dd" represent the year, month, and day respectively. The MAJIC database requires files to conform to this naming convention.
2. Transfer the .DMI files to a user's directory on the ICREMS VAX. This can be done using FTP (file transfer protocol) over the Internet or Kermit using a modem.
3. Log into the VAX account with a terminal emulator. At the "$" prompt, type "HPDATA." This runs the program that will send the files to temporary locations pending final entry into the MAJIC database.
4. Select option #2: "Electronic Data Entry of HP Cal Data," and proceed as directed by the program menus.
5. Enter the name of the first .DMI file to upload. An error file should print.
6. If there are more files to upload, go to step 4 and repeat the process until all the files have been submitted.

7. File the error printouts. Provide the maintainers of the MAJIC database with a list of all the .DMI files that have been submitted. They will perform the final update of the database with the calibration data.

The MONITOR program also prints out a calibration data record and a work order. Both of these should be filed to maintain a record of the calibration.

4.6 TESTCOMM.EXE AND TESTRPT.EXE

The TESTCOMM program tests the communication link between the host PC and its connected automated tester stations. The program prompts for the serial port used for communication and the address range of the testers. It then polls the tester stations and, if any are ready to transmit, uploads and saves a data set.

The TESTRPT program prints a formatted test results report and job ticket. It asks the user for the directory and file name of the data file and whether or not the report is to be sent to the printer. The program loads the data from the file into memory and creates a calibration data sheet and a work request form. These forms are printed or saved to a file.

4.7 BATCH FILES

Three batch file programs also come with the software package: HELP, TRESTORE, and TSAVE. The HELP batch file uses the program LIST.COM to display the system operating guide (OPSGUIDE.TXT) to the screen. LIST.COM is a freeware program that is bundled with the tester PC software. TSAVE copies all of the tables in the current directory (*.TBL) to the directory "C:\TESTER\OLD." TRESTORE copies all of the tables from the directory "C:\TESTER\OLD" into the current directory. These routines are used to make backup copies of the tables and to restore them as needed.

5. Automated Tester Firmware Description

The firmware for the ABT is also written in C. The code runs on the 8051 microcontroller (80C321) on the digital board. The code was written on a PC and compiled using the Archimedes C-51 cross-compiler. It was tested using Nohau emulator hardware. The Nohau emulator plugs into the target circuit in place of the actual 8051 microprocessor and executes the code that has been loaded into the emulator memory. It allows single-step execution, breakpoints, register examination, etc. After testing, the firmware was relinked and converted to Intel Hex format and copied onto a 32K×8 EPROM (87C257) using a programmer. The microcontroller executes the code from the EPROM.

The main program is BicrTest.c51. This firmware controls the operation of the tester and contains both the Bicron test procedure and ABT alignment procedure. It controls the hardware of the tester and instructs the user to calibrate the instrument, stores the data, and communicates this information to the host PC.

5.1 TESTER STATION OPERATION

The Bicron test procedure begins with a self-check of the tester. It then beeps and displays a logo screen with a login message. By pressing the button on the bar code reader wand and moving the wand smoothly across the bar code on the back of your badge, you indicate that you are ready to start a test run. (Note: The tester will beep whenever a valid bar code is read.)

All other display screens are divided top-to-bottom into sections that show:

1. the station address, your badge number, and the ID number of the instrument under test;
2. the procedure step number and name;
3. measurement data obtained from the instrument;
4. operator instructions; and
5. function key designators (e.g.: "OK", "<>").

Normally, all that is necessary is to follow the instructions presented on the screen and press the "OK" button when requested at the completion of each step. Some steps may require that you visually confirm a meter reading or set one of the instrument switches to a particular position; otherwise, the test procedure is mostly automatic.

The tester firmware closely follows the steps of the established manual test and adjustment procedure for a particular type of HPI. However, some new steps have been added to support the additional features of the automated system. Specific test procedure steps are described in Sect. 5.2. The actual calibration procedure is detailed in document number MMD/CMNA1570. This document should be followed during the actual calibration of a Bicron instrument. The calibration procedure for the automated tester itself is given in document number MMD/CTSZ1010.

At the end of the test procedure, a test results data set is sent to the host PC through the RS-485 port. It is important to complete this step successfully because once a new test cycle is started or the tester station is powered off, the results of the previous test are cleared.

The special function key (labeled "<>") is used to break out of the normal procedure sequence. By pressing this key, the following menu will appear:

==>RETRY this step

BACKUP one step

SKIP this step

ABORT procedure

Use the up/down keys to select a choice (indicated by the arrow): then press "OK" to execute it.

Using this menu, it is possible to repeat any step of the procedure as needed, to skip a step that is impossible to complete, or to abort and jump to the end of the procedure. The ABORT function skips to the data transmission step of the procedure, which allows a partial data set to be sent to the host.

5.2 BICRON SURVEYOR MX CALIBRATION PROCEDURE

5.2.1 Preparation

EXECUTE SELFTEST:

The tester station automatically performs a quick self-test to verify the operation of the ADC and pulse generator circuits. It uses the 0-V and +5-V reference sources to test the ADC input. The +5-V reading should be within 10 mV of the expected value. It then generates a series of pulse trains at 60% of full scale for each of the Bicron's ranges. The actual pulse rate should be within 1% of the expected rate. If the test fails, a message is displayed, and the unit MUST be realigned and/or repaired before using it.

SCAN BAR CODES:

Scan your badge and the tags on the tester station, the electrostatic voltmeter, and the instrument under test. Press the button on the bar code reader wand and move it smoothly across the bar code strip in either direction to read it; a beep indicates a successful read. Note: If a number has already been entered, you may skip over it by pressing "OK."

SELECT PROBE TYPE:

Use the up/down keys to select the type of probe to be used with the instrument. This will determine the high-voltage setting required and the final switch settings for the instrument.

PREPARE INSTRUMENT:

Make sure the instrument batteries are fresh and set all the switches as indicated. Connect the multipin test harness cable from the tester station to the Bicron instrument and be sure that the connector locks in place. Connect the probe cable from the Bicron instrument to the electrostatic voltmeter. Prop the instrument on its side with the calibration potentiometers up and the meter facing you.

5.2.2 Electronic "As-Found" Checks

CHECK BATT VOLTAGE:

Press OK after the battery voltage reading is stable. If it is below 8 V, then something is wrong (make sure that the 9-V batteries are fresh).

CHECK MECH. ZERO:

Check the Bicron meter and enter the amount of offset from zero using the up/down keys. Note: one tic mark equals 2% of full scale.

CHECK ELEC. ZERO:

Press OK after the reading of the electrical zero is stable to record the as-found value.

CALC SCALE FACTORS:

Visually confirm the meter reading at a nominal 20% and 80% of full scale. Use the up/down keys to enter the ACTUAL reading on the instrument meter at each point. This establishes the conversion between meter millivolts and meter deflection and affects the accuracy of subsequent readings.

CHECK HV SUPPLY:

Enter the high-voltage value as indicated by the electrostatic voltmeter.

5.2.3 Functional Checks

CHECK RESET OPER:

Press and release the Bicron RESET button as prompted to test the reset function in each operating range.

CHECK AUDIO OPER:

Turn the "audio" switch ON. The tester will prompt you to confirm the audio function for each range. Press the "YES" button if the tone sounds correct or the "NO" button if not.

CHECK ALARM OPER:

Turn the "audio" switch OFF. The tester will then automatically test for the presence of the alarm signal in each range within a certain tolerance.

CHECK SCALER OPER:

On instruments equipped with a scaler, activate the scaler by pressing the trigger button. When the counting period expires (1 min), use the up/down keys to enter the actual count value.

CHECK RESP TIME:

Set the "response" switch to "SLOW." The tester will measure the response time from 10% to 90% of scale for each operating range. Repeat this for the "FAST" mode. After checking both the slow and fast response times, a summary report will be displayed.

CHECK READINGS:

The tester automatically checks the meter readings at 20%, 50%, and 80% of scale in each operating range. At each point, it waits several seconds for the reading to stabilize and then proceeds to the next setting. When finished, it displays a summary report of the measured errors. (Note: If the reading bounces and does not stabilize, you can force it to proceed to the next setting by pressing "OK.")

5.2.4 Electronic Adjustment and "As-Left" Checks

ADJUST MECH. ZERO:

Adjust the meter mechanical zero and enter the "as-left" offset (normally zero).

ADJUST ELEC. ZERO:

Adjust the "ZERO" potentiometer until the millivolts value is within the specified tolerance.

ADJUST HV SUPPLY:

Adjust the "HV" potentiometer to obtain the specified voltage and enter the actual reading from the electrostatic voltmeter.

ADJUST SPAN:

Adjust the "SPAN" potentiometer until the Bicron meter agrees with the electrostatic voltmeter reading.

CALC SCALE FACTORS:

Same as before; required because the zero and span adjustments may have been modified.

ADJUST RANGE CAL:

Adjust the appropriate range calibration potentiometer (X1000, X100, X10, X1) until the error amount is zero. When adjusting the X1 potentiometer, turn slowly to allow for the instrument response time. The reading takes more time (12–20 s) to settle in the X1 range.

CHECK READINGS:

Same as before; this is the "as-left" check.

5.2.5 Probe-Specific Adjustments and Functional Checks

ADJUST HV SUPPLY:

Adjust the "HV" potentiometer to obtain the specified voltage and enter the actual reading from the electrostatic voltmeter. For alpha probes, adjust the voltage as needed to reject the gamma response and enter the final setting.

CHECK ANTI-SAT:

This step is performed ONLY if the probe type is a G-M tube. Connect the probe cable from the Bicron to the tester station. Set the Bicron "ANTI-SAT" switch ON. Press "OK" to begin the test. The tester will automatically check the antisaturation function for each operating range.

CHECK RESPONSE:

Connect the probe cable from the Bicron to the actual probe that will be used with the instrument. Verify that the probe provides some response to radiation using a check source in each operating range of the instrument.

CHECKOUT SUMMARY:

Displays a summary of the functional checks.

5.2.6 Test Equipment Disconnect and Data Set Transmission

DISCONNECT INSTR.:

Set the switches as indicated and disconnect the test harness from the instrument. Verify the operation of the "BATT CHECK" function. If the meter does not return to zero, be sure that the mode switch is in the OPERATE position.

ENTER WORK HOURS:

Enter the number of hours spent on the activity codes listed to the nearest 0.1 hour using the up/down keys. Press OK to proceed to the next line.

SCAN COMMENT CODES:

Use the bar code wand to scan in up to five comment codes from the list provided. Press OK when finished.

TRANSMIT RESULTS:

When the host computer polls the tester station, the test results data set will be transmitted. If there is no activity within a few seconds, confirm that the tester station is properly connected and that the host is running the MONITOR program.

5.3 STATIONARY INSTRUMENT CALIBRATION

Several portable Bicron instruments have been modified for stationary use. The ABT can be used to calibrate these stationary instruments. If the instrument uses an alpha probe, select "ORNL Alpha" on the automated tester. For a gamma probe, select "Pancake GMT." In these instruments, the battery check is skipped since they use ac power. The HV setting for all stationaries is 900 V. There is also an internal "tone" switch that must be turned "OFF" during the calibration. Otherwise, the calibration proceeds as described in the procedure for a portable instrument. At the end of the calibration the internal "tone" switch must be set to "ON" for alpha probe instruments and "OFF" for those with a gamma probe.

5.4 TESTER STATION ALIGNMENT PROCEDURE

The tester station firmware also contains the necessary instructions for performing an alignment and calibration procedure on its own circuitry. This section describes the steps in that procedure. The actual calibration procedure for the automated tester is given in document number MMD/CTSZ1010. The test equipment required is

- Tester unit calibration adapter box and cable.
- Fluke model 5100 precision dc voltage source.
- Frequency counter.
- Digital voltmeter (DVM).
- Jumper wire.

To enter the alignment mode, remove the base cover of the tester station and set the MODE switch (SW1.4) into the CLOSED (align) position, as shown in Fig. 5.1.

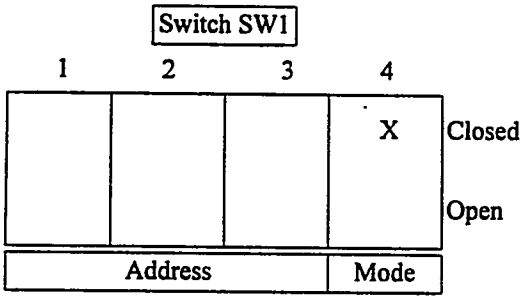


Fig. 5.1 Address selector switch settings for tester alignment.

Turn ON the tester power switch and follow the steps in the alignment procedure outlined below. Refer to the silk screen portion of Q6350-4 in Sect. 6.6 for the location of the test points (TPxx) and potentiometers (Rxx). Record the required observations on the calibration data sheet (see formal procedure MMD/CTSZ1010).

5.4.1 Procedure Steps

ALIGNMENT SETUP:

Disconnect Bicron harness cable from tester unit
Remove ground strap from Fluke 5100 output terminals
Connect adapter (+, -, ground) terminals to Fluke 5100 (+, -, ground) outputs
Connect adapter box cable to tester unit
Plug in the tester and turn it on. It will power up in the calibration/alignment mode.

ADJ BUFFER OFFSET

Connect DVM+ lead to TP5
Connect DVM- lead to TP10
Set adapter switches: Polarity (-), Mode (CMRR)
Set 5100: Function (DCV), Output (0.000 V), Opr/Stdby (OPR)
Record "as-found" value from DVM
Adjust R8 until DVM reads zero
Record "as-left" value from DVM

ADJ BUFFER GAIN

Set adapter switches: Polarity (-), Mode (NORM)
Connect jumper between TP4 and TP10
Set 5100: Function (DCV), Output (+2.000 V), Opr/Stdby (OPR)
Record "as-found" value from DVM
Adjust R11 until DVM reads +10.00 V
Record "as-left" value from DVM
Set 5100: Opr/Stdby (STDBY)
REMOVE JUMPER

ADJ BUFFER BALANCE

Set adapter switches: Polarity (+), Mode (CMRR)
Set 5100: Function (DCV), Output (+9.000 V), Opr/Stdby (OPR)
Record "as-found" value from DVM
Adjust R12 until DVM reads zero
Record "as-left" value from DVM
Set 5100: Opr/Stdby (STDBY)

VERIFY BUFFER ADJ

Set adapter switches: Polarity (+), Mode (NORM)
Set 5100: Function (DCV), Output (+2.000 V), Opr/Stdby (OPR)
Record "as-found" value from DVM
If DVM does not show +10.00 V, then:
    Adjust R11 until DVM reads +10.00 V
    Go back to step 4
Set 5100: Function (DCV), Output (0.000 V), Opr/Stdby (OPR)
Record "as-found" value from DVM
If voltmeter does not show 0.00 V, then:
    Go back to step 2

ADJ MUX OFFSET

Connect DVM+ lead to TP2
Record "as-found" value from DVM
Adjust R6 until DVM reads zero
Record "as-left" value from DVM

ADJUST S/H ZERO

Connect DVM+ lead to TP1
Record "as-found" value from DVM
Adjust R5 (sample-and-hold zero) until DVM reads zero
Record "as-left" value from DVM

ADJUST ADC OFFSET

Connect DVM+ lead to TP2
Set 5100: Function (DCV), Output (0.000 V), Opr/Stdby (OPR)
Adjust R6 until DVM reads 1.22 mV
Record "as-found" value from DVM
Adjust R9 until ADC value on display flickers between 0000 and 0001
Record ADC "as-left" value
Adjust R6 until DVM reads 0.00 mV
Verify that ADC value is 0000
Record ADC "as-left" value

ADJUST ADC GAIN

Set adapter switches: Polarity (+), Mode (NORM)
Connect DVM+ lead to TP2
Set 5100: Function (DCV), Output (+1.95 V), Opr/Stdby (OPR)
Record "as-found" data
Adjust R2 until displayed ADC value equals +1.95 V
Record "as-left" data

VERIFY ADC OPER

Disconnect DVM leads
Inject voltages at the following specified test settings:

Test points in millivolts:
0, 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000
Allowed tolerances:

From	0 mV	to	400 mV:	±2.0 mV
From	400 mV	to	2000 mV:	±0.5%

Observe and record the ADC response, as seen on the display, at each test setting

VERIFY PULSER OPER

Connect frequency counter (-, +) inputs to the adapter PULSE (+, -) terminals
The Hewlett Packard A Manual Level can be selected and the KHz filter may be used
Select pulse frequency test settings (given below) using the keypad

Test points in Hertz:
1.666, 4.166, 6.666, 16.666, 41.666, 66.666,
166.666, 416.666, 666.666, 1666.666, 4166.666, 6666.666
Allowed tolerance: ±0.5%

Measure and record the actual frequency at each test setting
Check pulse shape using oscilloscope and compare to the following specifications:

Pulse is negative going
Pulse height: Between -2.5 V and -5.0 V
Pulse width: Between 1.8 µs and 2.2 µs

Observe and record the pulse amplitude and width at each test setting

END ALIGNMENT

Disconnect adapter box
Set SW1.4 to the OPEN (operate) position
Power off and close tester case

6. Reference

This section contains a list of other documents that contain information about the Bicron Surveyor MX and the automated tester. This section also contains some drawings and short descriptions of the functions in both the host PC software and the automated tester firmware. The definition files and a short description of their contents are also given. Each function description has four parts:

- Function prototype in: Which header or source file contains the function prototype. If a header file is listed, then a ".c" or ".c51" file of the same name contains the function.
- Declaration: The declaration showing the function's inputs, outputs, and return value.
- Comments: A short description of the function.
- Function called by: List of other functions that call this function.

6.1 OTHER REFERENCES

There are other documents that have information on the Bicron Surveyor MX, the automated tester, and the calibration procedure. Here is a list of the sources with their contents.

README.TXT: Distributed with the software. Contains software installation instructions.

OPSGUIDE.TXT: Distributed with the software. Information on the system hardware and software and on the calibration procedure.

Q6350 Drawings: Block diagrams, circuit diagrams, and printed circuit board diagrams of the automated tester. Selected drawings are included in Sect. 6.6.

MMD/CMNA1570: Procedure for the Automated Electronic Calibration of the Bicron Surveyor MX and Bicron Surveyor MX Scaler Survey Meters.

MMD/CTSZ1010: Calibration Procedure for Automated Bicron MX Tester for Health Physics Instruments.

Manual No. 1084901: Technical Manual for the Bicron Surveyor MX Portable Count-Rate Meter with Digital Scaler Option. The manual number given is for Rev. A dated September 1991.

HPIC-SOP-1: Health Physics Instrument Committee Instrument Service Methodology. Revision 0 issued March 30, 1994.

6.2 HOST PC SOFTWARE DEFINITION FILES

calsheet.def: Format and definition of the calibration sheet layout and fields. The form is an array of strings with named fields for the data. See the file "formutil.def" for more definitions.

display.def:	Definitions of window handles and display locations for use with the Curses display window subsystem.
formutil.def:	Declaration of the STRING, FORM, and FIELD structures used to build the output forms.
tester.def:	Definitions used by the tester host PC software including probe types, instrument and calibration variables, and other data items.
workreq.def:	Format and definition of the work request form. The form is an array of strings with named fields for the data.

6.3 HOST PC SOFTWARE FUNCTIONS

6.3.1 ASY_FlushIn

Function prototype in: ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.

Declaration: void ASY_FlushIn(int port);

Comments: Flush the input buffer.

Function called by: None

6.3.2 ASY_FlushOut

Function prototype in: ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.

Declaration: void ASY_FlushOut(int port);

Comments: Flush the output buffer.

Function called by: None

6.3.3 ASY_GetStr

Function prototype in: ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.

Declaration: int ASY_GetStr(int port,char *str,int timeout);

Comments: Read characters from the port to a string until CR or timeout.

Function called by: GetReply

6.3.4 ASY_InitCom

Function prototype in: ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.

Declaration:	int ASY_InitCom(int port,int baud,int nbits,char parity,int nstop);
Comments:	Initialize the COM port for communication and set transmission options.
Function called by:	Monitor, TestComm

6.3.5 ASY_MapArgval

Function prototype in:	ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.
Declaration:	int ASY_MapArgval(int *tbl, int value);
Comments:	Return index of the given integer value in a table. The tables are defined in ASY_Util.c.
Function called by:	ASY_InitCom

6.3.6 ASY_PutStr

Function prototype in:	ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.
Declaration:	int ASY_PutStr(int port,char *str,int timeout);
Comments:	Write characters to the output queue and wait for it to empty.
Function called by:	SendCmd

6.3.7 ASY_StopCom

Function prototype in:	ASY_Util.h - Serial I/O using "C Asynch Mgr" library routines.
Declaration:	void ASY_StopCom(int port);
Comments:	Close the COM port.
Function called by:	Monitor, TestComm

6.3.8 CalcDspLoc

Function prototype in:	ListItem.h - Display a list of items for the user to select.
Declaration:	int CalcDspLoc(int ndx,int *row,int *col);
Comments:	Calculate the display location (row,col) given an index.
Function called by:	ListItems, ShowItem

6.3.9 CheckDir

Function prototype in: FileUtil.h - Utilities for managing files and directories.

Declaration: int CheckDir(char *dirnam,char *filnam,char * *dirlist,int listmax);

Comments: Find all files that match the given path (file and directory) name using the _dos_findfirst function. The path can have wildcards.

Function called by: SelectDir, SelectFile

6.3.10 CheckReply

Function prototype in: UpLoad.h - Upload dataset from instrument tester and save to disk file.

Declaration: int CheckReply(char *reply);

Comments: Calculate a checksum for the reply and see if it matches received checksum.

Function called by: FetchData

6.3.11 CheckStatus

Function prototype in: UpLoad.h - Upload dataset from instrument tester and save to disk file.

Declaration: int CheckStatus(int addr,char *status);

Comments: Send a message to the tester and see if it replies.

Function called by: Monitor, TestComm, FetchData

6.3.12 ClearMsgWn

Function prototype in: Display.h - Create and update display windows using Aspen Scientific Curses library.

Declaration: void ClearMsgWn(void);

Comments: Clear the message window.

Function called by: Monitor, SelectDir, SelectFile, Report

6.3.13 CloseLogFile

Function prototype in: LogFile.h - Write records to log file.

Declaration: void CloseLogFile(void);

Comments: Close the log file.

Function called by: UpdateLogFile

6.3.14 CloseTable

Function prototype in: TablUtil.h - Look up keyed record in table file. Read or update record data.

Declaration: void CloseTable(void);

Comments: Close the table file.

Function called by: FetchStaticItems, IncrReqNum

6.3.15 CmpDates

Function prototype in: SortDate.h - Sort date strings.

Declaration: int CmpDates(char * *date1,char * *date2);

Comments: Compare two date strings using the stricmp function.

Function called by: SortDates

6.3.16 CUR_DelWin

Function prototype in: CursUtil.h - Window creation and display functions using Curses library.

Declaration: void CUR_DelWin(struct _win_ *wn);

Comments: Delete a window.

Function called by: CUR_GetStr

6.3.17 CUR_GetChar

Function prototype in: CursProm.h - Data entry routines using the Curses library

Declaration: int CUR_GetChar(struct _win_ *wn,int row,int col,char *prompt,char *chr,char *match);

Comments: Get a character from an input string. Uses CUR_GetStr().

Function called by: Monitor

6.3.18 CUR_GetFloat

Function prototype in: CursProm.h - Data entry routines using the Curses library

Declaration: int CUR_GetFloat(struct _win_ *wn,int row,int col,char *prompt,char *fmt,float *val,float vmin,float vmax);

Comments: Get a floating point value from an input string. Uses CUR_GetStr()

Function called by: None

6.3.19 CUR_GetInt

Function prototype in: CursProm.h - Data entry routines using the Curses library

Declaration: int CUR_GetInt(struct _win_ *wn,int row,int col,char *prompt,int *val,int vmin,int vmax);

Comments: Get an integer value from an input string. Uses CUR_GetStr().

Function called by: Monitor

6.3.20 CUR_GetStr

Function prototype in: CursProm.h - Data entry routines using the Curses library

Declaration: char CUR_GetStr(struct _win_ *wn,int row,int col,char *prompt,char *valstr,int maxlen);

Comments: Read an input string from a data entry window.

Function called by: CUR_GetFloat, CUR_GetInt, CUR_GetChar

6.3.21 CUR_Halt

Function prototype in: CursUtil.h - Window creation and display functions using Curses library.

Declaration: void CUR_Halt(void);

Comments: End window session

Function called by: Monitor, Report

6.3.22 CUR_Init

Function prototype in: CursUtil.h - Window creation and display functions using Curses library.

Declaration: int CUR_Init(void);

Comments: Initialize Curses and set the display attributes.

Function called by: MonitorDisplay, ReportDisplay

6.3.23 CUR_MakeWin

Function prototype in: CursUtil.h - Window creation and display functions using Curses library.

Declaration: struct _win_ *CUR_MakeWin(int nr,int nc,int r1,int c1,char *title);

Comments: Create a window.

Function called by: MonitorDisplay, ReportDisplay

6.3.24 CUR_PutStr

Function prototype in: CursUtil.h - Window creation and display functions using Curses library.

Declaration: void CUR_PutStr(struct _win_ *wn,int row,int col,char *text);

Comments: Put text in a window.

Function called by: CUR_GetStr, CUR_MakeWin, CUR_ShowTime, ShowMsg, ShowStatus, ListItems, ShowItem

6.3.25 CUR_ShowTime

Function prototype in: CursUtil.h - Window creation and display functions using Curses library.

Declaration: void CUR_ShowTime(struct _win_ *wn,int row,int col,long tval);

Comments: Convert time to string, delete newline, display.

Function called by: Monitor

6.3.26 CvtDate

Function prototype in: FileUtil.h - Utilities for managing files and directories.

Declaration: void CvtDate(char *date1,char *date2)

Comments: Convert MM/DD/YY to YY-MM-DD format.

Function called by: MakeDir

6.3.27 CvtDateFmt

Function prototype in: SortDate.h - Sort date strings.

Declaration: void CvtDateFmt (char *datestr)

Comments: Convert date string from DDMMMYY to YY-MM-DD format.

Function called by: SortDates

6.3.28 FetchData

Function prototype in: UpLoad.h - Upload dataset from instrument tester and save to disk file.

Declaration: int FetchData(int addr,struct _iobuf *outfile);

Comments: Get data from tester and write to file.

Function called by: UploadData

6.3.29 FetchKeyedItems

Function prototype in: FetchTbl.h - Fetch specific data items from table files.

Declaration: void FetchKeyedItems(void);

Comments: Fetch instrument serial number and service designator, test equipment calibration due dates, and work comment text based on comment codes from SERNUM.TBL, CALDATE.TBL and COMMENTS.TBL.

Function called by: Monitor, Report, TestRpt

6.3.30 FetchStaticItems

Function prototype in: FetchTbl.h - Fetch specific data items from table files.

Declaration: void FetchStaticItems(void);

Comments: Fetch static data items from work request table (WORKREQ.TBL).

Function called by: Monitor, Report, TestRpt

6.3.31 FileExists

Function prototype in: FileUtil.h - Utilities for managing files and directories.

Declaration: int FileExists(char *path);

Comments: Does the file exist?

Function called by: UploadData

6.3.32 FRM_EraseField

Function prototype in: FormUtil.h - Functions to fill out, modify, and print the calibration forms.

Declaration: void FRM_EraseField(struct FormType *Form,int fldnum);

Comments: Erase a field in a form.

Function called by: None

6.3.33 FRM_MarkColumn

Function prototype in: FormUtil.h - Functions to fill out, modify, and print the calibration forms.

Declaration: void FRM_MarkColumn(struct FormType *Form,int fldnum,int offset,char *str);

Comments: Mark a column (row + offset) in a form.

Function called by: PrintCalSheet, PrintWorkReq

6.3.34 FRM_MarkField

Function prototype in: FormUtil.h - Functions to fill out, modify, and print the calibration forms.

Declaration: void FRM_MarkField(struct FormType *Form,int fldnum,char *str);

Comments: Write text into a form given the field number.

Function called by: PrintCalSheet, FRM_EraseField, PrintWorkReq

6.3.35 FRM_PrintForm

Function prototype in: FormUtil.h - Functions to fill out, modify, and print the calibration forms.

Declaration: int FRM_PrintForm(struct FormType *Form,char *filnam);

Comments: Write the form to an output file or the printer.

Function called by: PrintCalSheet, PrintWorkReq

6.3.36 FRM_WriteText

Function prototype in: FormUtil.h - Functions to fill out, modify, and print the calibration forms.

Declaration: void FRM_WriteText(struct FormType *Form,int row,int col,int len,char *str);

Comments: Write text into the specified form at specified location.

Function called by: FRM_MarkField, FRM_MarkColumn

6.3.37 GetDataRec

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int GetDataRec(void);

Comments: Read a single data record (80 character line) from the file into a string.

Function called by: LoadFileHeader, LoadEquipID, LoadVoltCheck, LoadReadCheck, LoadScalerCheck, LoadRespTime, LoadPerfCheck, LoadWorkHours, LoadCmtCodes, ReadDataSet

6.3.38 GetReply

Function prototype in: UpLoad.h - Upload dataset from instrument tester and save to disk file.

Declaration: int GetReply(int addr,char *reply);

Comments: Read the tester's reply from the communication port.

Function called by: CheckStatus, FetchData

6.3.39 IncrReqNum

Function prototype in: FetchTbl.h - Fetch specific data items from table files.

Declaration: int IncrReqNum(void);

Comments: Increment work request number in WORKREQ.TBL.

Function called by: Monitor

6.3.40 ListItems

Function prototype in: ListItem.h - Display a list of items for the user to select.

Declaration: void ListItems(void);

Comments: Display item selection list.

Function called by: ShowItem, SelectItem

6.3.41 LoadCmtCodes

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadCmtCodes (void)

Comments: Function not defined in header file. It reads the comment codes.

Function called by: ReadDataSet

6.3.42 LoadEquipID

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadEquipID(void);

Comments: Read badge, tester, meter, and instrument numbers.

Function called by: ReadDataSet

6.3.43 LoadFileHeader

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadFileHeader(void);

Comments: Read file date, time, and work request number.

Function called by: ReadDataSet

6.3.44 LoadPerfCheck

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadPerfCheck(void);

Comments: Read performance check values for range, audio, alarm, reset, antisat, and source.

Function called by: ReadDataSet

6.3.45 LoadReadCheck

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadReadCheck(void);

Comments: Read range value data.

Function called by: ReadDataSet

6.3.46 LoadRespTime

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadRespTime(void);

Comments: Read response time values.

Function called by: ReadDataSet

6.3.47 LoadScalerCheck

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadScalerCheck(void);

Comments: Read scaler check values.

Function called by: ReadDataSet

6.3.48 LoadVoltCheck

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadVoltCheck(void);

Comments: Read voltage checks.

Function called by: ReadDataSet

6.3.49 LoadWorkHours

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int LoadWorkHours(void);

Comments: Read work hours.

Function called by: ReadDataSet

6.3.50 MakeDir

Function prototype in: FileUtil.h - Utilities for managing files and directories.

Declaration: void MakeDir(char *date,char *dirnam);

Comments: Create a directory with a name same as the date.

Function called by: OpenLogFile, UploadData

6.3.51 MakePath

Function prototype in: FileUtil.h - Utilities for managing files and directories.

Declaration: void MakePath(char *dirnam,char *filnam,char *path);

Comments: Create a pathname to the given file in the given directory.

Function called by: CheckDir, OpenLogFile, ReadDataSet, UploadData

6.3.52 Monitor

Function prototype in: Monitor.c - Poll testers and process data.

Declaration: void main ();

Comments: This is the main compiled program that runs on the PC. It polls the instrument testers, uploads datasets, and prints reports.

Function called by: None

6.3.53 MonitorDisplay

Function prototype in: Display.h - Create and update display windows using Aspen Scientific Curses library.

Declaration: void MonitorDisplay(void);

Comments: Create and open the Monitor program display windows.

Function called by: Monitor

6.3.54 OpenLogFile

Function prototype in: LogFile.h - Write records to log file.

Declaration: int OpenLogFile(void);

Comments: Create log file directory and open log file (.LOG) for appending.

Function called by: UpdateLogFile, Monitor

6.3.55 OpenTable

Function prototype in: TablUtil.h - Look up keyed record in table file. Read or update record data.

Declaration: int OpenTable(char *filnam);

Comments: Open table file for update.

Function called by: FetchStaticItems, FetchKeyedItems, IncrReqNum

6.3.56 PassFail

Function prototype in: CalSheet.h - Fill out form with calibration data and print it.

Declaration: char *PassFail(int flag);

Comments: Returns condition of a calibration test in a format for printing in the form.

Function called by: PrintCalSheet

6.3.57 Pause

Function prototype in: Display.h - Create and update display windows using Aspen Scientific Curses library.

Declaration: void Pause(int nsecs);

Comments: Pause for a given number of seconds.

Function called by: Monitor, Report

6.3.58 Pause

Function prototype in: TestComm.c - Test dataset upload function.

Declaration: void Pause (int msec)

Comments: Pause for a given number of milliseconds. Local function in TestComm.c.

Function called by: TestComm

6.3.59 PrintCalSheet

Function prototype in: CalSheet.h - Fill out form with calibration data and print it.

Declaration: int PrintCalSheet(char *filnam);

Comments: Print the form with the calibration data to a file. The data should already exist.

Function called by: Monitor, Report, TestRpt

6.3.60 PrintWorkReq

Function prototype in: WorkReq.h - Generate work request form and print it.

Declaration: int PrintWorkReq(char *filnam);

Comments: Function that creates the work request form and prints it.

Function called by: Monitor, Report, TestRpt

6.3.61 ReadDataSet

Function prototype in: ReadFile.h - Read test results dataset from disk. Load values into working memory.

Declaration: int ReadDataSet(char *dirnam,char *filnam);

Comments: Read all values from the data file on disk into memory.

Function called by: Monitor, Report, TestRpt

6.3.62 ReadRecd

Function prototype in: TablUtil.h - Look up keyed record in table file. Read or update record data.

Declaration: int ReadRecd(char *key,char *data);

Comments: Find and read the keyed record.

Function called by: FetchStaticItems, FetchKeyedItems

6.3.63 Report

Function prototype in: Report.c - Read test results dataset; format and print report.

Declaration:	void main ();
Comments:	Compiled program that runs on the PC. It allows the user to read a test results dataset from disk and format and print a report.
Function called by:	None

6.3.64 ReportDisplay

Function prototype in:	Display.h - Create and update display windows using Aspen Scientific Curses library.
Declaration:	void ReportDisplay(void);
Comments:	Create and open the Report generation display windows.
Function called by:	Report

6.3.65 SeekRecd

Function prototype in:	TablUtil.h - Look up keyed record in table file. Read or update record data.
Declaration:	int SeekRecd(char *key);
Comments:	Search table file for record containing the given key value.
Function called by:	ReadRecd, UpdateRecd

6.3.66 SelectDir

Function prototype in:	Report.c - Read test results dataset; format and print report.
Declaration:	int SelectDir (char *dirnam)
Comments:	Let user select a data file directory from a list on screen. Local function in Report.c.
Function called by:	Report

6.3.67 SelectFile

Function prototype in:	Report.c - Read test results dataset; format and print report.
Declaration:	int SelectFile (char *dirnam, char *filnam)
Comments:	Let user select a data file (*.DAT) from a list on screen. Local function in Report.c.
Function called by:	Report

6.3.68 SelectItem

Function prototype in: ListItem.h - Display a list of items for the user to select.

Declaration: int SelectItem(struct _win_ *wn,char * *list,int lsize,int colwid,int *fdsp,int *ndx);

Comments: Display the item selection list and allow user to select an item.

Function called by: SelectDir, SelectFile

6.3.69 SendCmd

Function prototype in: UpLoad.h - Upload dataset from instrument tester and save to disk file.

Declaration: int SendCmd(int addr,char *cmd);

Comments: Send a command to the tester.

Function called by: CheckStatus, FetchData .

6.3.70 ShowItem

Function prototype in: ListItem.h - Display a list of items for the user to select.

Declaration: void ShowItem(int ndx,unsigned short attr);

Comments: Display the indexed item using the specified attribute.

Function called by: SelectItem

6.3.71 ShowMsg

Function prototype in: Display.h - Create and update display windows using Aspen Scientific Curses library.

Declaration: void ShowMsg(struct _win_ *wn,int row,int col,char *msg);

Comments: Display message text.

Function called by: Monitor, SelectDir, SelectFile, Report

6.3.72 ShowStatus

Function prototype in: Display.h - Create and update display windows using Aspen Scientific Curses library.

Declaration: void ShowStatus(int addr);

Comments: Show status of the tester stations.

Function called by: Monitor

6.3.73 SortDates

Function prototype in: SortDate.h - Sort date strings.

Declaration: void SortDates(char * *list,int num);

Comments: Sort a list of date strings using the quicksort algorithm.

Function called by: SelectDir

6.3.74 TestComm

Function prototype in: TestComm.c - Test dataset upload function.

Declaration: void main ()

Comments: Compiled program that runs on the PC. Test dataset upload function by testing comm ports and uploading data into *.DAT files.

Function called by: None

6.3.75. TestRpt

Function prototype in: TestRpt.c - Test report printing.

Declaration: int main (int argc, char **argv)

Comments: Compiled program that runs on the PC. Read test results dataset (.DAT file). Print formatted test results report and job ticket.

Function called by: None

6.3.76 UpdateLogFile

Function prototype in: LogFile.h - Write records to log file.

Declaration: int UpdateLogFile(void);

Comments: Open the day's log file and append calibration record for update to MAJIC database.

Function called by: Monitor

6.3.77 UpdateRecd

Function prototype in: TablUtil.h - Look up keyed record in table file. Read or update record data.

Declaration: int UpdateRecd(char *key,char *data):

Comments: Update the table record and write it to the file.

Function called by: IncrReqNum

6.3.78 UploadData

Function prototype in: UpLoad.h - Upload dataset from instrument tester and save to disk file.

Declaration: int UploadData(int addr,char *dirnam,char *filnam);

Comments: Upload a data file from the tester and store it in the proper directory. This is the top level routine to create output ".DAT" data files in this file.

Function called by: Monitor, TestComm

6.4 AUTOMATED TESTER FIRMWARE DEFINITION FILES

buttons.def: Button definition file. Defines the blank, OK, star, up, down, yes, and no buttons.

h82c52.def: Definitions for use with Harris 82C52 UART on the digital board. Defines the addresses of the UART's registers.

lsc_comm.def: Definitions for LSC communication. Contains a macro to calculate baudrate generator reload value (see 8051 handbook). Defines the message buffers.

rsc_comm.def: Defines buffers for use with the remote serial channel (RSC). This is the RS-422 input/output port.

systimer.def: Defines the software timers.

t6963c.def: Command codes for Toshiba T6963C LCD controller.

tester.def: Definitions of global data used by instrument tester firmware. Includes the probe type, operating mode, function codes, and other data definitions used in the calibration.

tlx1013.def: Panel-specific definitions for Toshiba TLX-1013 LCD panel.

tstrlogo.def: Definition of the automated tester logo.

unitspec.def: Unit specific register and address assignments. Includes the Bicron operating ranges, peripheral port addresses, and ADC and pulse counter signal selection codes.

6.5 AUTOMATED TESTER FUNCTIONS

6.5.1 AdjustRangeCal

Function prototype in: ChkVolt.h - Instrument test procedures.

Declaration: BYTE AdjustRangeCal (void);

Comments: Prompts the user to adjust the Range Cal potentiometers if required.

Function called by: ExecTestProc

6.5.2 AdjustSpan

Function prototype in: ChkVolt.h - Instrument test procedures.

Declaration: BYTE AdjustSpan (void);

Comments: Prompts the user to adjust the span.

Function called by: ExecTestProc

6.5.3 ALP_AdjADCGain

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjADCGain (void);

Comments: Instructions for adjusting the ADC gain (R2).

Function called by: ExecAlignProc

6.5.4 ALP_AdjADCOffset

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjADCOffset (void);

Comments: Instructions for adjusting the ADC offset (R9)

Function called by: ExecAlignProc

6.5.5 ALP_AdjBufBalance

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjBufBalance (void);

Comments: Instructions for adjusting the Buffer Balance (adjust R12).

Function called by: ExecAlignProc

6.5.6 ALP_AdjBufGain

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjBufGain (void);

Comments: Instructions for adjusting the Buffer Gain (adjust R11).

Function called by: ExecAlignProc

6.5.7 ALP_AdjBufOffset

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjBufOffset (void);

Comments: Instructions for adjusting the Buffer Offset.

Function called by: ExecAlignProc

6.5.8 ALP_AdjMuxOffset

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjMuxOffset (void);

Comments: Instructions for adjusting the mux offset potentiometer (R6).

Function called by: ExecAlignProc

6.5.9 ALP_AdjSampZero

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_AdjSampZero (void);

Comments: Instructions for adjusting the sample/hold zero (R5).

Function called by: ExecAlignProc

6.5.10 ALP_Quit

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_Quit (void);

Comments: End alignment procedure.

Function called by: ExecAlignProc

6.5.11 ALP_Setup

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_Setup (void);

Comments: Prompt for equipment setup. Provides the user with instructions on setting up the test apparatus.

Function called by: ExecAlignProc

6.5.12 ALP_VerifyADC

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_VerifyADC (void);

Comments: Verify ADC alignment using external voltage source.

Function called by: ExecAlignProc

6.5.13 ALP_VerifyBuf

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_VerifyBuf (void);

Comments: Instructions for verifying the buffer alignment.

Function called by: ExecAlignProc

6.5.14 ALP_VerifyPulser

Function prototype in: Align.h - Tester alignment procedures.

Declaration: BYTE ALP_VerifyPulser (void);

Comments: Verify pulse generator output using external instruments.

Function called by: ExecAlignProc

6.5.15 Beep

Function prototype in: BarCode.h - Read packet from HP bar code reader via local serial channel.

Declaration: void Beep (BYTE n);

Comments: Tell bar code reader to generate beeps (each beep lasts ~80 ms).

Function called by: CheckReadings, CheckRespTime, ErrMsg, ReadBarCode, XmtDataset

6.5.16 BicrTest

Function prototype in: BicrTest.c51 - Control program for the Automated Bicron Tester.

Declaration: int main (void);

Comments: Main program. Initializes the tester and sets up communication. Executes test or alignment procedure based on the operating mode.

Function called by: None

6.5.17 CalcAvg

Function prototype in: AD574A.h - Initiate 12-bit A/D conversion; read value; convert to volts.

Declaration: WORD CalcAvg (WORD lastavg, WORD newval, BYTE n);

Comments: Calculate a running average.

Function called by: ALP_AdjADCGain, ALP_VerifyADC, CheckBattery, ReadMeter, TestADC

6.5.18 CalcFreqDiv

Function prototype in: PulseGen.h - Calculate and generate a desired output frequency.

Declaration: BYTE CalcFreqDiv (long freq, BYTE *d1, BYTE *d2, BYTE *d3);

Comments: Calculate frequency divisors required to generate the given frequency from the PXO-1000 Crystal-controlled oscillator and the 74LS163 counter.

Function called by: InjectPulses

6.5.19 CalcSclFac

Function prototype in: ChkRead.h - Instrument test procedures.

Declaration: BYTE CalcSclFac (void);

Comments: Prompt the user for the actual percent full-scale reading from the instrument and calculate scale conversion factors.

Function called by: ExecTestProc

6.5.20 CheckAlarm

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration: BYTE CheckAlarm (void);

Comments: Prompt user to turn audio off and check if the over-range alarm works in each range.

Function called by: ExecTestProc

6.5.21 CheckAntiSat

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration: BYTE CheckAntiSat (void);

Comments: Check antisaturation operation.

Function called by: ExecTestProc

6.5.22 CheckAudio

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration: BYTE CheckAudio (void);

Comments: Prompt user to turn on audio and check if audio works in each range.

Function called by: ExecTestProc

6.5.23 CheckBattery

Function prototype in: ChkVolt.h - Instrument test procedures.

Declaration: BYTE CheckBattery (void);

Comments: Measure and record the instrument's battery voltage.

Function called by: ExecTestProc

6.5.24 CheckButtons

Function prototype in: Buttons.h - Interpret operator function selections.

Declaration: BYTE CheckButtons (void);

Comments: Check for button hit; return selected function code depending on which button was pressed. Returns NULL if there was no switch interrupt.

Function called by: AdjustRangeCal, AdjustSpan, ALP_AdjADCGain, ALP_VerifyADC, ALP_VerifyPulser, CalcSclFac, CheckAlarm, CheckAntiSat, CheckAudio, CheckBattery, CheckElecZero, CheckHVSupply, CheckMechZero, CheckReadings, CheckReset, CheckRespTime, CheckScaler, CheckSourceResp, EnterWorkHours, ErrMsg, EscapeFunc, ReadBarCode, SelectProbeType, ShowADC, ShowLogo, TestComm, WaitForOK, WaitHost

6.5.25 CheckElecZero

Function prototype in: ChkVolt.h - Instrument test procedures.

Declaration: BYTE CheckElecZero (BYTE nf);

Comments: Measure and record the instrument's electronic zero.

Function called by: ExecTestProc

6.5.26 CheckHVSupply

Function prototype in: ChkVolt.h - Instrument test procedures.

Declaration: BYTE CheckHVSupply (BYTE nf, int volts);

Comments: Measure and adjust the high-voltage supply. The voltmeter is used to measure the voltage.

Function called by: ExecTestProc

6.5.27 CheckMechZero

Function prototype in: ChkVolt.h - Instrument test procedures.

Declaration: BYTE CheckMechZero (BYTE nf);

Comments: Check and adjust the instrument's mechanical zero offset.

Function called by: ExecTestProc

6.5.28 CheckReadings

Function prototype in: ChkRead.h - Instrument test procedures.

Declaration: BYTE CheckReadings (BYTE flag);

Comments: Check readings at 20, 50, and 80% of each range.

Function called by: ExecTestProc

6.5.29 CheckReset

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration: BYTE CheckReset (void);

Comments: Check the operation of the RESET function.

Function called by: ExecTestProc

6.5.30 CheckRespTime

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration: BYTE CheckRespTime (void);

Comments: Check response times for each range (x1-x1000) and mode (slow and fast).

Function called by: ExecTestProc

6.5.31 CheckScaler

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration: BYTE CheckScaler (void);

Comments: Check scaler operation and let the user enter the actual counts per minute (cpm) as seen on the detector. Also save the nominal cpm read by the tester.

Function called by: ExecTestProc

6.5.32 CheckSourceResp

Function prototype in: ChkPerf.h - Instrument test procedures.

Declaration:	BYTE CheckSourceResp (void);
Comments:	Connect a probe to the detector and check for meter response to a check source.
Function called by:	ExecTestProc

6.5.33 CheckSummary

Function prototype in:	ChkPerf.h - Instrument test procedures.
Declaration:	BYTE CheckSummary (void);
Comments:	Display summary report of function checks.
Function called by:	ExecTestProc

6.5.34 ClearResults

Function prototype in:	BicrTest.c51 - Control program for the Automated Bicron Tester.
Declaration:	BYTE ClearResults (void);
Comments:	Clear and initialize all data variables. Local function in BicrTest.c51.
Function called by:	BicrTest

6.5.35 ConnectInstr

Function prototype in:	Hookup.h - Instrument test procedures.
Declaration:	BYTE ConnectInstr (void);
Comments:	Display instructions on preparing the instrument for testing.
Function called by:	ExecTestProc

6.5.36 CountPulses

Function prototype in:	PulseCtr.h - Pulse counting function.
Declaration:	BYTE CountPulses (int msec, WORD *count);
Comments:	Setup T1 as a 16-bit pulse counter. Count for the specified millisecond interval; then check the counter value.
Function called by:	ReadAlarmSig, TestRate

6.5.37 DiscInstr

Function prototype in: Hookup.h - Instrument test procedures.

Declaration: BYTE DiscInstr (void);

Comments: Display instructions on disconnecting the instrument.

Function called by: ExecTestProc

6.5.38 EnterComments

Function prototype in: Hookup.h - Instrument test procedures.

Declaration: BYTE EnterComments (void);

Comments: Prompt the user to scan in up to five comment codes.

Function called by: ExecTestProc

6.5.39 EnterWorkHours

Function prototype in: Hookup.h - Instrument test procedures.

Declaration: BYTE EnterWorkHours (void);

Comments: Prompt the user to enter the number of hours spent on calibration, adjustment, and maintenance.

Function called by: ExecTestProc

6.5.40 ErrMsg

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ErrMsg(char *str);

Comments: Beep and display an error message.

Function called by: ExecSelfTest

6.5.41 EscapeFunc

Function prototype in: Buttons.h - Interpret operator function selections.

Declaration: BYTE EscapeFunc (void);

Comments:	Display escape function menu and let user select escape function from RETRY, BACKUP, SKIP, and ABORT.
Function called by:	ExecAlignProc, ExecTestProc

6.5.42 ExecAlignProc

Function prototype in:	BicrTest.c51 - Control program for the Automated Bicron Tester.
Declaration:	BYTE ExecAlignProc (void);
Comments:	Execute tester alignment procedure steps: buffers, mux, ADC gain, and pulser. Local function in BicrTest.c51.
Function called by:	BicrTest

6.5.43 ExecSelfTest

Function prototype in:	SelfTest.h - Check ADC circuit and pulse generator function.
Declaration:	BYTE ExecSelfTest (void);
Comments:	Test the operation of the ADCs and the pulse generator. This function calls the functions TestADC and TestPulseGen.
Function called by:	ExecTestProc

6.5.44 ExecTestProc

Function prototype in:	BicrTest.c51 - Control program for the Automated Bicron Tester.
Declaration:	BYTE ExecTestProc (void);
Comments:	Execute Bicron MX test procedure steps: as-found, functional, and as-left checks. The program also performs probe-specific checks. It then transmits the data set to the host PC. Local function in BicrTest.c51.
Function called by:	BicrTest

6.5.45 I82C59A_Handler

Function prototype in:	I82C59A.h - Set up the Intel 8259 interrupt controller (using INT0).
Declaration:	interrupt [IE0_VECTOR] void I82C59A_Handler (void);
Comments:	Interrupt service routine to handle interrupts from the I82C59A interrupt controller.
Function called by:	None

6.5.46 Init_I82C59A

Function prototype in: I82C59A.h - Set up the Intel 8259 interrupt controller (using INT0).

Declaration: void Init_I82C59A (void);

Comments: Initialize the interrupt controller by writing the initialization command words and output command words to it.

Function called by: BicrTest, TestComm

6.5.47 InitBarCode

Function prototype in: BarCode.h - Read packet from HP bar code reader via local serial channel.

Declaration: BYTE InitBarCode (void);

Comments: Disable automatic tone generation.

Function called by: ExecTestProc

6.5.48 InitPorts

Function prototype in: PortCtl.h - Set/clear selected bits in various I/O ports.

Declaration: void InitPorts (void);

Comments: Set up external ports A, B, C (see UNITSPEC.DEF) on I82C55A, disable pulse injection, and select default instrument operating range.

Function called by: BicrTest, TestComm

6.5.49 InitRegisters

Function prototype in: SysInit.h - Initialize 8051 control registers.

Declaration: void InitRegisters (void);

Comments: Set 8051 control registers, initial timer/counter operating modes, and port 1 pins. The assignments are

- T0: Delay timer
- T1: Baud rate generator for LSC/pulse counter
- INT0: Intel 82C59A interrupt controller
- INT1: ---

Function called by: BicrTest, TestComm

6.5.50 InjectPulses

Function prototype in: PulseGen.h - Calculate and generate a desired output frequency.

Declaration: BYTE InjectPulses (int range, int percent);

Comments: Select the instrument operating range, calculate the required frequency divisors, set the output pulse frequency, and enable pulse injection. Calculate the error between the nominal and actual output frequencies.

Function called by: AdjustRangeCal, ALP_VerifyPulser, CalcSclFac, CheckAlarm, CheckAntiSat, CheckAudio, CheckElecZero, CheckHVSupply, CheckMechZero, CheckReadings, CheckReset, CheckRespTime, CheckScaler, CheckSourceResp, TestPulseGen, TestRate

6.5.51 LCD_ClearRow

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_ClearRow (BYTE row);

Comments: Copy blanks to a text row on the LCD.

Function called by: CheckAntiSat, CheckReadings, CheckReset, CheckRespTime, CheckSourceResp, ErrMsg, ScanBarCodes, TestRate, WaitForOK

6.5.52 LCD_DrawPic

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_DrawPic (const char *picaddr, WORD grfaddr, BYTE nlines, BYTE ncols);

Comments: Write graphics on the LCD panel.

Function called by: ShowButtons, ShowLogo

6.5.53 LCD_Enable

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_Enable (BYTE flag);

Comments: Turn the LCD display on or off depending on the input variable "flag."

Function called by: ExecAlignProc, ShowLogo. TestComm

6.5.54 LCD_GrafFill

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_GrafFill (BYTE val);

Comments: Fill graphic RAM area (both half screens) of the LCD.

Function called by: ExecAlignProc, ShowLogo, TestComm

6.5.55 LCD_Init

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_Init (void);

Comments: Specify T6963C RAM assignments and set the display mode.

Function called by: ExecAlignProc, ShowLogo, TestComm

6.5.56 LCD_PutCmd

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_PutCmd (BYTE x);

Comments: Wait for "ready" condition; then output command byte to LCD controller.

Function called by: LCD_ClearRow, LCD_DrawPic, LCD_Enable, LCD_GrafFill, LCD_Init, LCD_SetAddr, LCD_TextFill, LCD_WriteText

6.5.57 LCD_PutData

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_PutData (BYTE x);

Comments: Wait for "ready" condition; then output data byte to LCD controller.

Function called by: LCD_ClearRow, LCD_DrawPic, LCD_GrafFill, LCD_Init, LCD_SetAddr, LCD_TextFill, LCD_WriteText

6.5.58 LCD_SetAddr

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_SetAddr (WORD ramaddr);

Comments: Set the address pointer on the LCD controller. Points to LCD panel display RAM area.

Function called by: LCD_ClearRow, LCD_DrawPic, LCD_GrafFill, LCD_TextFill, LCD_WriteText

6.5.59 LCD_TextFill

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_TextFill (BYTE val);

Comments: Fill text RAM area (both half screens) of the LCD.

Function called by: ExecAlignProc, NewScreen, ShowLogo, TestComm

6.5.60 LCD_WriteText

Function prototype in: TLX1013.h - Display control functions for TLX-1013 LCD panel using T6963C LCD controller.

Declaration: void LCD_WriteText (BYTE row, BYTE col, char *text);

Comments: Write text to the LCD panel.

Function called by: CheckSummary, DiscInstr, EnterComments, EnterWorkHours, ErrMsg, NewScreen, ShowLogo, ShowString, ShowText, ShowValue

6.5.61 LSC_Init

Function prototype in: LSC_Comm.h - LSC communication with the Hewlett Packard bar code reader.

Declaration: void LSC_Init (WORD baudrate);

Comments: Initialize the LSC at a specified baud rate.

Function called by: Beep, InitBarCode, ReadBarCode

6.5.62 LSC_IntHandler

Function prototype in: LSC_Comm.h - LSC communication with the Hewlett Packard bar code reader.

Declaration: interrupt [LSC_VECTOR] void LSC_IntHandler (void);

Comments: Interrupt service routine to handle LSC interrupts. It copies incoming bytes to a message buffer until a carriage return (CR) is received.

Function called by: None

6.5.63 LSC_PutByte

Function prototype in: LSC_Comm.h - LSC communication with the Hewlett Packard bar code reader.

Declaration: void LSC_PutByte (BYTE n);

Comments: Disable LSC interrupt and output a single byte.

Function called by: None

6.5.64 LSC_PutString

Function prototype in: LSC_Comm.h - LSC communication with the Hewlett Packard bar code reader.

Declaration: void LSC_PutString (char *str);

Comments: Disable LSC interrupt and output a string.

Function called by: Beep, InitBarCode

6.5.65 LSC_RcvDisable

Function prototype in: LSC_Comm.h - LSC communication with the Hewlett Packard bar code reader.

Declaration: void LSC_RcvDisable (void);

Comments: Disable reception and interrupts on the LSC.

Function called by: ReadBarCode

6.5.66 LSC_RcvEnable

Function prototype in: LSC_Comm.h - LSC communication with the Hewlett Packard bar code reader.

Declaration: void LSC_RcvEnable (void);

Comments: Clear interrupt flags and enable reception and interrupts on the LSC.

Function called by: ReadBarCode

6.5.67 NewScreen

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void NewScreen (TEXT *text, BYTE buttonset);

Comments: Display the standard screen heading along with the specified text and set of buttons.

Function called by: AdjustRangeCal, AdjustSpan, ALP_AdjADCGain, ALP_AdjADCOffset, ALP_AdjBufBalance, ALP_AdjBufGain, ALP_AdjBufOffset, ALP_AdjMuxOffset, ALP_AdjSampZero, ALP_Quit, ALP_Setup, ALP_VerifyADC, ALP_VerifyBuf, ALP_VerifyPulser, CalcSclFac, CheckAlarm, CheckAntiSat, CheckAudio, CheckBattery, CheckElecZero, CheckHVSupply, CheckMechZero, CheckReadings, CheckReset, CheckRespTime, CheckScaler, CheckSourceResp, CheckSummary, ConnectInstr, DiscInstr, EnterComments, EnterWorkHours, EscapeFunc, ExecSelfTest, ScanBarCodes, SelectProbeType, TestComm, XmtDataset

6.5.68 Pause

Function prototype in: SysTimer.h - Set 8051 counter/timer (T0) to interrupt at fixed interval. Update software timers at each interval.

Declaration: void Pause (int msec);

Comments: Set a software timer and wait the given number of milliseconds until it expires.

Function called by: Beep, SelectSignal, TestRate

6.5.69 Read_AD574A

Function prototype in: AD574A.h - Initiate 12 bit A/D conversion; read value; convert to volts.

Declaration: BYTE Read_AD574A (WORD *adcval);

Comments: Read a 12-bit value from ADC.

Function called by: ReadMillivolts, ShowADC

6.5.70 ReadAlarmSig

Function prototype in: ChkPerf.c51 - Instrument test procedures.

Declaration: BYTE ReadAlarmSig (void);

Comments: Select audio pulse train for input, count pulses for approximately 100 ms, and check for the presence of the alarm signal (> 250 Hz). Used as a local function.

Function called by: CheckAlarm

6.5.71 ReadBarCode

Function prototype in: BarCode.h - Read packet from HP bar code reader via LSC.

Declaration: BYTE ReadBarCode (char *str, BYTE maxlen);

Comments: Enable LSC reception, wait for valid bar code scan, and copy data to a string.

Function called by: EnterComments, ScanBarCodes

6.5.72 ReadMeter

Function prototype in: Meter.h - Read meter millivolts, convert to equivalent percent full scale (% FS), cpm, and HV units.

Declaration: BYTE ReadMeter (void);

Comments: Read meter millivolts; convert to equivalent % FS, cpm, and HV units from a running average.

Function called by: AdjustRangeCal, AdjustSpan, CalcSclFac, CheckAlarm, CheckAntiSat, CheckAudio, CheckElecZero, CheckHVSupply, CheckReadings, CheckReset, CheckRespTime

6.5.73 ReadMillivolts

Function prototype in: AD574A.h - Initiate 12-bit A/D conversion; read value; convert to volts.

Declaration: BYTE ReadMillivolts (int *mvolts);

Comments: Read ADC value and convert to millivolts using the current scale factor.

Function called by: ALP_AdjADCGain, ALP_VerifyADC, CheckBattery, ReadMeter, TestADC

6.5.74 RSC_GetString

Function prototype in: RSC_Comm.h - RSC I/O. RSC0 = Multidrop RS-422 (RS-485) communication with host PC. RSC1 = RS-232 communication with target instrument.

Declaration: BYTE RSC_GetString (BYTE port);

Comments: Read incoming bytes from the RSC into a buffer until a CR is read. If it is a status request, send an immediate reply.

Function called by: I82C59A_Handler

6.5.75 RSC_Init

Function prototype in: RSC_Comm.h - RSC I/O. RSC0 = Multidrop RS-422 (RS-485) communication with host PC. RSC1 = RS-232 communication with target instrument.

Declaration: void RSC_Init (BYTE port);

Comments: Initialize the RSC UARTs.

Function called by: BicrTest, TetstComm

6.5.76 RSC_PutString

Function prototype in: RSC_Comm.h - RSC I/O. RSC0 = Miltidrop RS-422 (RS-485) communication with host PC. RSC1 = RS-232 communication with target instrument.

Declaration: BYTE RSC_PutString (BYTE port, char *str);

Comments: Transmit a string over the RSC.

Function called by: RSC_GetString, SendRecd

6.5.77 RSC_RcvEnable

Function prototype in: RSC_Comm.h - RSC I/O. RSC0 = Miltidrop RS-422 (RS-485) communication with host PC. RSC1 = RS-232 communication with target instrument.

Declaration: void RSC_RcvEnable (BYTE port);

Comments: Flush UART registers and clear message buffer.

Function called by: BicrTest, TetstComm, WaitHost

6.5.78 ScanBarCodes

Function prototype in: Hookup.h - Instrument test procedures.

Declaration: BYTE ScanBarCodes (void);

Comments: Prompt the user to scan the bar codes on the user's badge, the tester, voltmeter, and Bicron instrument.

Function called by: ExecTestProc

6.5.79 SelectProbeType

Function prototype in: Hookup.h - Instrument test procedures.

Declaration: BYTE SelectProbeType (void):

Comments: Display the probe selection menu and let the user select a probe type.

Function called by: ExecTestProc

6.5.80 SelectPulses

Function prototype in: PortCtl.h - Set/clear selected bits in various I/O ports.

Declaration: void SelectPulses (BYTE signum);

Comments: Set pulse counter signal select bits (P1.2 and P1.3).

Function called by: ReadAlarmSig, TestRate

6.5.81 SelectRange

Function prototype in: PortCtl.h - Set/clear selected bits in various I/O ports.

Declaration: BYTE SelectRange (BYTE range);

Comments: Select the specified operating range (X1-X1000, battery or high voltage).

Function called by: AdjustSpan, CheckHVSupply, InitPorts, InjectPulses

6.5.82 SelectSignal

Function prototype in: PortCtl.h - Set/clear selected bits in various I/O ports.

Declaration: void SelectSignal (BYTE signum);

Comments: Set the ADC signal select bits and scale factor based on the selected signal.

Function called by: ALP_AdjBufOffset, ALP_VerifyADC, CheckBattery, ReadMeter, TestADC

6.5.83 SendRecd

Function prototype in: XmtData.c51 - Transmit complete dataset to host system.

Declaration: BYTE SendRecd (char *str);

Comments: Take a record string, append a checksum and carriage return, and send it to the host. Check the confirmation code and retry if there is an error.

Function called by: XmtDataset

6.5.84 SetPulseFreq

Function prototype in: PulseGen.h - Calculate and generate a desired output frequency.

Declaration: BYTE SetPulseFreq (BYTE d1, BYTE d2, BYTE d3);

Comments: Apply the precalculated divisors to the base frequency.

Function called by: InjectPulses

6.5.85 ShowADC

Function prototype in: Align.c51 - Tester alignment procedures.

Declaration: BYTE ShowADC (BYTE row);

Comments: Display the ADC value on the LCD panel.

Function called by: ALP_AdjADCOffset

6.5.86 ShowBarCodes

Function prototype in: Hookup.h - Instrument test procedures.

Declaration: void ShowBarCodes (void);

Comments: Display the user's badge, tester, voltmeter, and Bicron instrument numbers.

Function called by: ScanBarCodes

6.5.87 ShowButtons

Function prototype in: Buttons.h - Interpret operator function selections.

Declaration: void ShowButtons (void);

Comments: Display button icons for specified functions on the LCD panel. Some of the functions are OK, up, down, yes, and no.

Function called by: NewScreen

6.5.88 ShowCPM

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowCPM (BYTE row);

Comments: Display the current operating range and both the nominal and actual CPM values.

Function called by: CheckScaler

6.5.89 ShowLogo

Function prototype in: ShowLogo.h - Display logo screen on LCD panel.

Declaration: void ShowLogo (void);

Comments: Initialize display panel, display logo screen on LCD, wait for button hit, then clear screen.

Function called by: ExecTestProc

6.5.90 ShowPctErr

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowPctErr (BYTE row);

Comments: Display the percentage error between nominal and actual percent full scale readings.

Function called by: AdjustRangeCal, CheckReadings

6.5.91 ShowPFS

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowPFS (BYTE row);

Comments: Display the current operating range and both the nominal and actual percent full scale (% FS) values.

Function called by: AdjustRangeCal, CalcSclFac, CheckAlarm, CheckAntiSat, CheckAudio, CheckReadings, CheckReset, CheckRespTime

6.5.92 ShowRange

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowRange (BYTE row);

Comments: Display the current operating range (X1–X1000, battery or high voltage).

Function called by: CheckSourceResp

6.5.93 ShowString

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowString (BYTE row, BYTE col, char *tag, char *str);

Comments: Display a tag followed by a string.

Function called by: ShowADC, ShowBarCodes, ShowCPM, ShowPFS, ShowRange

6.5.94 ShowText

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowText (BYTE row, char *text);

Comments: Display the given text at the specified row.

Function called by: CheckAlarm, CheckAntiSat, CheckAudio, CheckBattery, CheckReadings, CheckReset, CheckRespTime, CheckSourceResp, EnterWorkHours, EscapeFunc, ScanBarCodes, SelectProbeType, WaitForOK, XmtDataset

6.5.95 ShowValue

Function prototype in: Display.h - Routines to display text and graphic items on LCD panel.

Declaration: void ShowValue (BYTE row, BYTE col, char *tag, long val, BYTE width, BYTE dp, char *units);

Comments: Display a tag, formatted value, and units.

Function called by: AdjustSpan, ALP_AdjADCGain, ALP_VerifyADC, ALP_VerifyPulser, CalcSclFac, CheckBattery, CheckElecZero, CheckHVSupply, CheckMechZero, CheckReadings, CheckRespTime, NewScreen, ShowCPM, ShowPctErr, ShowPFS, TestADC, TestRate

6.5.96 StartSysTimer

Function prototype in: SysTimer.h - Set 8051 counter/timer (T0) to interrupt at fixed interval. Update software timers at each interval.

Declaration: void StartSysTimer (void);

Comments: Load and start the T0 timer and enable it to interrupt the 8051.

Function called by: BicrTest

6.5.97 StopSysTimer

Function prototype in: SysTimer.h - Set 8051 counter/timer (T0) to interrupt at fixed interval. Update software timers at each interval.

Declaration: void StopSysTimer (void);

Comments: Stop the T0 timer and disable interrupts to the 8051.

Function called by: None

6.5.98 SysTimerInt

Function prototype in: SysTimer.h - Set 8051 counter/timer (T0) to interrupt at fixed interval. Update software timers at each interval.

Declaration: interrupt [T0_VECTOR] void SysTimerInt (void);

Comments: Interrupt service routine to decrement software timers at each T0 interrupt.

Function called by: None

6.5.99 TestADC

Function prototype in: SelfTest.h - Check ADC circuit and pulse generator function.

Declaration: BYTE TestADC (void);

Comments: Test ADCs using 0-V and +5-V reference sources.

Function called by: ExecSelfTest

6.5.100 TestPulseGen

Function prototype in: SelfTest.h - Check ADC circuit and pulse generator function.

Declaration: BYTE TestPulseGen (void);

Comments: Test the pulse generator at all ranges (X1–X1000).

Function called by: ExecSelfTest

6.5.101 TestRate

Function prototype in: SelfTest.h - Check ADC circuit and pulse generator function.

Declaration: BYTE TestRate (BYTE range, BYTE pct);

Comments: Generate pulses at a specified rate given the range and calculate the amount of absolute error between the nominal and actual pulse rates.

Function called by: TestPulseGen

6.5.102 WaitForOK

Function prototype in: Buttons.h - Interpret operator function selections.

Declaration: BYTE WaitForOK (BYTE row);

Comments: Prompt user to press OK; then wait for it.

Function called by: ALP_AdjBufBalance, ALP_AdjBufGain, ALP_AdjBufOffset, ALP_AdjMuxOffset, ALP_AdjSampZero, ALP_Quit, ALP_Setup, ALP_VerifyBuf, CheckAlarm, CheckAntiSat, CheckAudio, CheckReadings, CheckRespTime, CheckSummary, ConnectInstr, DiscInstr, XmtDataset

6.5.103 WaitHost

Function prototype in: XmtData.c51 - Transmit complete dataset to host system.

Declaration: WaitHost (BYTE *confirm);

Comments: Wait for message from host and read the confirmation code.

Function called by: SendRecd, XmtDataset

6.5.104 XmtDataset

Function prototype in: XmtData.c51 - Transmit complete dataset to host system.

Declaration: BYTE XmtDataset (void);

Comments: Wait for the host to poll and then begin sending the elements of the dataset. The dataset includes the equipment identifiers, work hours, comment codes, and checks of the voltage, meter reading, scaler, response time, and performance.

Function called by: ExecTestProc, TestComm

6.6 DRAWINGS

This section contains the Q6350 set of illustrations. The drawings are listed in the first illustration (Q6350-0). Some of the drawings have been reduced for inclusion in this report.

BICRON INSTRUMENT TESTER MA

DRAWING NUMBER	DRAWING TITLE
Q6350-000	MASTER DRAWING LIST
Q6350-001	DIGITAL CONTROLLER BOARD SCHEMATIC DIAGRAM
Q6350-002	ANALOG BOARD SCHEMATIC DIAGRAM
Q6350-003	DIGITAL CONTROLLER PRINTED CIRCUIT BOARD
Q6350-004	ANALOG PRINTED CIRCUIT BOARD
Q6352-005	HIGH VOLTAGE PRINTED CIRCUIT BOARD
Q6350-006	TEST HARNESS PRINTED CIRCUIT BOARDS
Q6350-007	TEST HARNESS HOUSING MECHANICAL DETAILS
Q6350-007	TEST HARNESS HOUSING DETAILS WITH 3-D VIEW
Q6350-008	HOUSING TOP MECHANICAL DETAILS
Q6350-009	BAR CODE READER LOGIC (GENERIC SERIAL UNIT) OBSOLETE
Q6350-010	HARNESS CABLING SCHEMATIC DIAGRAM
Q6350-011	EQUIPMENT AND RS-485 CONNECTIONS
Q6350-012	BOARD INTERCONNECTION SCHEMATIC DIAGRAM
Q6350-013	DIGITAL CONTROLLER BOARD FUNCTIONAL BLOCK DIAGRAM
Q6350-014	ANALOG BOARD FUNCTIONAL BLOCK DIAGRAM
Q6350-015	OPERATOR INTERFACE FUNTIONAL BLOCK DIAGRAM
Q6350-016	HIGH LEVEL BLOCK DIAGRAM

ASTER DRAWING LIST

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA ENERGY SYSTEMS, INC.

MASTER DRAWING LIST

BICRON INSTRUMENT TESTER

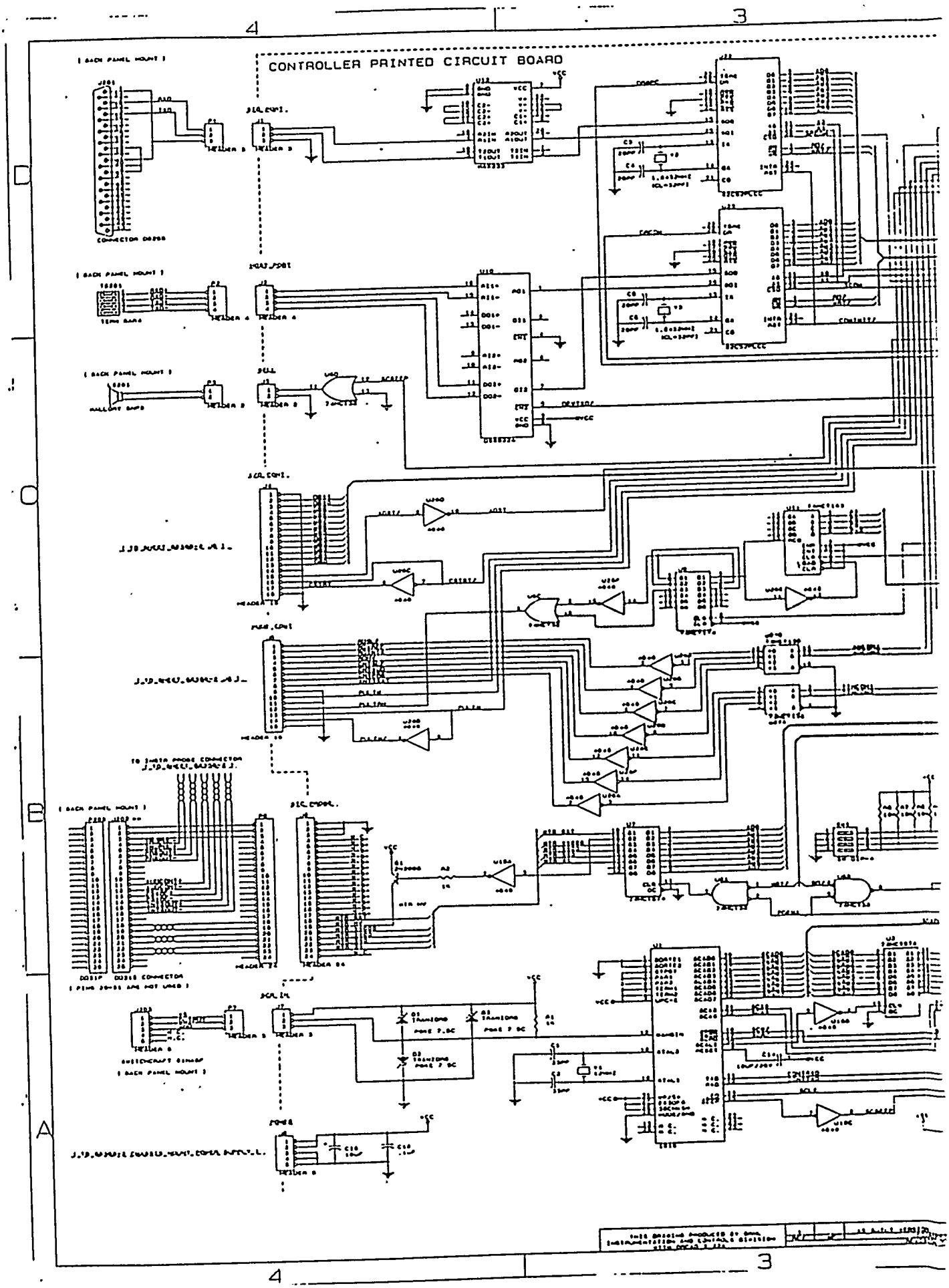
FILE DATE	FILE POINT	BLDG	SHT.	OF
	HC1 EC1	3500	1	1

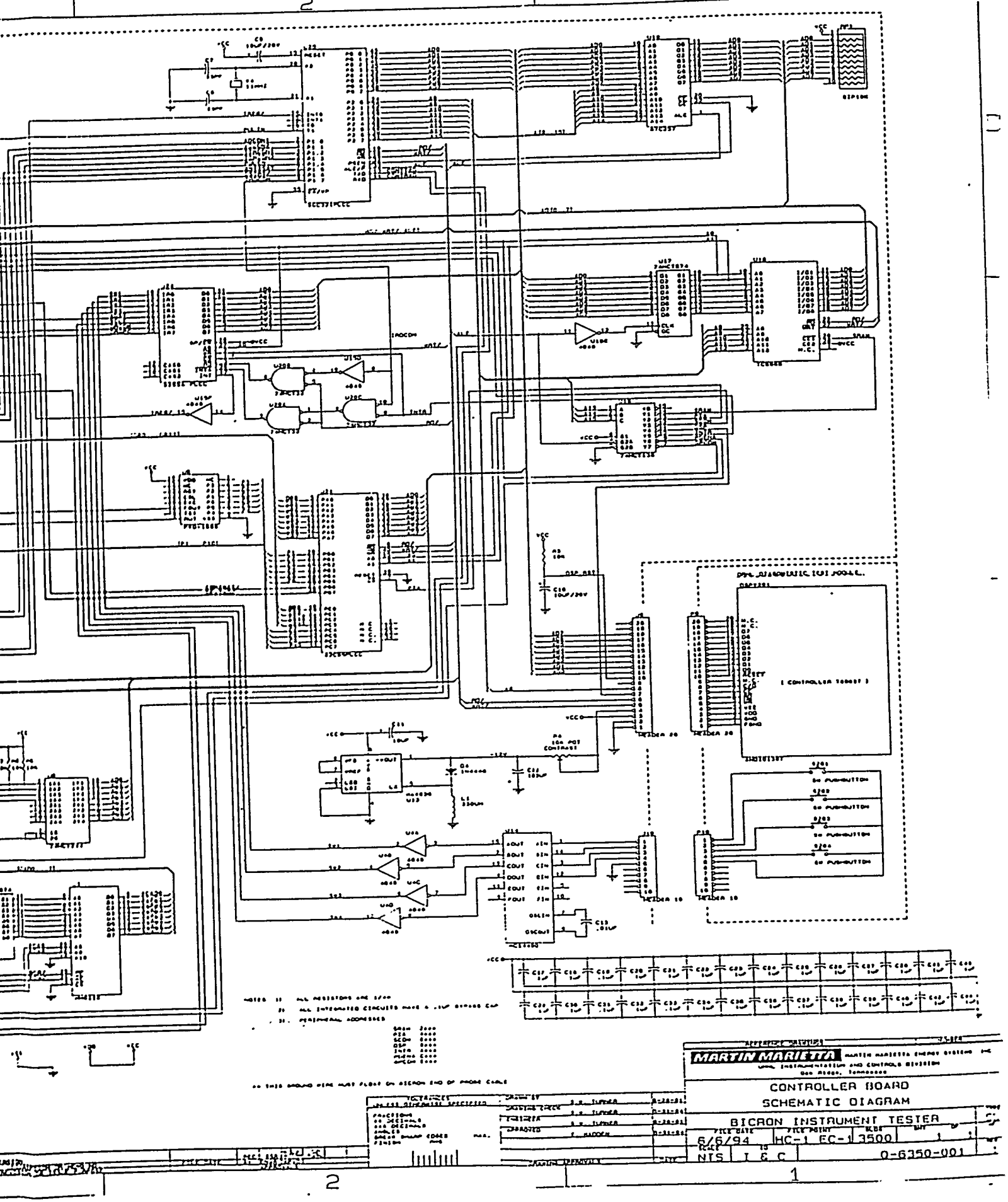
SCALE	ID	
N/A	I&C	06350-000

TYPE BD REV U

DRAWN BY	R.E. COOPER	7/95	
CHECKED		7/15	
APPROVED		10/15	
		12/15	
DRAWING APPROVALS		DATE	

FILE DATE	BY	CHKD	ENGR	APPR	APPR	APPR


CONTROLLER PRINTED CIRCUIT BOARD



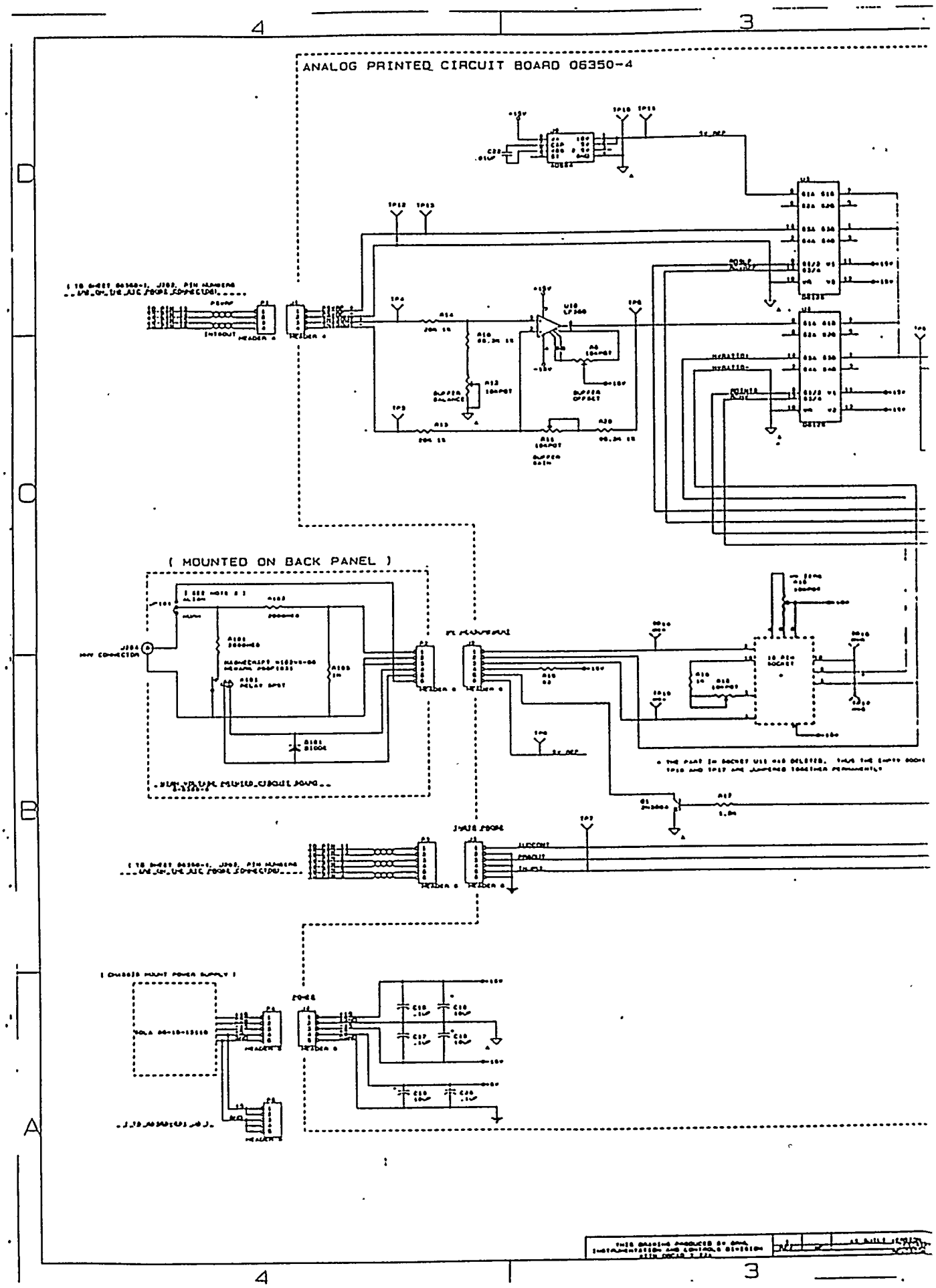
MARTIN MARIETTA

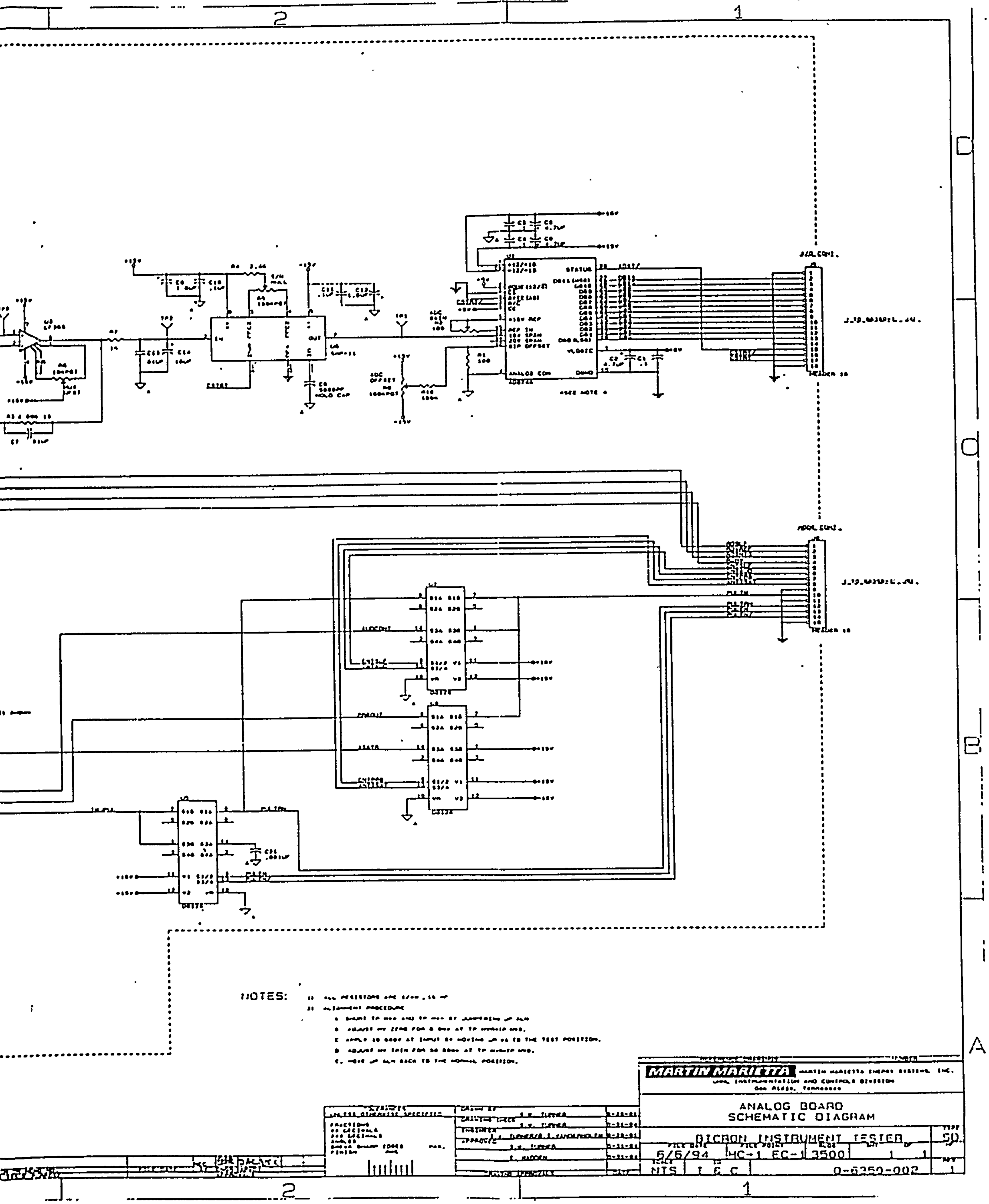
CONTROLLER BOARD
SCHEMATIC DIAGRAM

BICRON INSTRUMENT TESTER

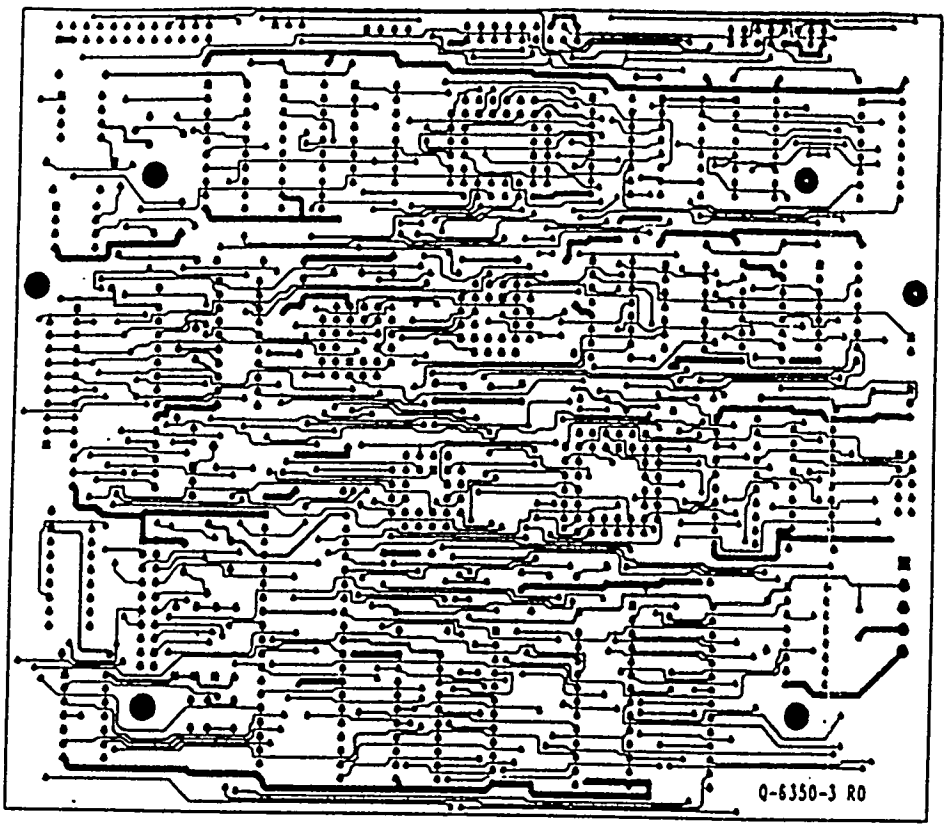
PRICE DATE	PRICE PRINT	SIZE	DWT
6/6/94	HC-1 EC-1 3500		
SCALE	IS		
NTS	I & C	0-6350-001	

2 1


ANALOG PRINTED CIRCUIT BOARD 06350-4
(MOUNTED ON BACK PANEL)



NOTES:
MARTIN MARIETTA
ANALOG BOARD
SCHEMATIC DIAGRAM
DICRON INSTRUMENT TESTER
6/6/94
HC-1 EC-1
3500
NTS
I & C
0-6350-002
SD.

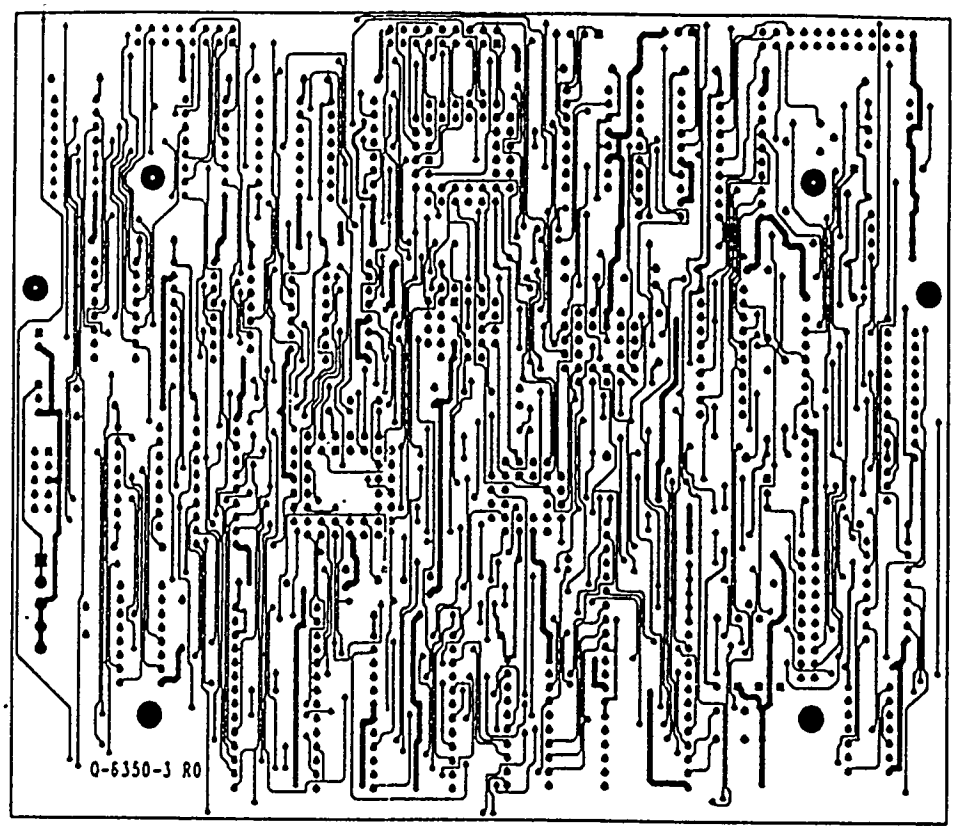


FRONT

REV.	HF	DESCRIPTION	FILE DATE	BY	ENGR	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			

UNLESS OTHERWISE

FRACTIONS :
XX DECIMALS :
XXX DECIMALS :
ANGLES :
BREAK SHARP ED



REAR

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE: MAR 21 '91
INITIAL: SS

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operating for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

CONTROLLER
PRINTED CIRCUIT BOARD

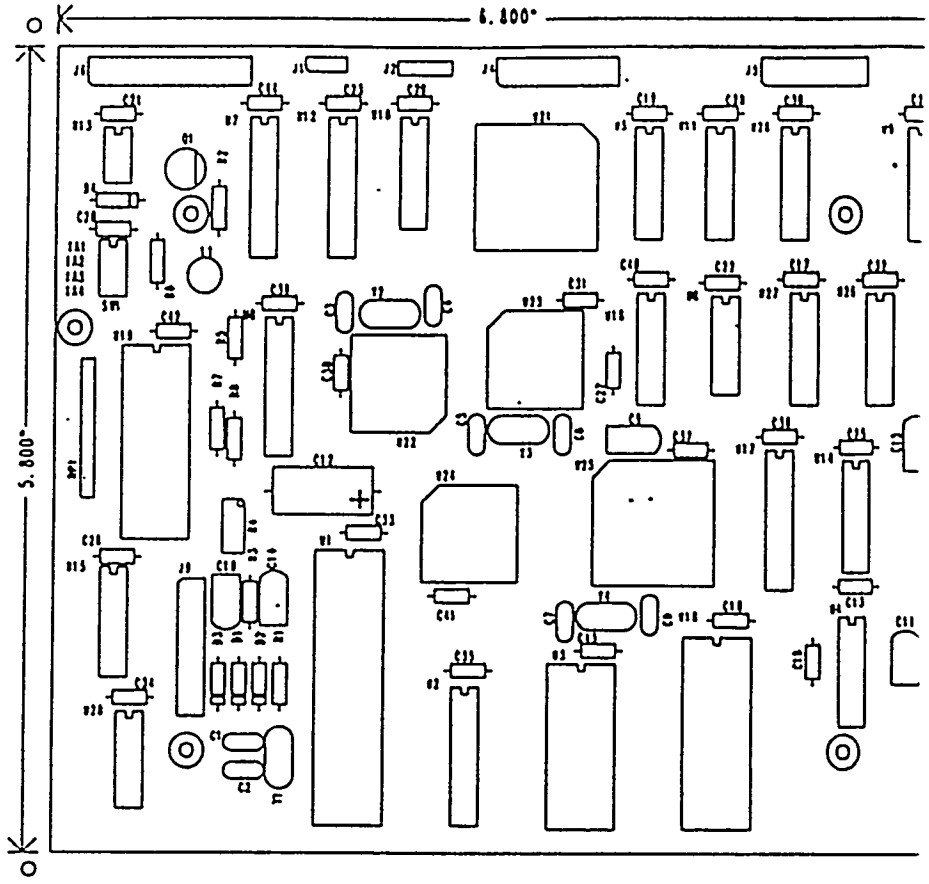
BICRON INSTRUMENT TESTER — TYPE: PC

FILE DATE	FILE POINT	BLOCK	SHT.	OF	REV
9-25-91	HC-1 EC-1	3500	1	2	..

SCALE	ID		REV
1 : 1	I&C	Q-6350-3	0

DRAWN BY		3-91
ENGINEER	G. W. Turner	9-24-91
APPROVED	E. Maddox	9-24-91
DRAWING APPROVALS		DATE

TOLERANCES UNLESS OTHERWISE SPECIFIED
FRACTIONS ± 1/64"
DECIMALS ± .010"
DECIMALS ± .005"
ANGLES ± 0° 30'
BREAK SHARP EDGES



SILK SCREEN

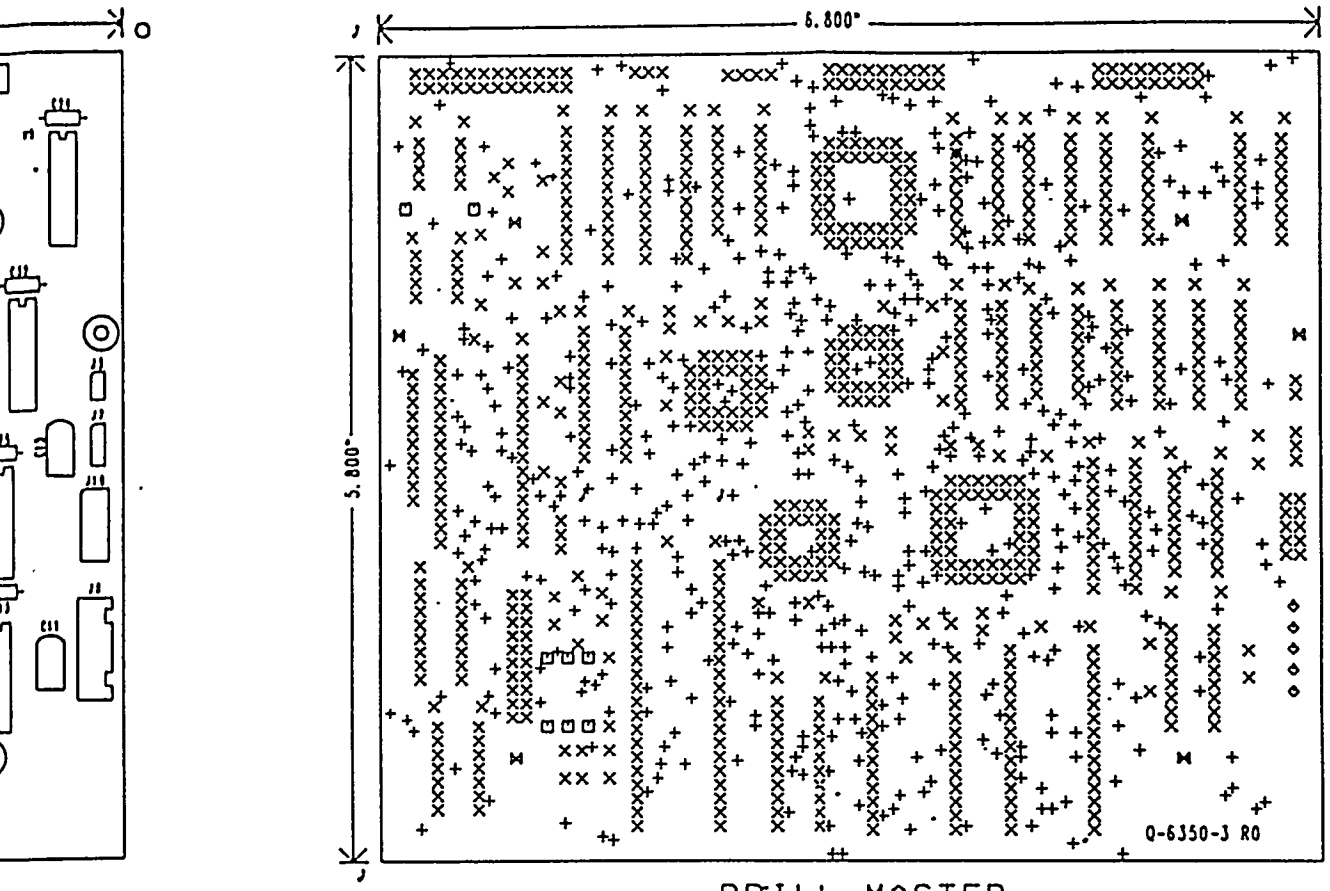
NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/1-B-2-B.
2. ALL HOLES WITH LANDS ON BOTH SIDES OF THE BOARD SHALL BE 100% PLATED THRU. COPPER PLATING INSIDE THE HOLES SHALL BE .001" MINIMUM IN ACCORDANCE WITH MIL-C-14550. FRONT TO REAR REGISTRATION TO BE WITHIN .005".
3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728. THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.
4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.
5. DO NOT USE THIS DRAWING AS A WORK SHEET. MASTER POSITIVE(S) OF WIRING WILL BE SUPPLIED BY ORNL.
6. LIMIT ON ALL DIMENSIONS .015 IN UNLESS OTHERWISE NOTED.
7. HOLE LOCATION TOLERANCE .003 IN.
8. DIMENSIONS OF ALL HOLES NOTED BELOW ARE FINISHED HOLE DIAMETERS AFTER PLATING THRU.

UNLES

FRACTI
XX DEC
XXX DE
ANGLES
BACAR

REV.	MF	DESCRIPTION	FILE DATE	BY	ENGR	APVD
		REVISION OR ISSUE PURPOSE	APPROVALS			



DRILL MASTER

SIZE	QTY	STN
28	498	+
35	838	X
40	8	□
52	5	◇
55	2	Y
100	3	T
125	1	⋈

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE:
INITIAL: MAR 21 '91

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

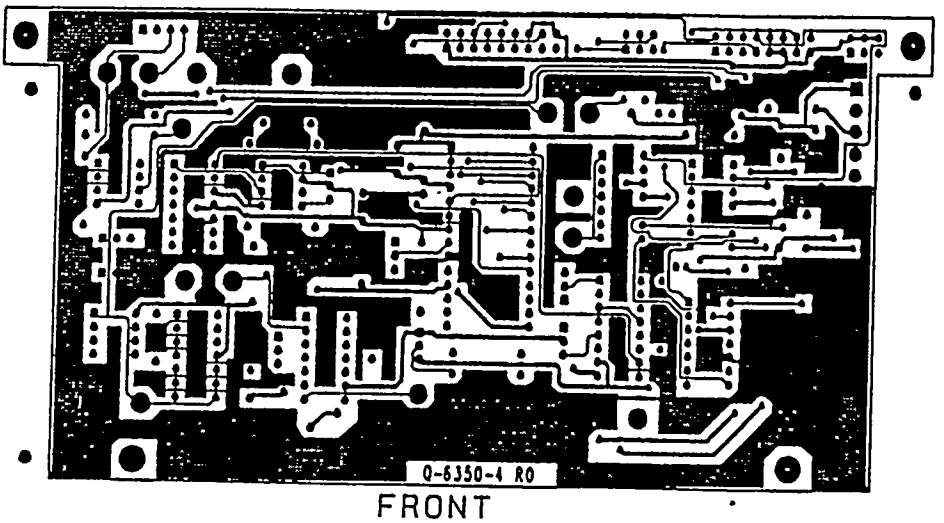
CONTROLLER
PRINTED CIRCUIT BOARD

BICRON INSTRUMENT TESTER — TYPE: PC

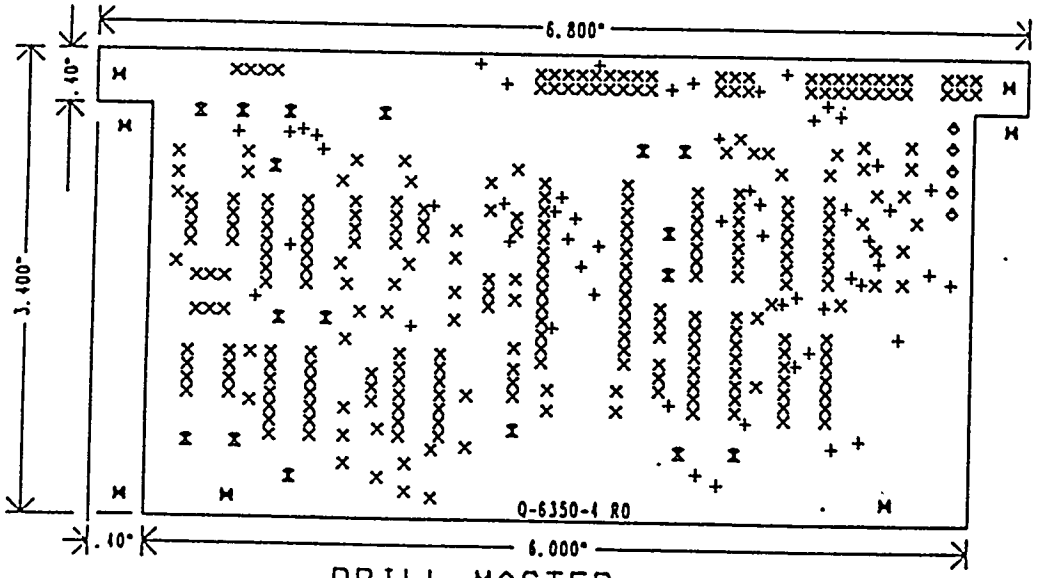
FILE DATE	FILE PRINT	SHEET	SHT. OF	REV
9-25-91	HC-1 EC-1	3500	2 2	A

SCALE	ID		REV
1 : 1	I&C	Q-6350-3	A

TOLERANCES UNLESS OTHERWISE SPECIFIED			DRAWN BY	A.A. MAPLES	3-91
FRACTIONS	±	1/64"	DRAWING CHECK		
xx DECIMALS	±	.010"	ENGINEER	H. W. Turner	9-24-91
xxx DECIMALS	±	.005"	APPROVED	E. Madden	9-24-91
ANGLES	±	0° 30'			
BREAK SHARP EDGES					
			DRAWING APPROVALS		DATE



FRONT

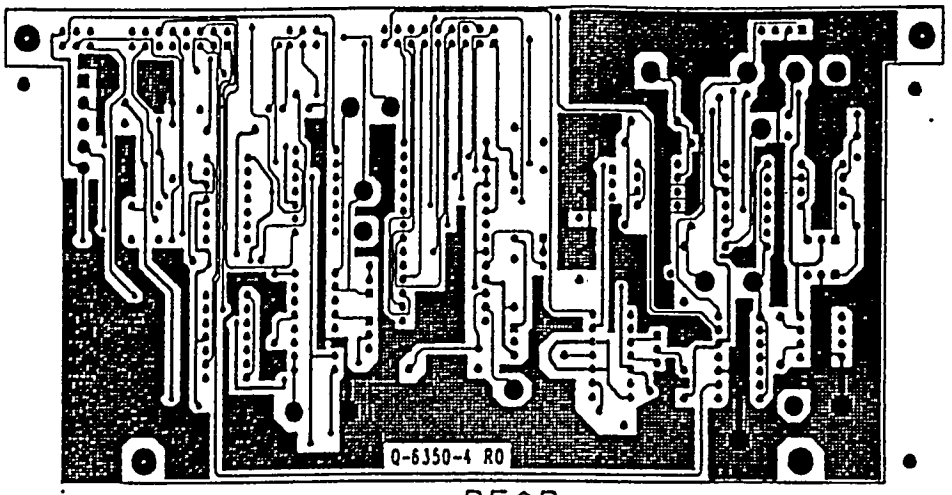


DRILL MASTER

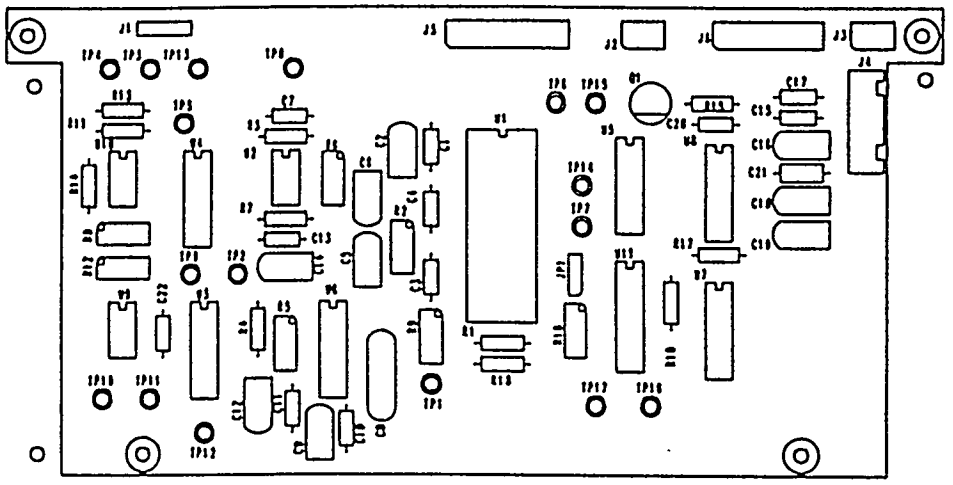
NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/1-8-2-8.
2. ALL HOLES WITH LANDS ON BOTH SIDES OF THE BOARD SHALL BE 100% PLATED THRU. COPPER PLATING INSIDE THE HOLES SHALL BE .001" MINIMUM IN ACCORDANCE WITH MIL-C-[illegible]50. PRINT TO DRILL REGISTRATION TO BE WITHIN .005".
3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728. THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.
4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.
5. DO NOT USE THIS DRAWING AS A WORK SHEET. OF VIAINS WILL BE SUPPLIED BY CAML.
6. LIMIT ON ALL DIMENSIONS .010 IN UNLESS [illegible]
7. HOLE LOCATION TOLERANCE .003 IN.
8. DIMENSIONS OF ALL HOLES AFTER SOLDER AND [illegible] PLATING THRU.

REV.		DESCRIPTION	FILE DATE	BY	ENGR	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			



REAR



SILK SCREEN

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE:
INITIAL: MAR 21 '91 S.S

SIZE	DIM	SYM
25	50	+
35	315	×
50	.5	◇
80	.8	⧗
.125	1	⋈

S-CEET, MASTER POSITIVE(S)
...
UNLESS DIMENSIONS NOTED.

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA ENERGY SYSTEMS, INC.

operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee, Paducah, Kentucky

ANALOG PRINTED CIRCUIT BOARD

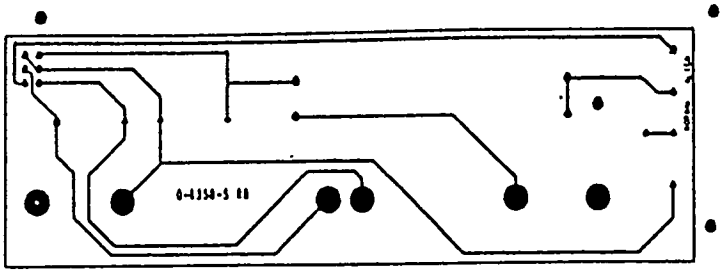
BICRON INSTRUMENT TESTER

TYPE: PC

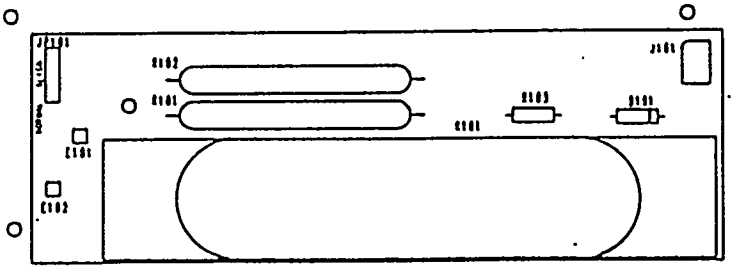
TOLERANCES & DIMENSIONS SPECIFIED	DRAWN BY: R.A. MAPLES	5-91
	DRAWING APPROVALS	DATE

FILE DATE	FILE POINT	BLOCK	SHT. OF	REV
9-25-91	HC-1 EC-1	3500	1 1	

SCALE	ID		REV
1 : 1	I&C	Q-6350-4	Ø



REAR



SILK SCREEN

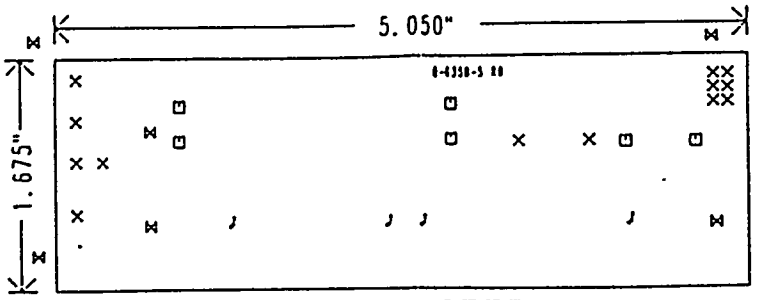
NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/0-0-2-0.
2. THIS BOARD SHALL HAVE CIRCUIT ON THE SIDE WITH A SILK SCREEN ON THE REAR.
3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728. THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.
4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.
5. DO NOT USE THIS DRAWING IF WIRING WILL BE SUPPLI
6. LIMIT ON ALL DIMENSIONS
7. HOLE LOCATION TOLERANCE
8. DIMENSIONS OF ALL HOLES

UNLESS OT

DIMENSIONS ON DECIMAL DIMENSIONS ANGLES DECAL SHEE

REV.	MF	DESCRIPTION	FILE DATE	BY	CHKD	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			



DRILL MASTER

SIZE	QTY	SYM
35	13	X
40	6	⊡
93	2	Z
125	6	⋈
136	2	Y

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE: MAR 21 '91
INITIAL:

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

HIGH VOLTAGE PRINTED CIRCUIT BOARD

BICRON INSTRUMENT TESTER — TYPE: PC

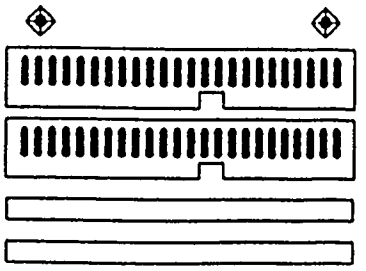
FILE DATE	FILE POINT	BLDG	SHT.	OF	REV
9-25-91	HC-1 EC-1	3500	1	1	

SCALE	TO		REV
1 : 1	I&C	Q-6350-5	0

ARING AS A WORK SHEET. MASTER POSITIVE(S)
SUPPLIED BY DANL.
DECIMALS .010 IN UNLESS OTHERWISE NOTED.
ANGLES .005 IN.
UNLESS NOTED OTHER WISE ALL DIMENSIONS.

TOLERANCES
UNLESS OTHERWISE SPECIFIED
FRACTIONS ± 1/64"
DECIMALS ± .010"
DECIMALS ± .005"
ANGLES ± 0° 30'
ALL SHARP EDGES

DRAWN BY	DATE
A.A. MAPLES	3-91
D.W. Turner	9-27-91
E. Mahler	9-27-91
DRAWING APPROVALS	DATE



CIRCUIT

. NOTES:

1. MATERIAL TO BE .0625 I
   FR4, PER MIL-P-13949/4.
   FL-GFP-0625-C1/8-B-2-C
2. REGISTRATION TO BE WIT
3. CLEAN AND SOLDER PLATE
   MIL-P-81728.THE FUSED T
   THICK MINIMUM.
4. AFTER PROCESSING. ETCHE
   AND SCRATCHES.

	HOLE SIZE	
+	#48	.048 IN
●	#51	.067 IN
::	1/8	.125 IN

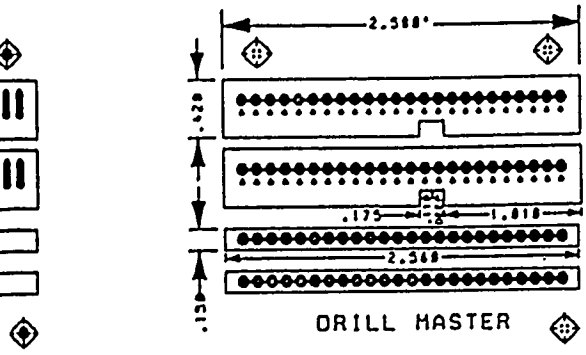
NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/8-B-2-B.
2. THIS BOARD ONLY HAS CIRCUIT ON THE BACK.
3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728.THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.
4. AFTER PROCESSING. ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.

5. DO NOT USE THIS DRAWIN
   OF VIAING WILL BE SUPPL
6. LIMIT ON ALL DIMENSION
7. HOLE LOCATION TOLERANCE
8. DIMENSIONS OF ALL HOLES

UNLESS
FRACTION
TO DECIM
THREE DECI
ANGLES
BREAK SH

REV.	MF	DESCRIPTION	FILE DATE	BY	ENGR	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			



.0625 IN THICK EPOXY FIBERGLASS TYPE
13949/4. SPECIFIC DESIGNATION SHALL BE
0-8-2-C.

BE WITHIN .003".

R PLATE ALL COPPER IN ACCORDANCE WITH
FUSED TIN-LEAD SHALL BE .0003 INCH

8. ETCHED BOARDS TO BE FREE OF VOIDS

LE SIZE	QUAN.
.040 IN DIAMETER	40
.067 IN DIAMETER	96
.125 IN DIAMETER	3

IS DRAWING AS A WORK SHEET. MASTER POSITIVE(S)
L BE SUPPLIED BY DANL.
DIMENSIONS .010 IN UNLESS OTHERWISE NOTED.
TOLERANCE .003 IN.
ALL HOLES NOTED BELOW ARE FINISHED HOLE DIAMETERS.

TOLERANCES UNLESS OTHERWISE SPECIFIED	Drawn By A.A. MAPLES	3-91
FRACTIONS ± 1/64"	Machine Data	
00 DECIMALS ± .010"	Checked	
000 DECIMALS ± .003"	Approved	
ANGLES ± 0° 30'		
BREAK SHARP EDGES		
	DRAWING APPROVALS	DATE

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

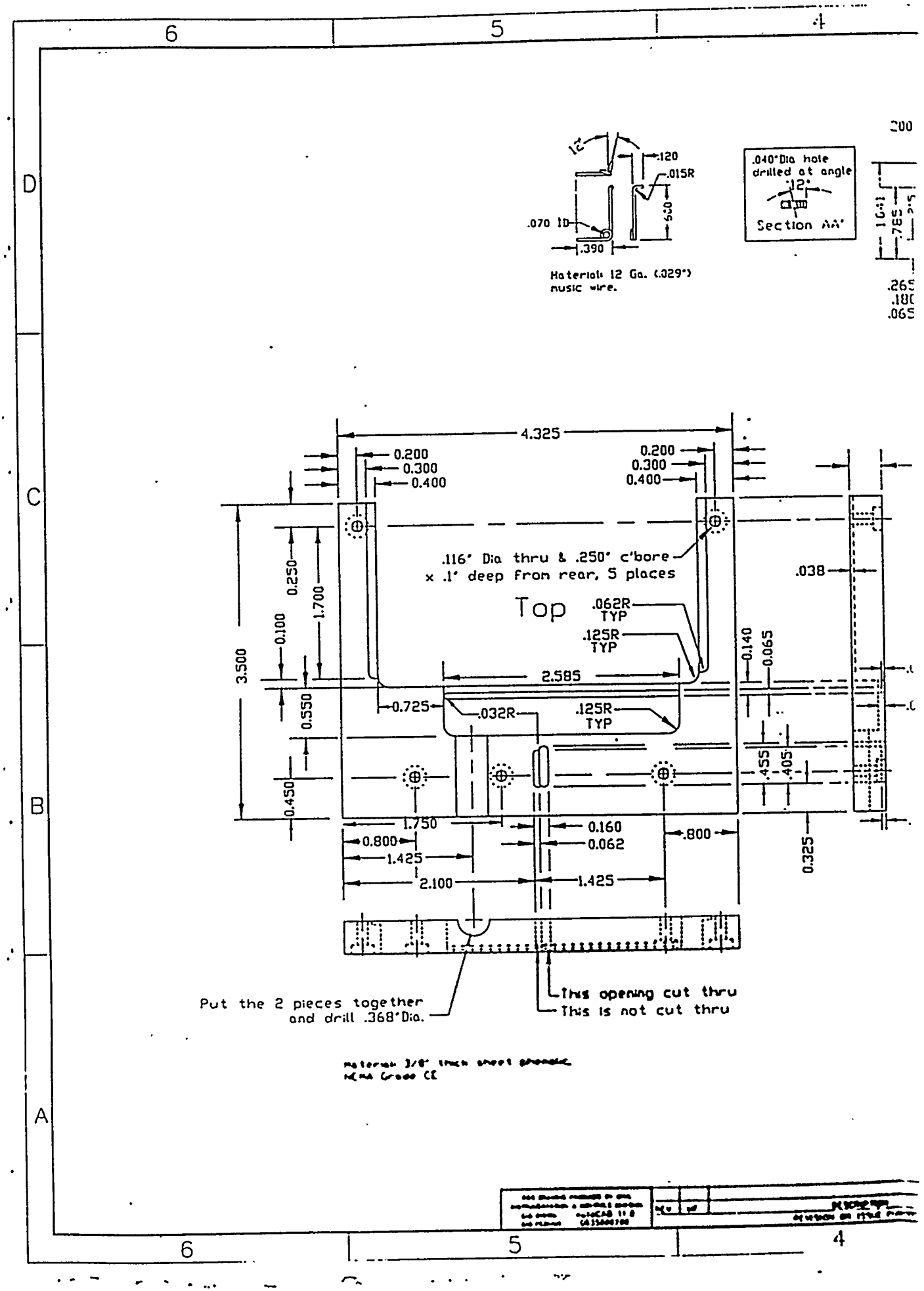
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

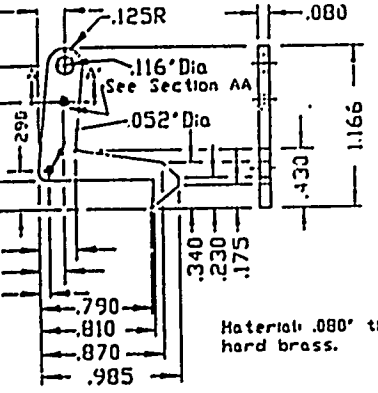
PROBE
PRINTED CIRCUIT BOARDS

BICRON INSTRUMENT TESTER — TYPE: PC

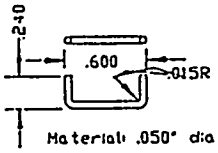
FILE DATE	FILE POINT	BLOCK	SHT.	OF	REV
	HC-1 EC-1	3500	1	1	

SCALE	IS		REV
1 : 1	I&C	Q-6350-6	Ø

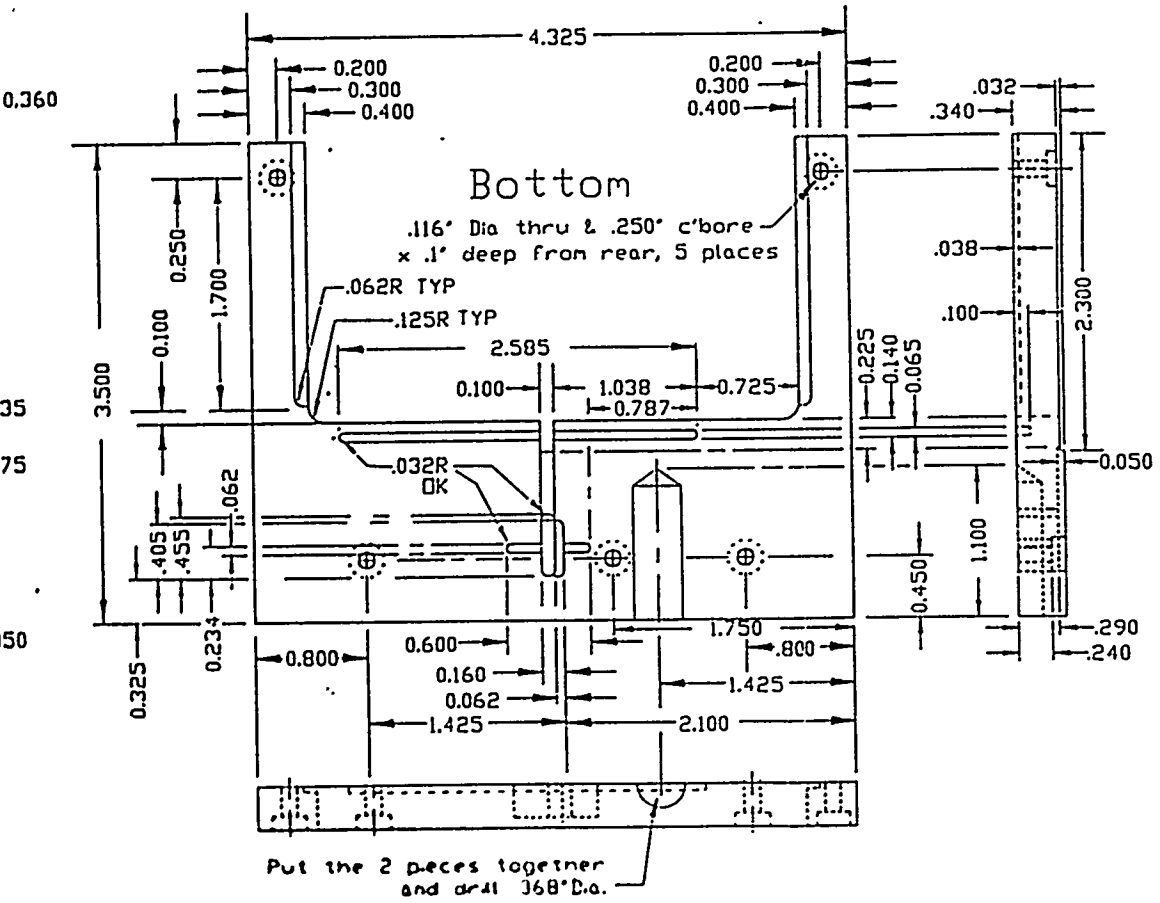

6
5
4
D
C
B
A
12°
.120
.015R
.070 ID
.390
.040°Dia hole drilled at angle
12°
Section AA'
Material 12 Ga. (.029") music wire.
4.325
0.200
0.300
0.400
.116° Dia thru & .250° c'bore x .1° deep from rear, 5 places
Top
.062R TYP
.125R TYP
.038
0.250
1.700
0.100
3.500
0.550
0.450
2.585
0.725
.032R
.125R TYP
0.140
0.065
.455
.405
1.750
0.800
1.425
2.100
0.160
0.062
.800
1.425
0.325
Put the 2 pieces together and drill .368°Dia.
This opening cut thru
This is not cut thru



Material .080" thick half hard brass.



Material .050" dia nickel plated steel. Use a large paper clip.



Put the 2 pieces together and drill .368" Dia.

Material 3/8" thick sheet phenolic, CHA Grade CE

REFERENCE DRAWINGS	NAMES

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC

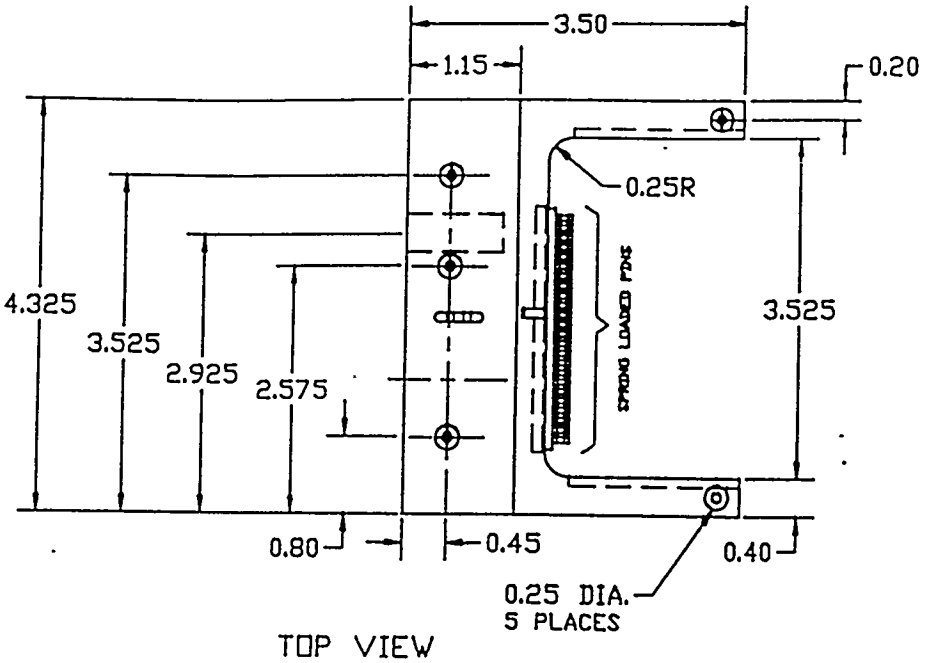
PROBE HOUSING MECHANICAL DETAILS

BICRON INSTRUMENT TESTER

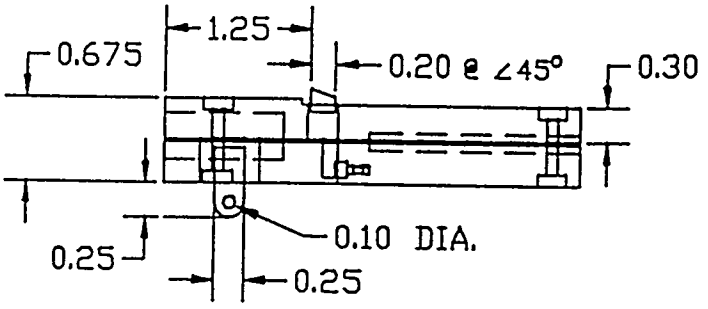
TOLERANCES UNLESS OTHERWISE SPECIFIED			
	R. A. MAPLES	3-91	
		3-19-92	
		3-92	
DRAWING APPROVALS		DATE	

FILE DATE	FILE POINT	BLDG	SHT	OF	TYPE
MAR 1 5 1992	HC-1 EC-1	3500	1	1	MD

SCALE	JO		REV
	1 & C	0-6350-7	0

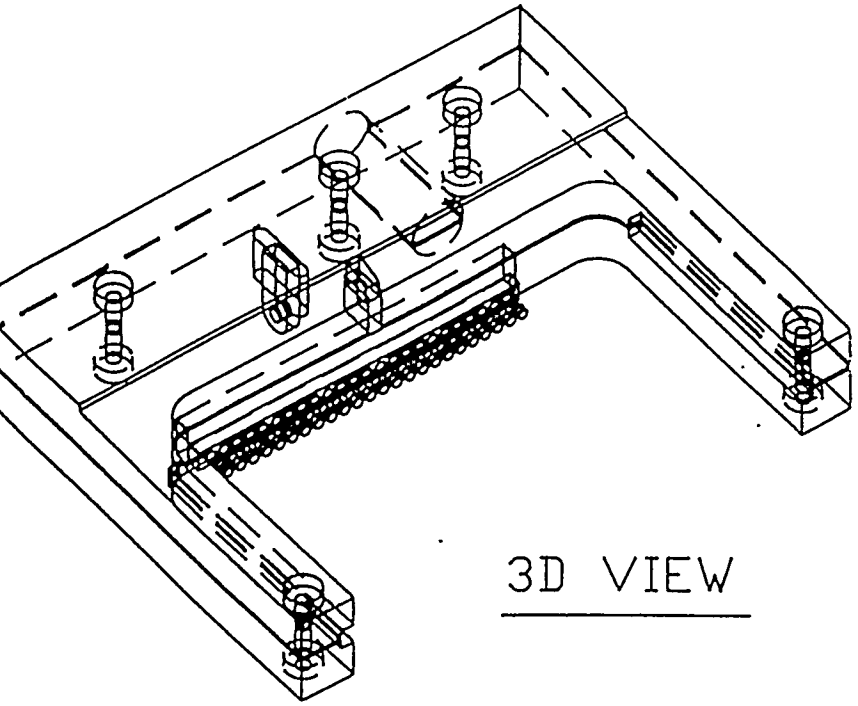


TOP VIEW

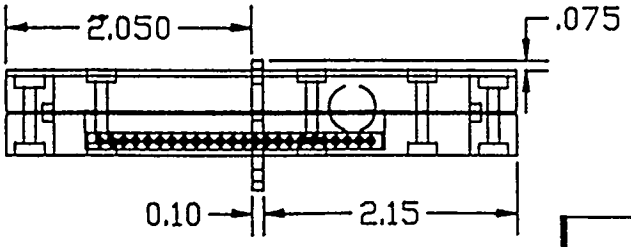


END VIEW

REV.	MF		FILE DATE	BY	ENGR	APVD		
		DESCRIPTION OR MEMO NO.	APPROVALS					



3D VIEW



FRONT VIEW

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

TEST HARNESS HOUSING MECHANICAL DETAILS

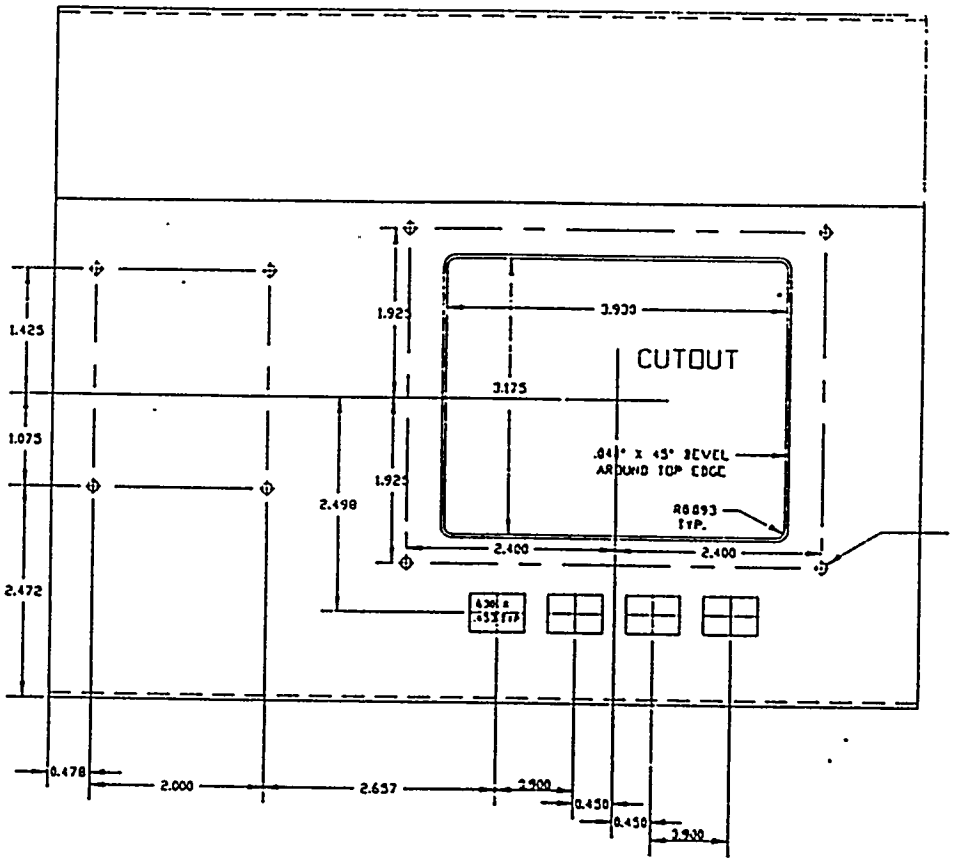
BICRON INSTRUMENT TESTER

TYPE: MD

DRAWN BY	DATE
D.E COOPER	9/95
CHECKED	9/95
APPROVED	9/95
D. a. Clayton	9/95
RT Rosebury	9/95
DRAWING APPROVALS	DATE

FILE DATE	FILE POINT	BLDG	SHT. OF	MF
	HC- EC-		2 2	

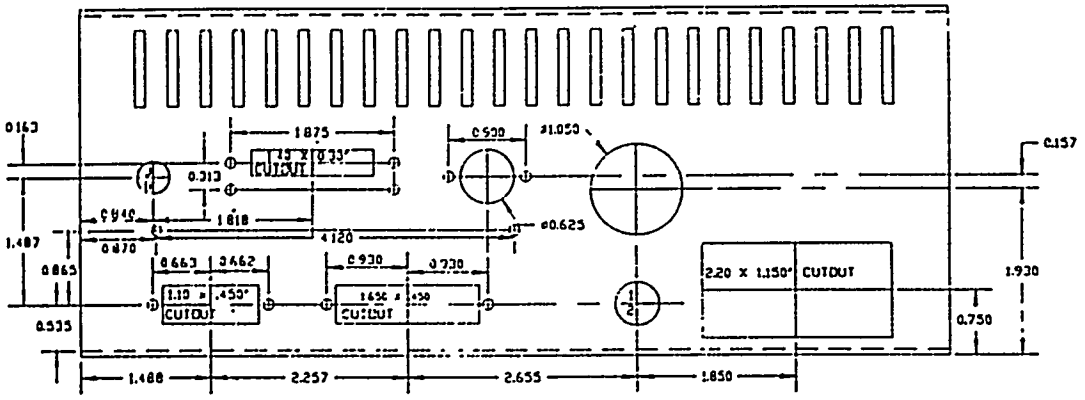
SCALE	ID		REV
	I&C	Q-6350-007	0



CABINET TOP VIEW

NOTES:
1. ENCLU
4.0" HI
2. ENCLU
4.0" HI

REV.	DESCRIPTION
	REVISION PER ISSUE PACKET



CABINET REAR VIEW

...SURE TOP IS HAMMOND #1456KG4 PWH: 10" WIDE, 7.2" DEEP
...IGH.
...SURE CASE IS HAMMOND #1456KG4CWW, 10" WIDE, 7.2" DEEP,
...IGH.

TOLERANCES UNLESS OTHERWISE SPECIFIED		DRAWN R. A. MAPLES	3-91
FRACTIONS	± 1/64"	CHECKED	
XX DECIMALS	± .015"	ENGR APPVD	3-19-92
XXX DECIMALS	± .005"	MFG APPVD	3-92
ANGLES	± 0° 30'		
BREAK SHARP EDGES			

	FILE DATE	BY	ENGR APVD			
			MFG APPVD			

DRAWING APPROVALS — DATE

REFERENCE DRAWINGS — NUMBER

MARTIN MARIETTA — MARTIN MARIETTA ENERGY SYSTEMS, INC.

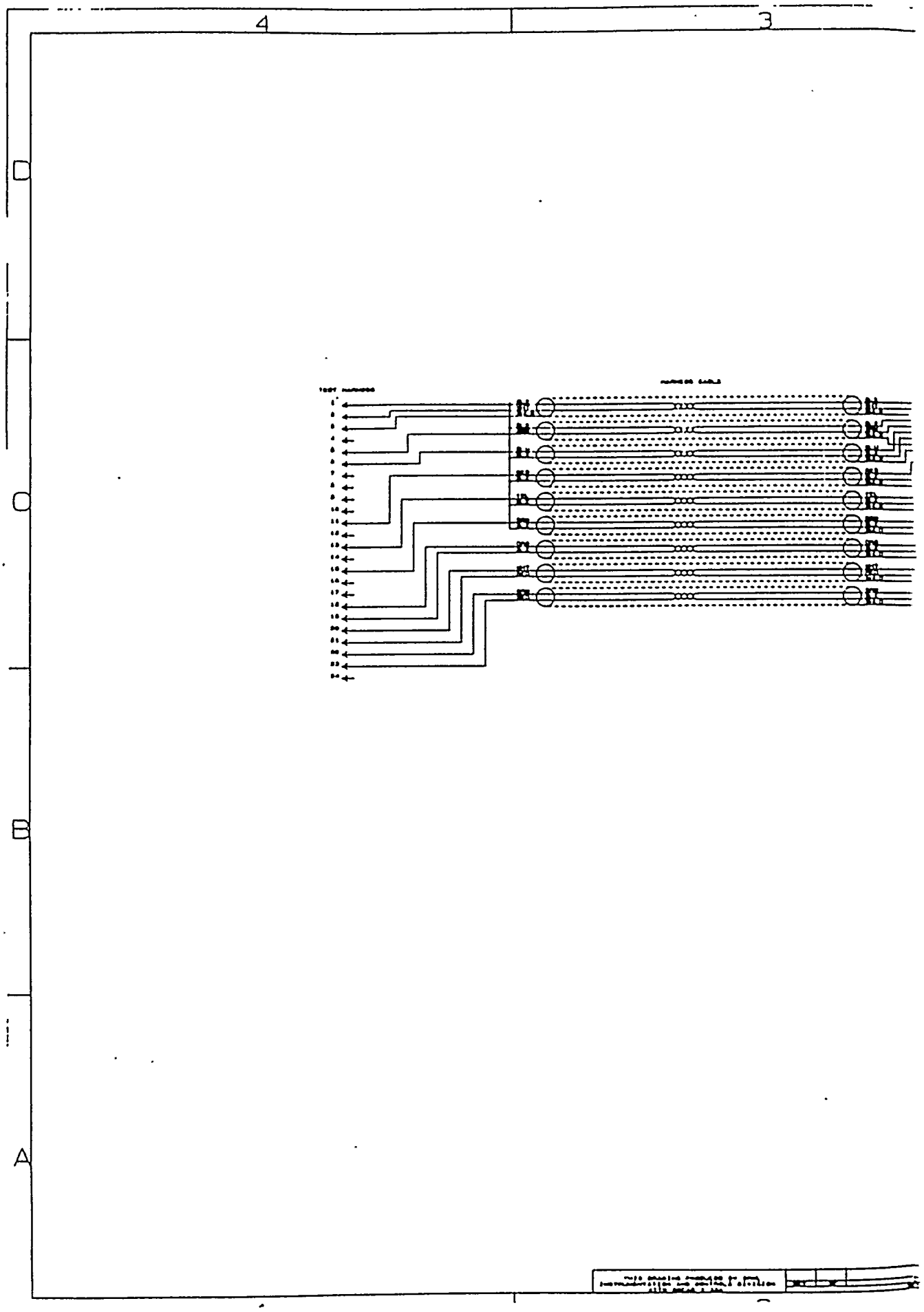
operated for the Department of Energy under U.S. Government contract at its facilities in Oak Ridge, Tennessee, Paducah, Kentucky, and Portsmouth, Ohio

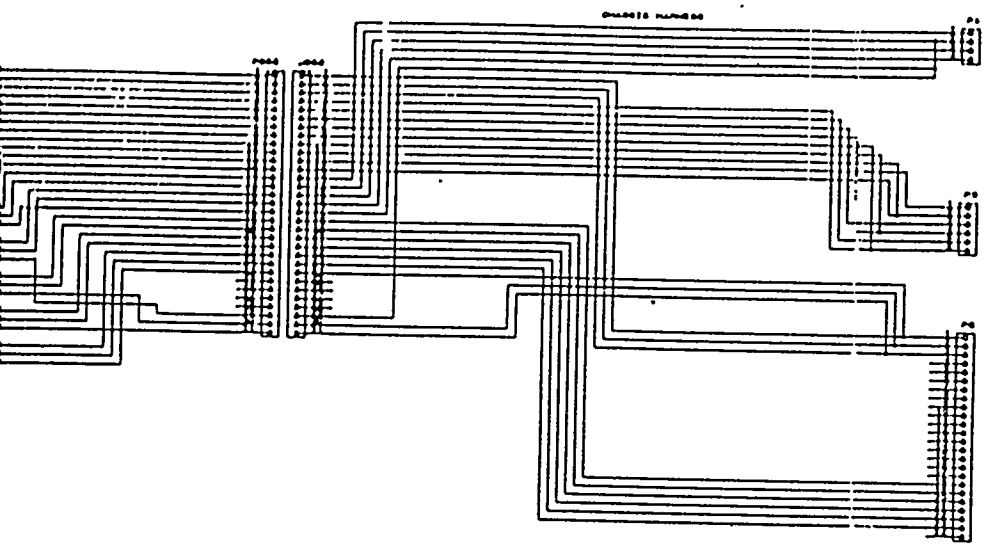
HOUSING TOP MECHANICAL DETAILS

BICRON INSTRUMENT TESTER

FILE DATE	FILE POINT	SHEET	OF	REV
MAR 16 bxc	HC-1 EC-1	1	1	

SCALE	ID	DRAWING NO.	REV
FULL	I & C	Q-6350-8	0


4
3
D
C
B
A

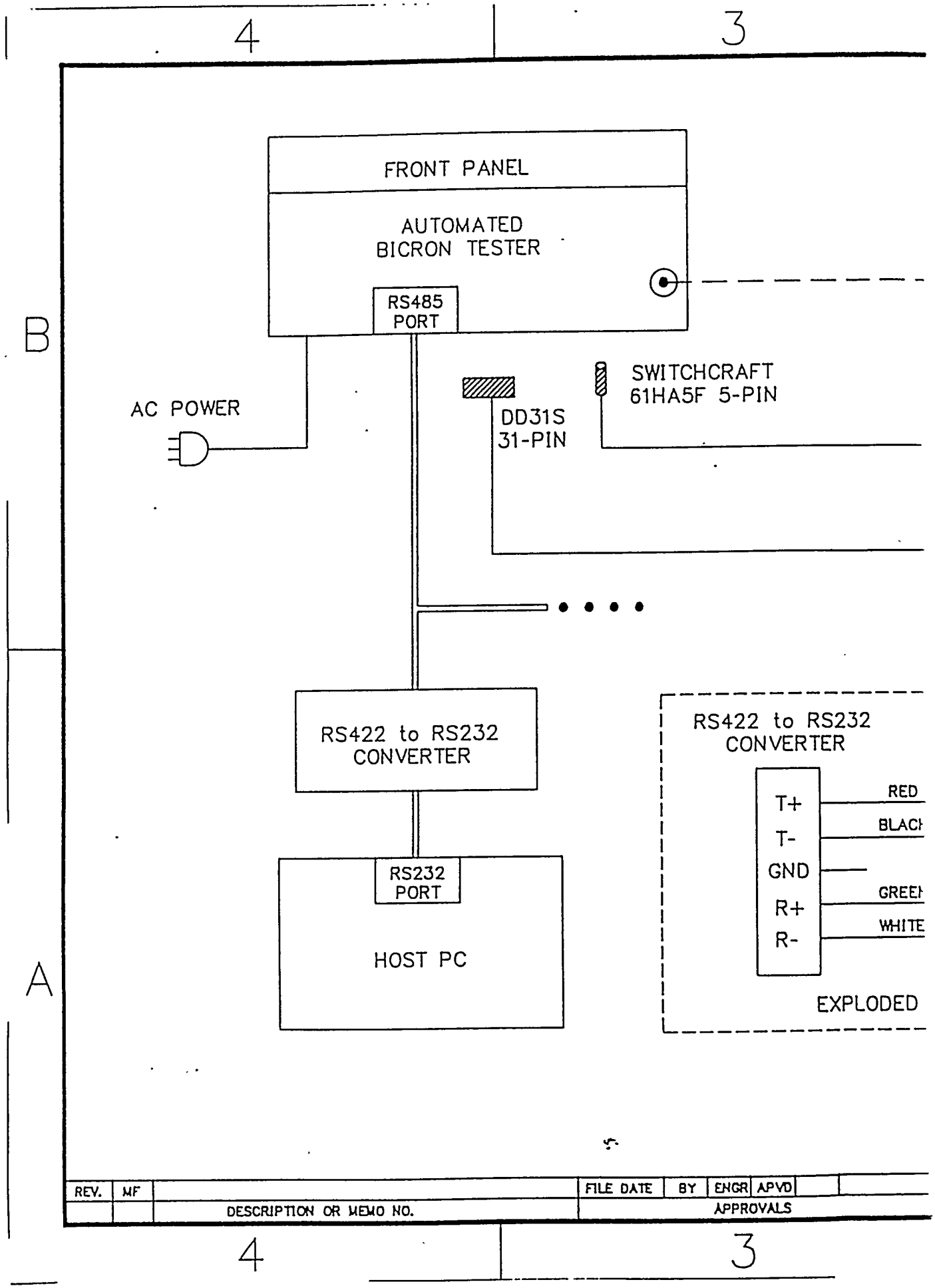


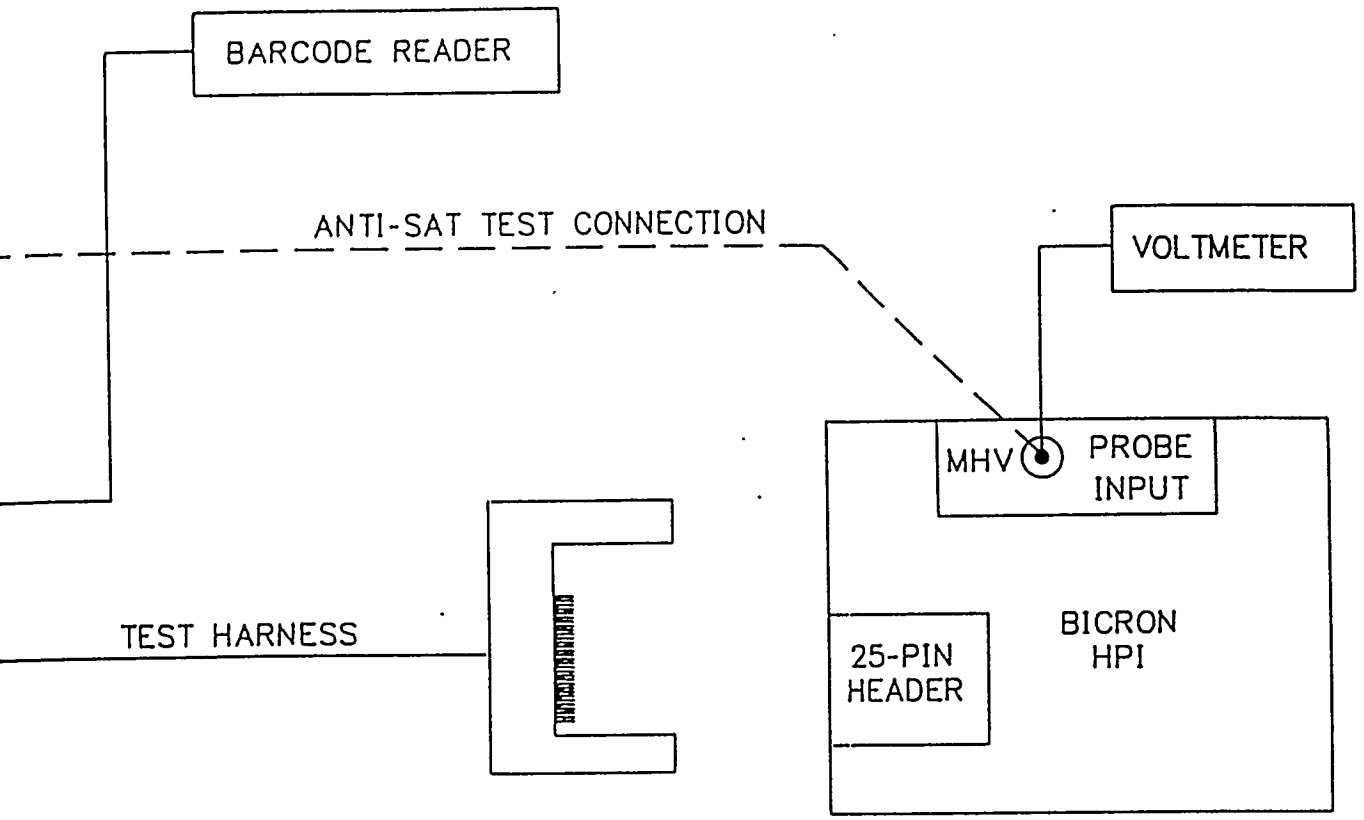
MARTIN MARIETTA

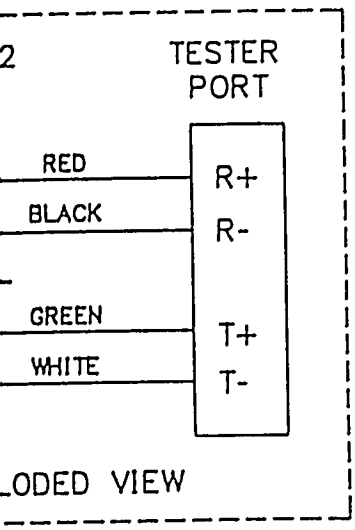
HARNESS CABLING
SCHEMATIC DIAGRAM

BICRON INSTRUMENT TESTER

		HC-1	EC-1	3500		
NTS	I E C				0-6350-010	0

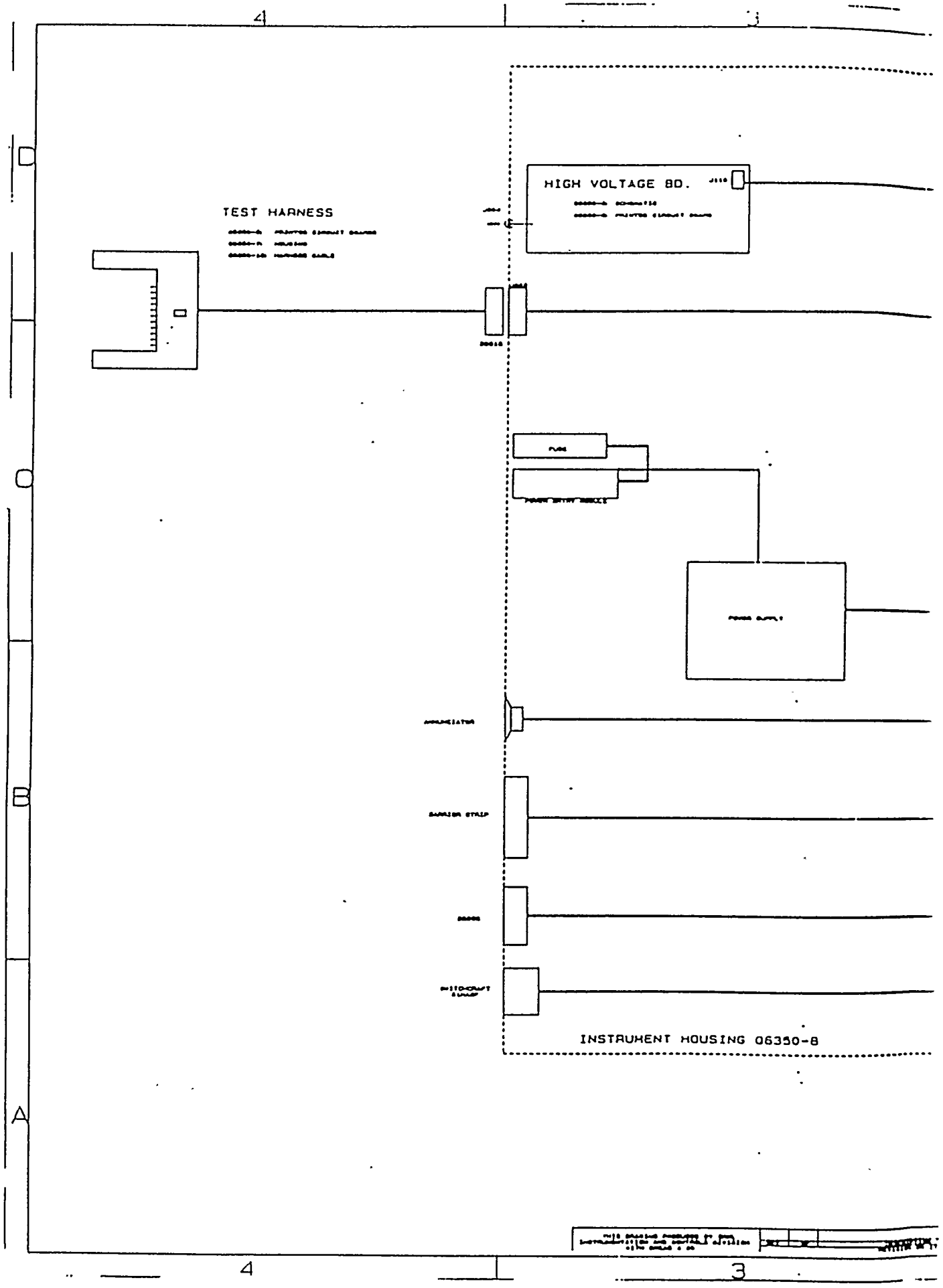

4
3
B
A
FRONT PANEL
AUTOMATED
BICRON TESTER
RS485
PORT
AC POWER
DD31S
31-PIN
SWITCHCRAFT
61HA5F 5-PIN
RS422 to RS232
CONVERTER
RS232
PORT
HOST PC
RS422 to RS232
CONVERTER
T+
T-
GND
R+
R-
RED
BLACK
GREEN
WHITE
EXPLODED
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3

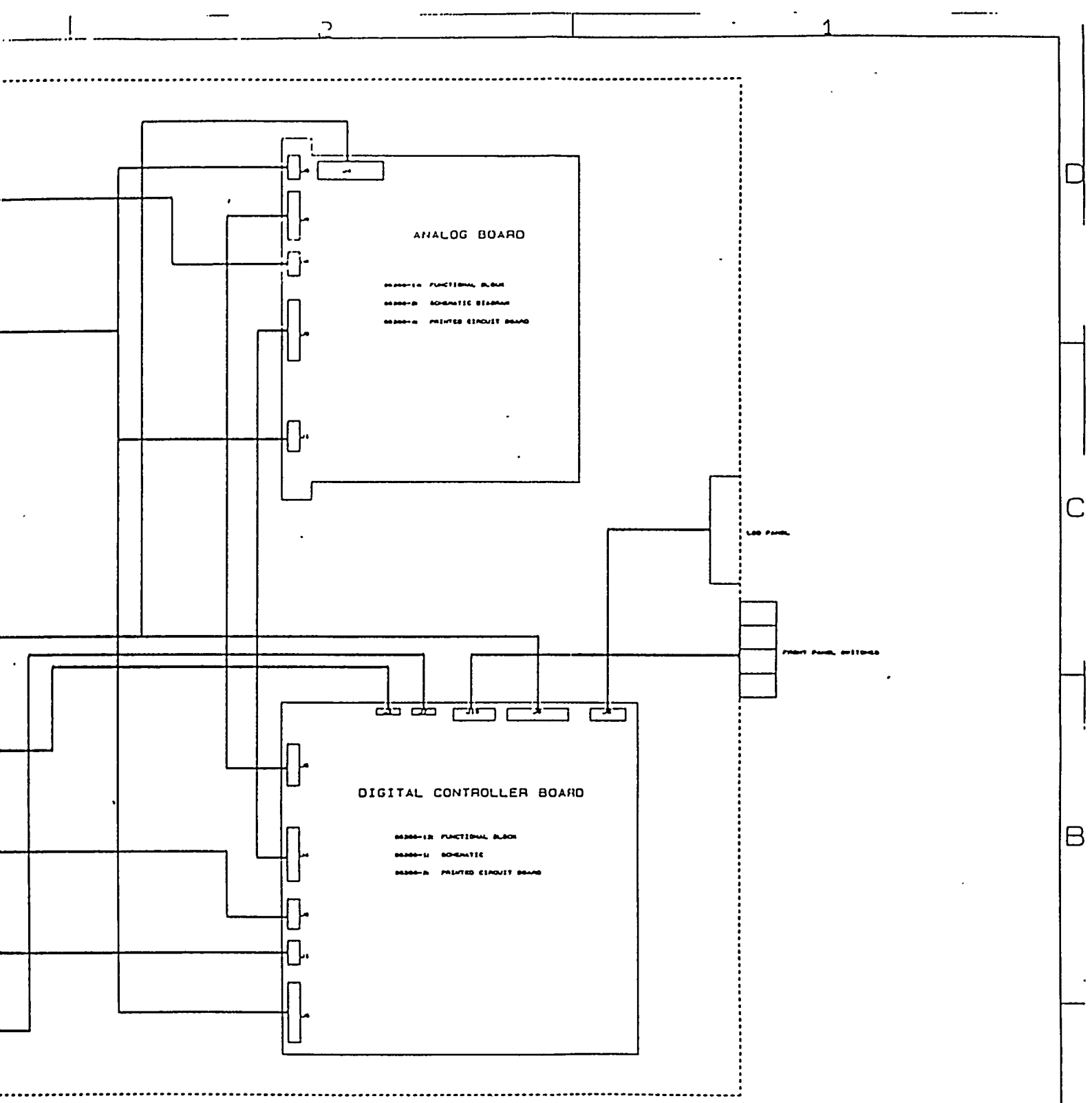




DRAWN BY	R.E. COOPER	7/95
DESIGNER CHECK		9/95
ENGINEER		9/95
APPROVED	S. A. Clayton	9/95
	R. T. Roseberry	9/95
DRAWING APPROVALS		DATE

REFERENCE DRAWINGS			NUMBER	
MARTIN MARIETTA ENERGY SYSTEMS, INC. operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400 Oak Ridge, Tennessee Paducah, Kentucky				
EQUIPMENT & RS-485 CONNECTION DIAGRAM				
BICRON INSTRUMENT TESTER				TYPE BD
FILE DATE		FILE POINT HC- EC-	BLDG	SHT. 1 OF 1
SCALE	ID I&C	Q-6350-011		REV Ø


4
3
D
C
B
A
HIGH VOLTAGE BD.
TEST HARNESS
INSTRUMENT HOUSING 06350-8

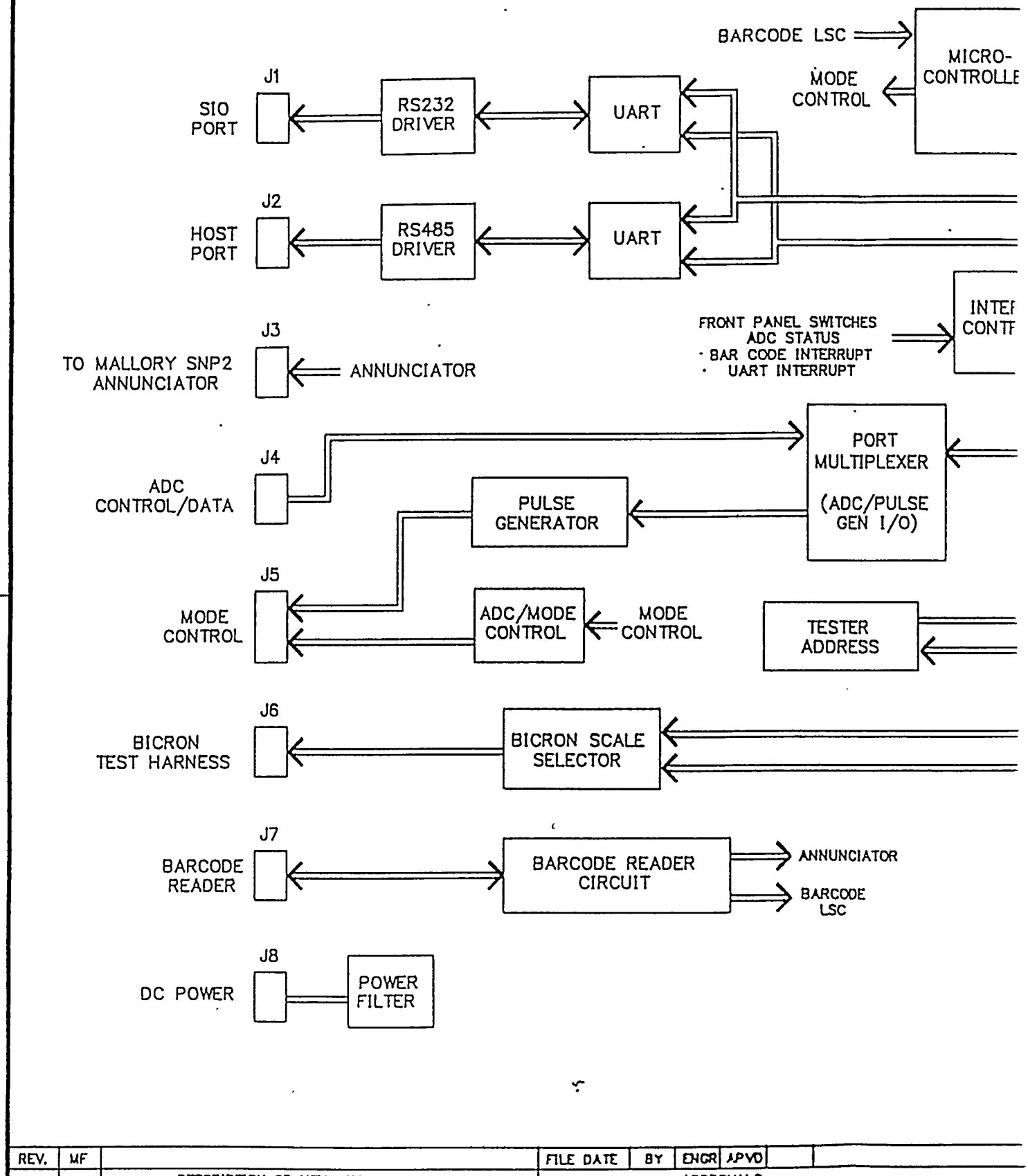


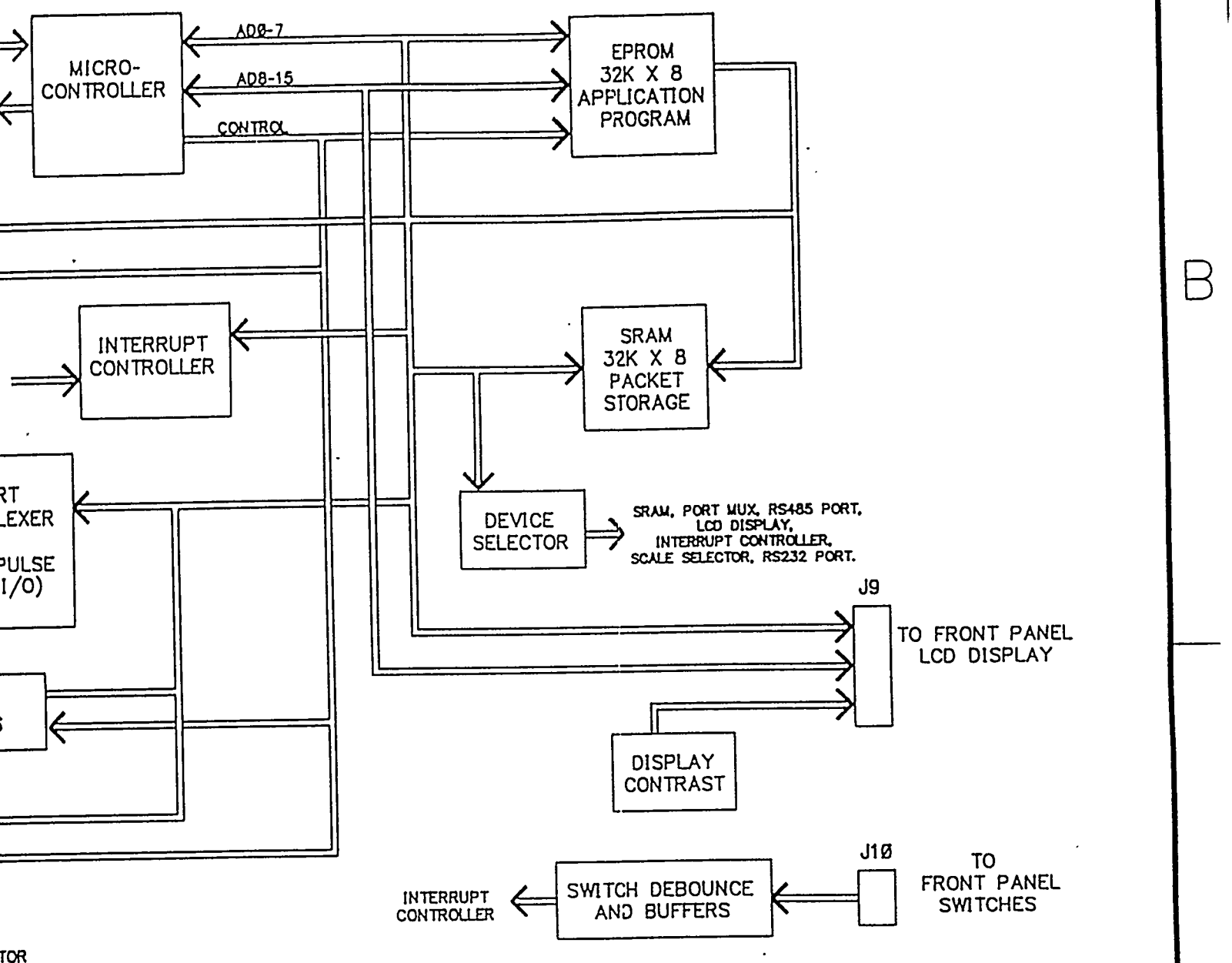
MARTIN MARIETTA

BOARD INTERCONNECTION
SCHEMATIC DIAGRAM

BICRON INSTRUMENT TESTER

SCALE	SIZE	DRAWING NO.	REV
NTS	I & C	0-6350-012	0


4
3
B
A
BARCODE LSC
MICRO-
CONTROLLE
MODE
CONTROL
J1
SIO
PORT
RS232
DRIVER
UART
J2
HOST
PORT
RS485
DRIVER
UART
INTEF
CONTF
FRONT PANEL SWITCHES
ADC STATUS
· BAR CODE INTERRUPT
· UART INTERRUPT
J3
TO MALLORY SNP2
ANNUNCIATOR
ANNUNCIATOR
PORT
MULTIPLEXER
(ADC/PULSE
GEN I/O)
J4
ADC
CONTROL/DATA
PULSE
GENERATOR
J5
MODE
CONTROL
ADC/MODE
CONTROL
MODE
CONTROL
TESTER
ADDRESS
J6
BICRON
TEST HARNESS
BICRON SCALE
SELECTOR
J7
BARCODE
READER
BARCODE READER
CIRCUIT
ANNUNCIATOR
BARCODE
LSC
J8
DC POWER
POWER
FILTER
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3



REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

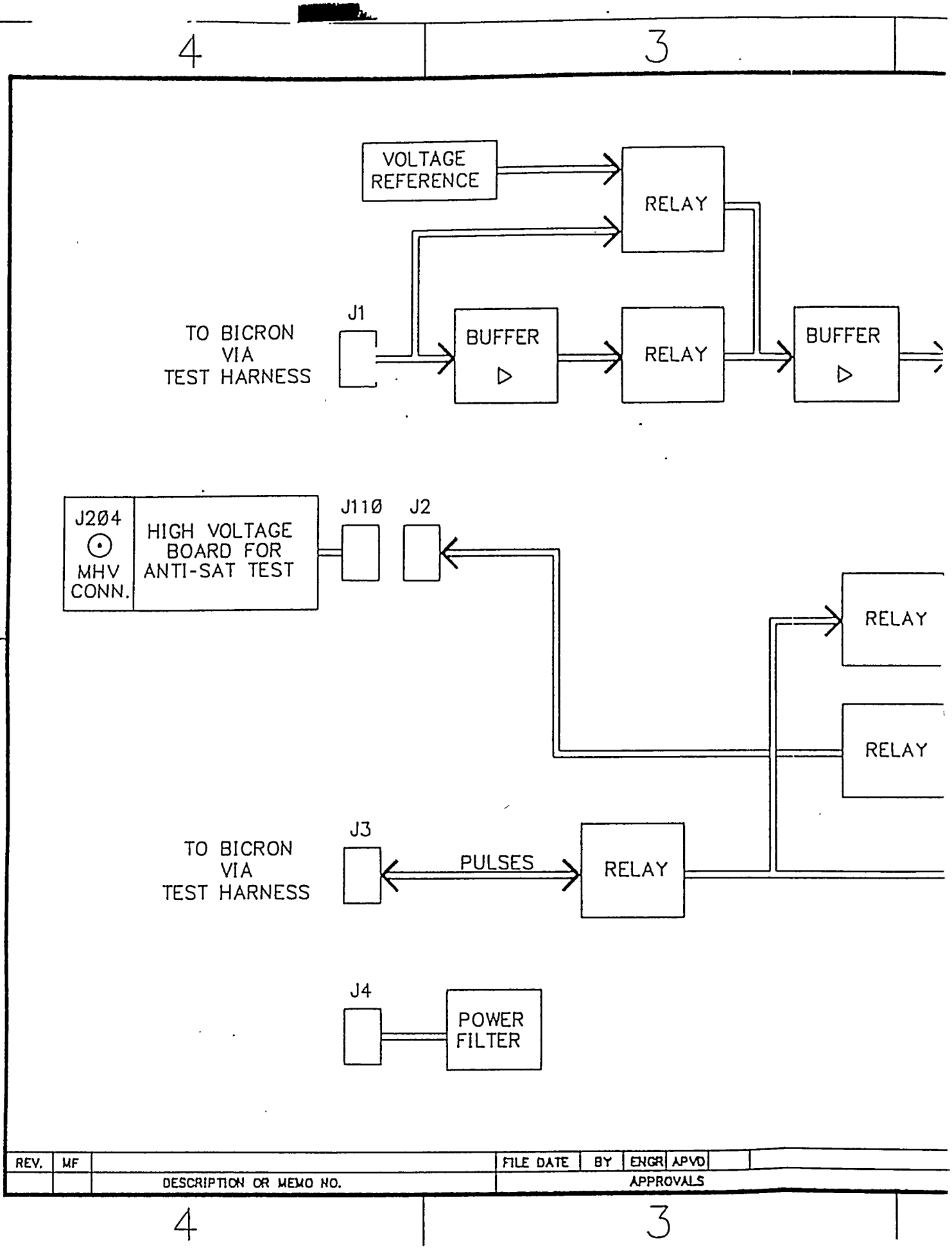
DIGITAL CONTROLLER BOARD FUNCTIONAL BLOCK DIAGRAM

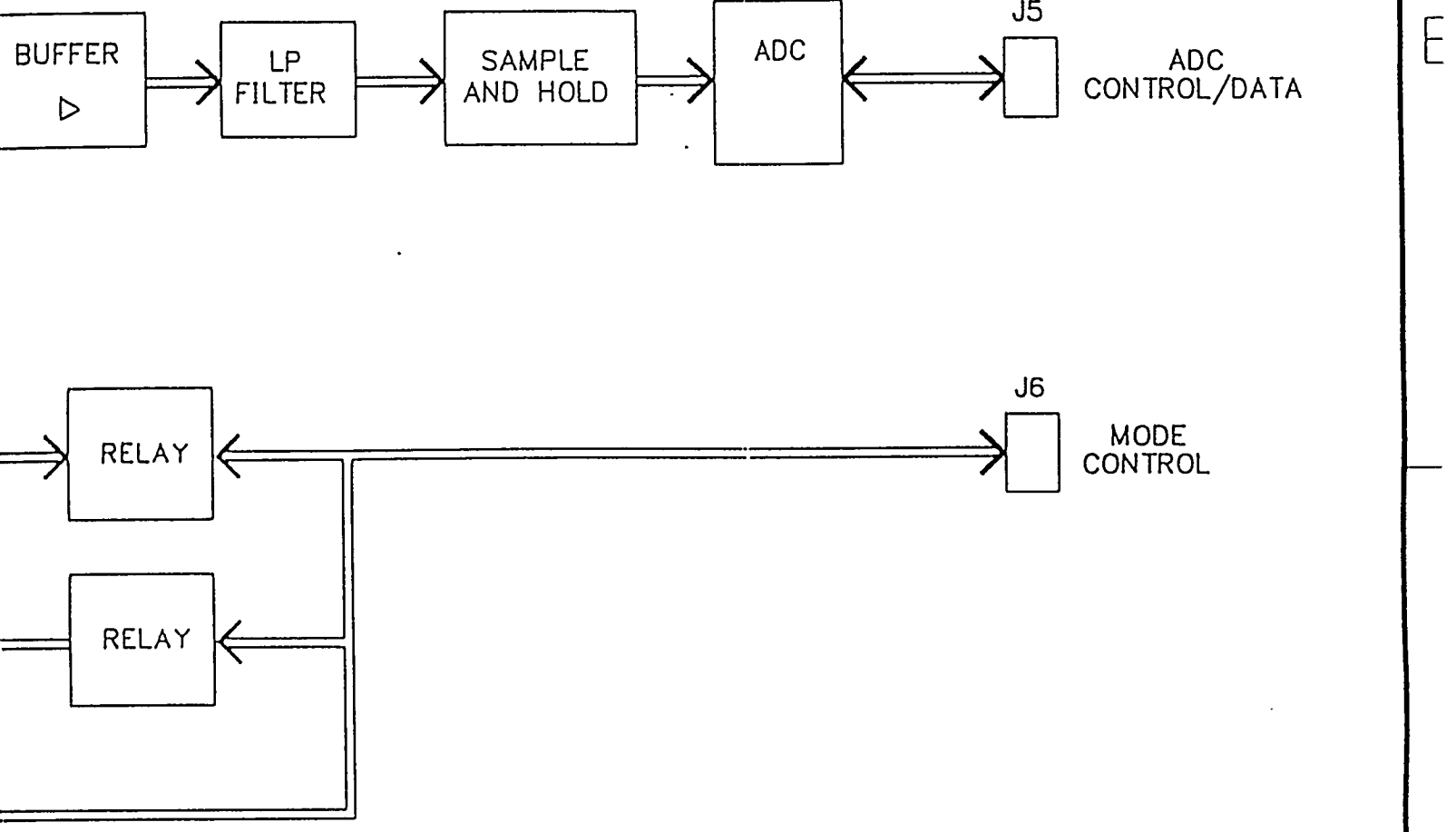
BICRON INSTRUMENT TESTER — TYPE: BD

FILE DATE	FILE POINT	BLDG	SHT. OF	WF
	HC- EC-		1 1	

SCALE	ID		REV
	I&C	Q-6350-013	Ø

DRAWN BY		DATE
	R.E. COOPER	7/95
DESIGNER CHECK		9/95
DESIGNER		9/95
APPROVED	D. A. Clayton	9/95
	R T Bradley	9/95
DRAWING APPROVALS		DATE


4
3
B
A
VOLTAGE REFERENCE
RELAY
J1
TO BICRON VIA TEST HARNESS
BUFFER
RELAY
BUFFER
J204
MHV CONN.
HIGH VOLTAGE BOARD FOR ANTI-SAT TEST
J110
J2
RELAY
RELAY
J3
TO BICRON VIA TEST HARNESS
PULSES
RELAY
J4
POWER FILTER
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3



REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

ANALOG BOARD FUNCTIONAL BLOCK DIAGRAM

BICRON INSTRUMENT TESTER — TYPE: BD

FILE DATE	FILE POINT	BLDG	SHT. OF	WF
	HC- EC-		1 1	

SCALE	ID		REV
	I&C	Q-6350-014	Ø

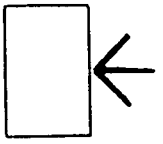
DRAWN BY	R.E. COOPER	7/95
CHECKED		9/95
		9/95
APPROVED	A. Clayton	9/95
		9/95
	DRAWING APPROVALS	DATE

4 3

B

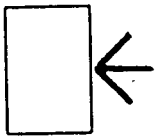
P9

TO DIGITAL
CONTROLLER BOARD



P10

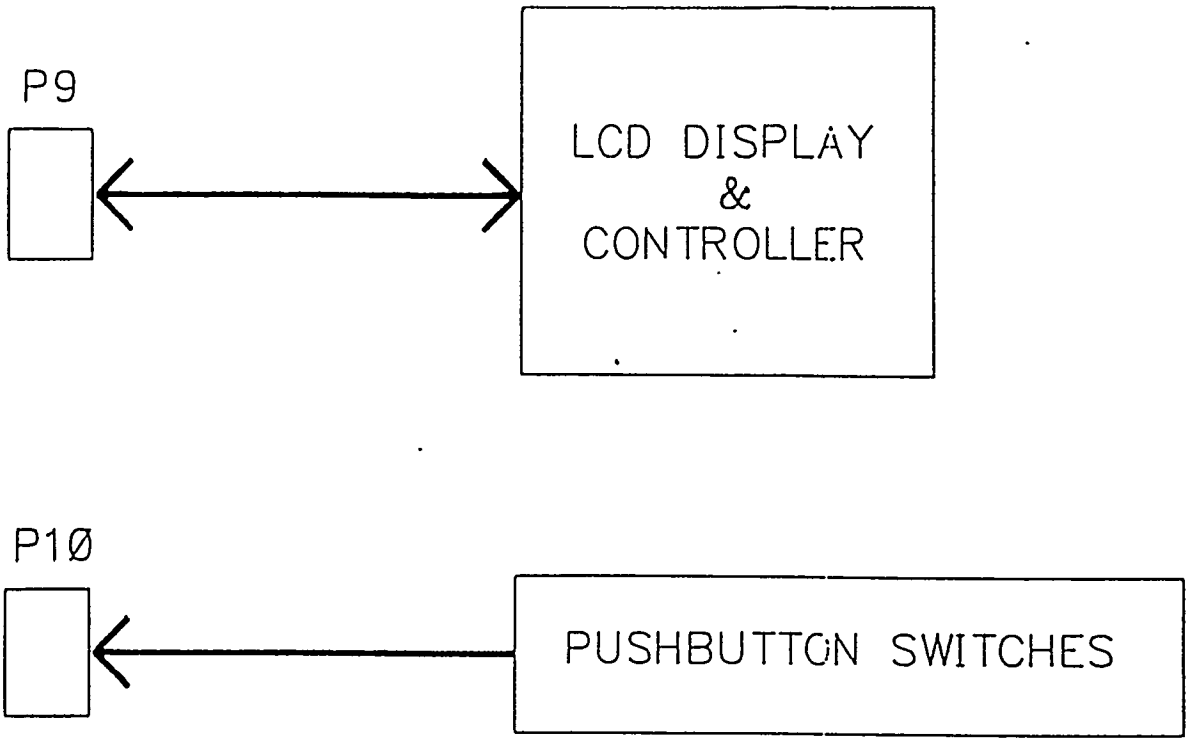
TO SWITCH DEBOUNCE & BUFFERS
ON CONTROLLER BOARD



A

REV.	MF		FILE DATE	BY	ENGR	APVD		
		DESCRIPTION OR MEMO NO.	APPROVALS					

4 3



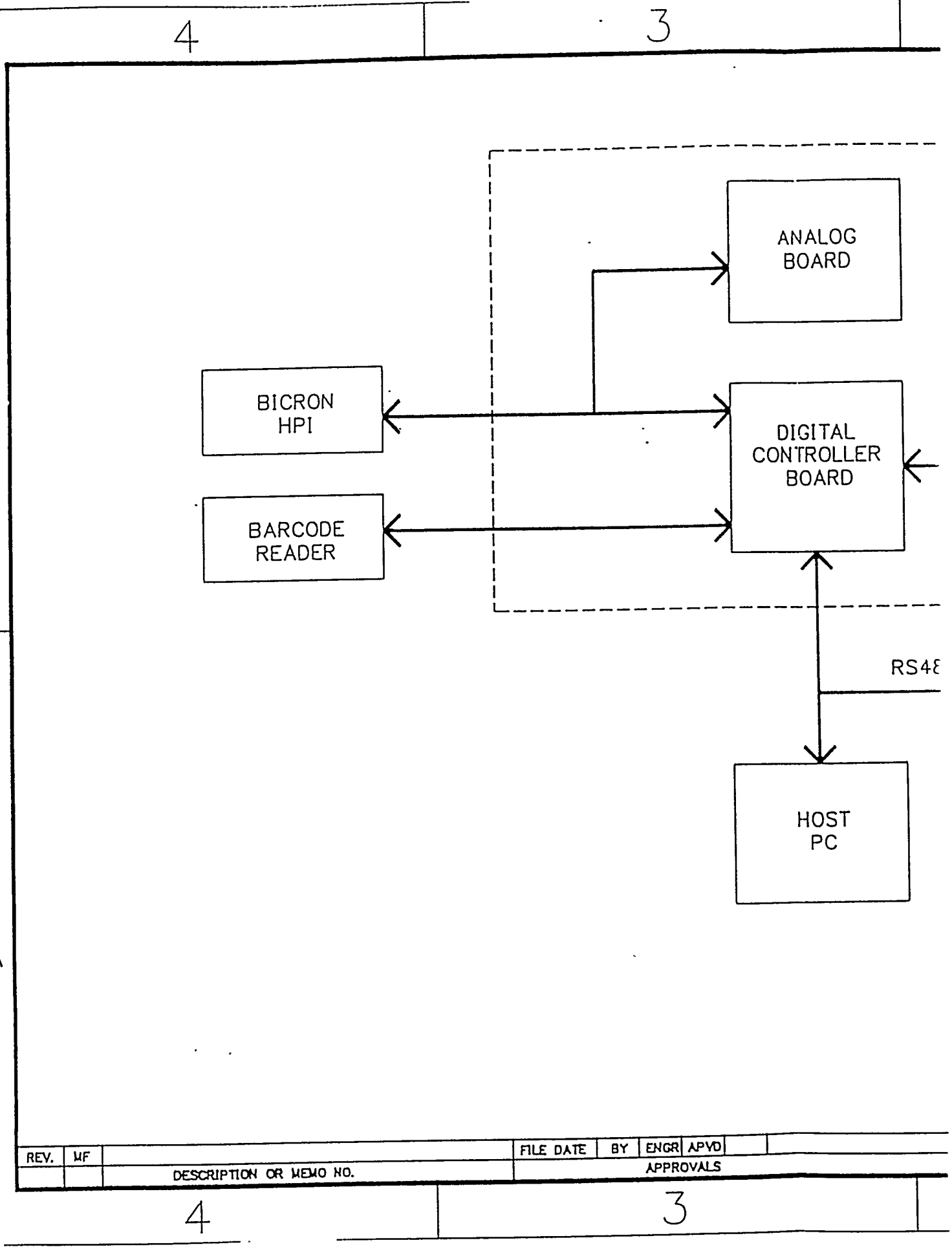
REFERENCE DRAWINGS	NUMBER

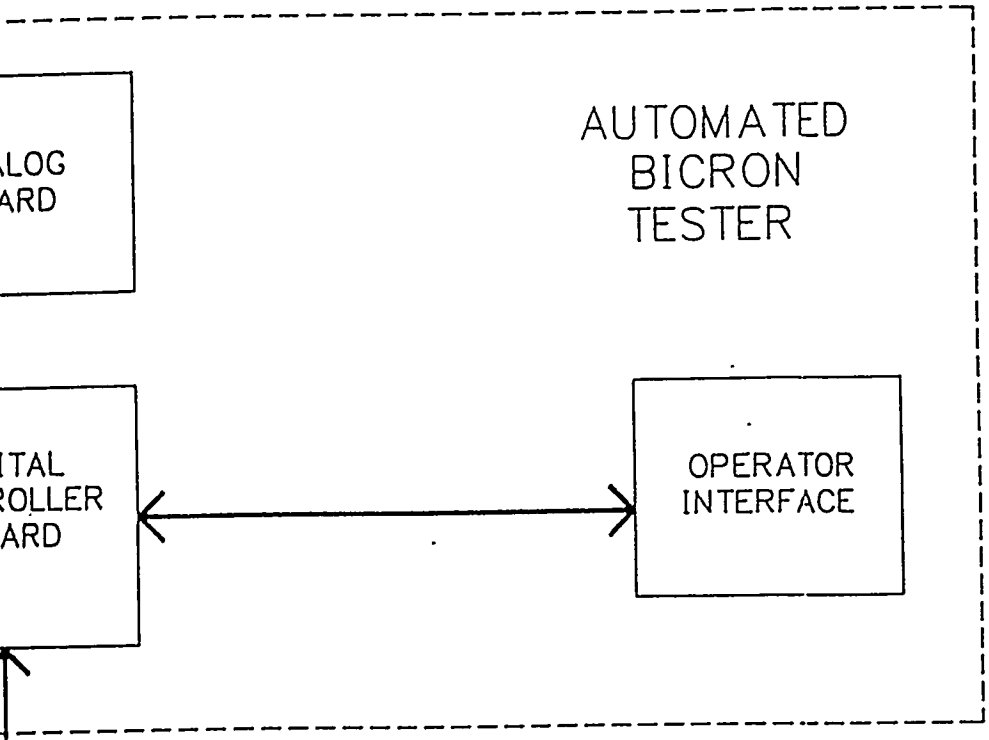
MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

OPERATOR INTERFACE FUNCTIONAL BLOCK DIAGRAM

BICRON INSTRUMENT TESTER					TYPE BD
FILE DATE		FILE POINT HC- EC-	BLDG	SHT. 1 OF 8	WF
SCALE	ID I&C	Q-6350-015			REV Ø

DRAWN BY R.E. COOPER	7/95
DRAWING CHECK	9/95
	9/95
APPROVED A Clayton	9/95
R Kornblum	9/5.5
DRAWING APPROVALS	DATE


4
3
B
A
ANALOG BOARD
BICRON HPI
DIGITAL CONTROLLER BOARD
BARCODE READER
RS48
HOST PC
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3



RS485 MULTIDROP LINE

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

HIGH LEVEL BLOCK DIAGRAM

BICRON INSTRUMENT TESTER — TYPE: BD

DRAWN BY	R.E. COOPER	7/95
DRAWING CHECK		9/95
		9/95
APPROVED	Dwight Clayton	9/95
		9/95
DRAWING APPROVALS		DATE

FILE DATE	FILE POINT	BLDG	SHT. OF	M#
	HC- EC-		1 1	
SCALE	ID		REV
	I&C	Q-6350-016	Ø

INTERNAL DISTRIBUTION

1-5. G. S. Rao
6. S. R. Maddox
7. G. W. Turner
8. R. I. Vandermolen
9. G. T. Alley
10. B. G. Eads
11. R. A. Hess
12. J. M. Jansen
13. D. W. McDonald
14. G. N. Miller
15. J. D. White
16. R. E. Uhrig
17. J. O. Stiegler
18. R. T. Roseberry
19. P. J. Chiaro
20. D. A. Clayton
21. B. P. Adkisson
22-24. W. R. Blodgett
25-26. Central Research Library
27. Y-12 Technical Reference Section
28-29. Laboratory Records
30. Laboratory Records–Record Copy
31. ORNL Patent Section
32. I&C Division Publications Office

EXTERNAL DISTRIBUTION

33. Assistant Manager for Energy Research and Development, DOE-ORO, P.O. Box 2001, Oak Ridge, TN 37831-8600

34. Ron Moore, The RM Group, Inc., 12024 Broadwood Drive, Knoxville, TN 37922

35. David Norton, Houston Advanced Research Center, 4800 Research Forest Drive, The Woodlands, TX 77381

36. M. M. Sevik, Carderock Division, Naval Surface Warfare Center, Code 70, Bethesda, MD 20084-5000

37. Ernesto Suarez, Pratt & Whitney, P.O. Box 109600, Mail Stop 716-87, West Palm Beach, FL 33410

38-39. Office of Scientific and Technical Information, U.S. Department of Energy, P.O. Box 62, Oak Ridge, TN 37831

BICRON INSTRUMENT TESTER M

DRAWING NUMBER	DRAWING TITLE
Q6350-000	MASTER DRAWING LIST
Q6350-001	DIGITAL CONTROLLER BOARD SCHEMATIC DIAGRAM
Q6350-002	ANALOG BOARD SCHEMATIC DIAGRAM
Q6350-003	DIGITAL CONTROLLER PRINTED CIRCUIT BOARD
Q6350-004	ANALOG PRINTED CIRCUIT BOARD
Q6352-005	HIGH VOLTAGE PRINTED CIRCUIT BOARD
Q6350-006	TEST HARNESS PRINTED CIRCUIT BOARDS
Q6350-007	TEST HARNESS HOUSING MECHANICAL DETAILS
Q6350-007	TEST HARNESS HOUSING DETAILS WITH 3-D VIEW
Q6350-008	HOUSING TOP MECHANICAL DETAILS
Q6350-009	BAR CODE READER LOGIC (GENERIC SERIAL UNIT) OBSOLETE
Q6350-010	HARNESS CABLING SCHEMATIC DIAGRAM
Q6350-011	EQUIPMENT AND RS-485 CONNECTIONS
Q6350-012	BOARD INTERCONNECTION SCHEMATIC DIAGRAM
Q6350-013	DIGITAL CONTROLLER BOARD FUNCTIONAL BLOCK DIAGRAM
Q6350-014	ANALOG BOARD FUNCTIONAL BLOCK DIAGRAM
Q6350-015	OPERATOR INTERFACE FUNTIONAL BLOCK DIAGRAM
Q6350-016	HIGH LEVEL BLOCK DIAGRAM

ASTER DRAWING LIST

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA ENERGY SYSTEMS, INC.

MASTER DRAWING LIST

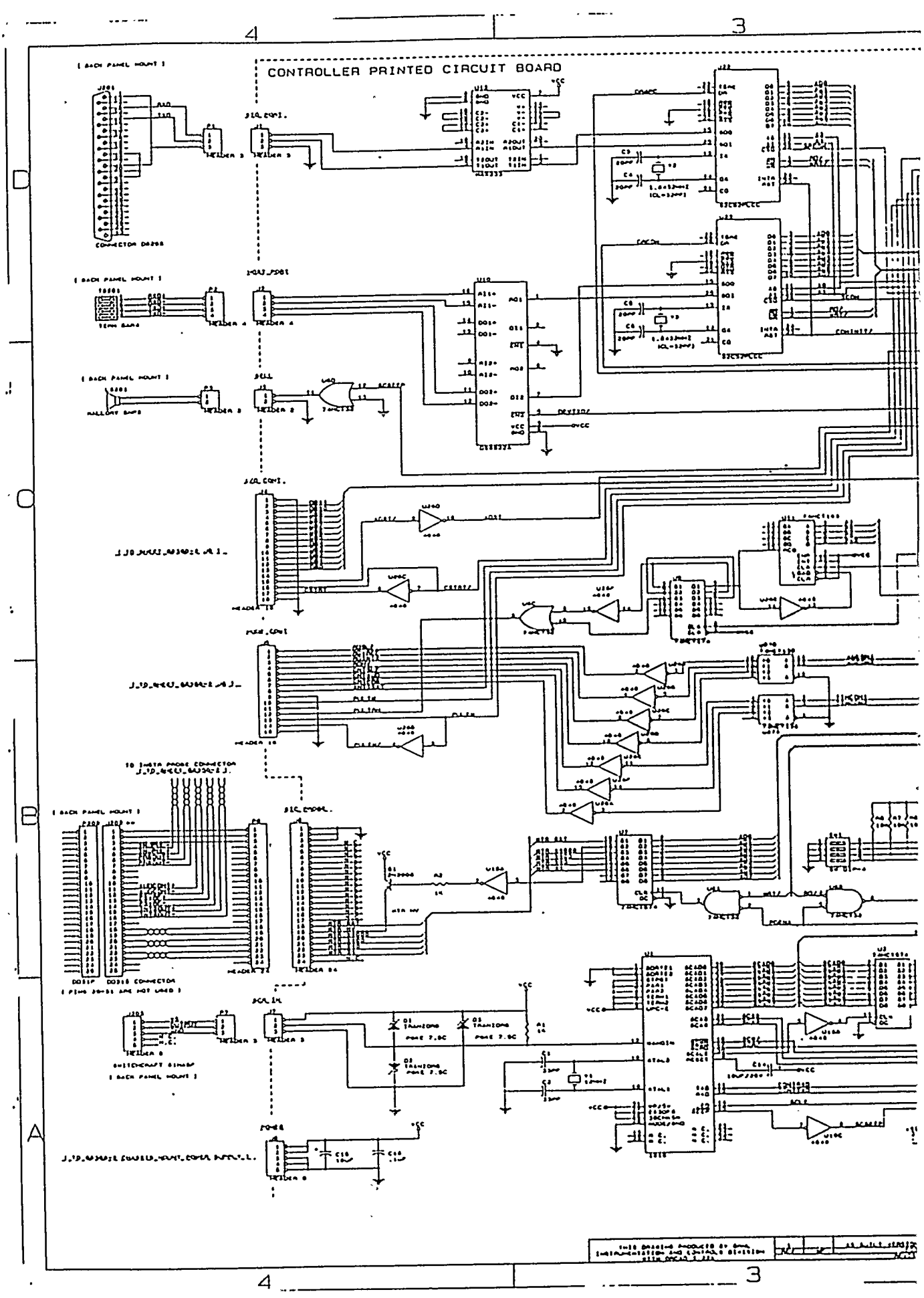
BICRON INSTRUMENT TESTER

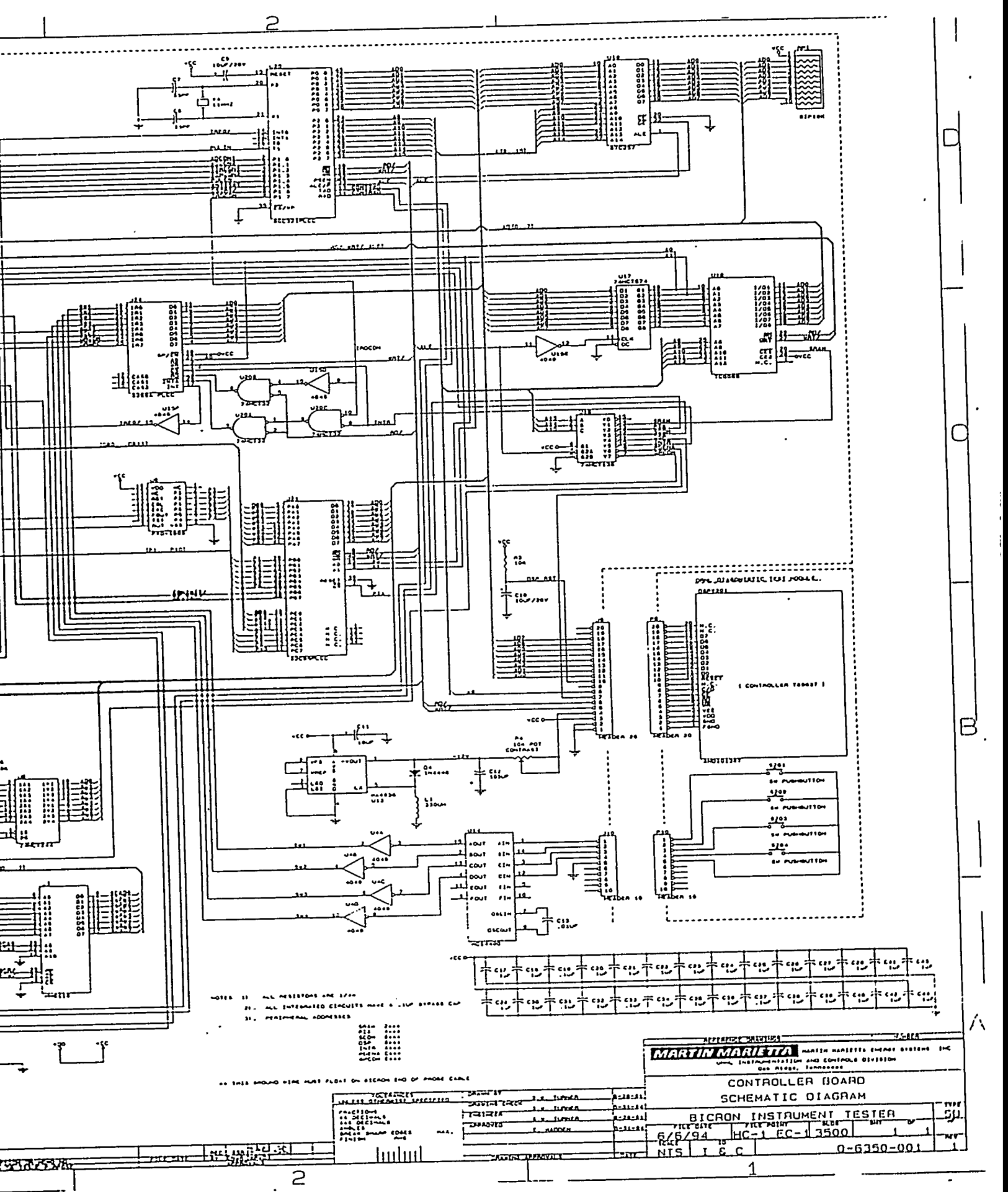
TOLERANCES UNLESS OTHERWISE SPECIFIED	
DRAWN BY	R.E. COOPER 7/85
DRAWING APPROVALS	DATE

FILE DATE	FILE POINT	BLDG	SHT.	OF	TYPE
	HC1 EC1	3500	1	1	BD

SCALE	ID	NUMBER	REV
N/A	I&C	Q6350-000	(1)

FILE DATE	BY	CHKD	CHKD	APVD	APVD	APVD
APPROVALS						


CONTROLLER PRINTED CIRCUIT BOARD



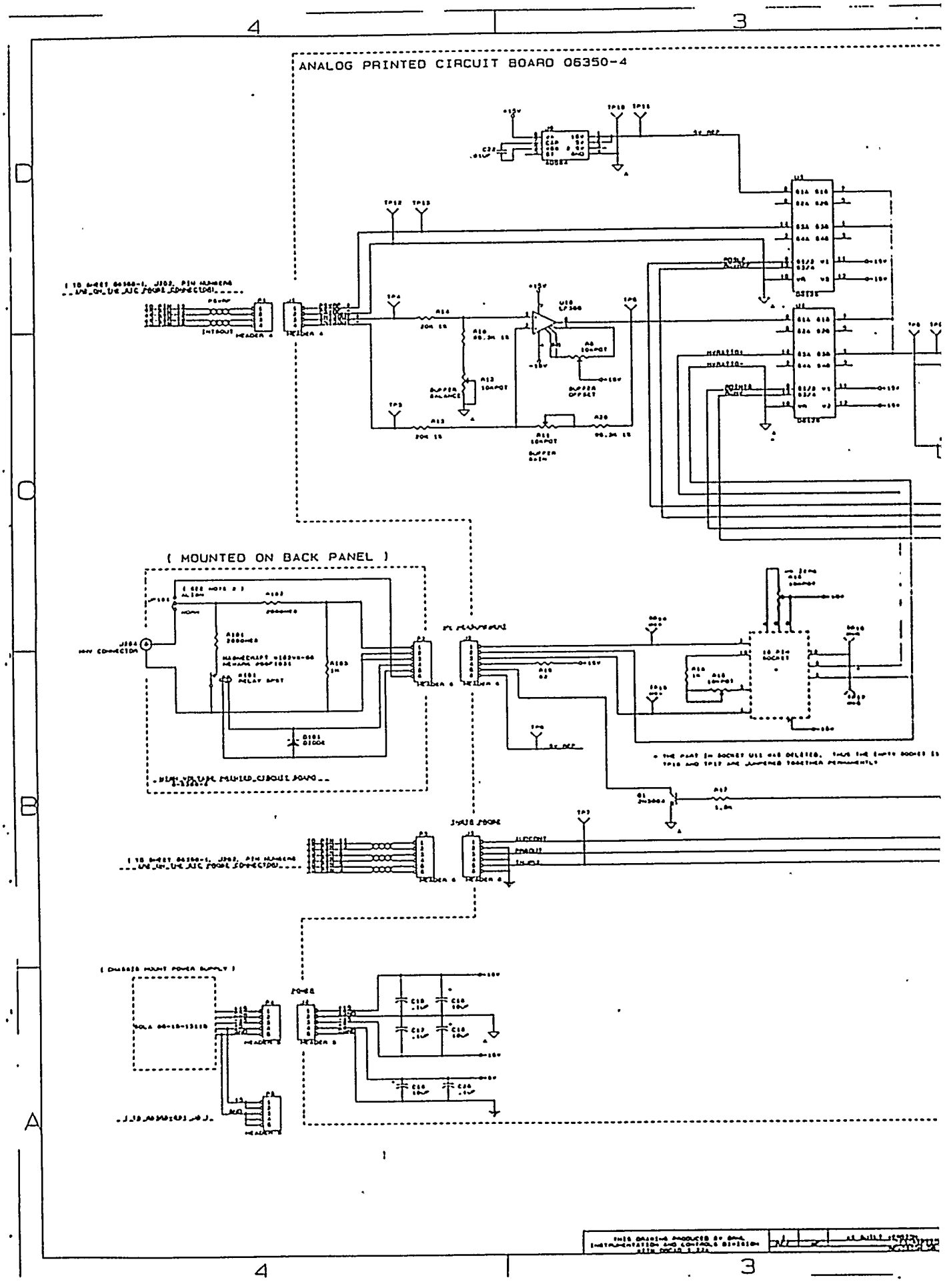
MARTIN MARIETTA

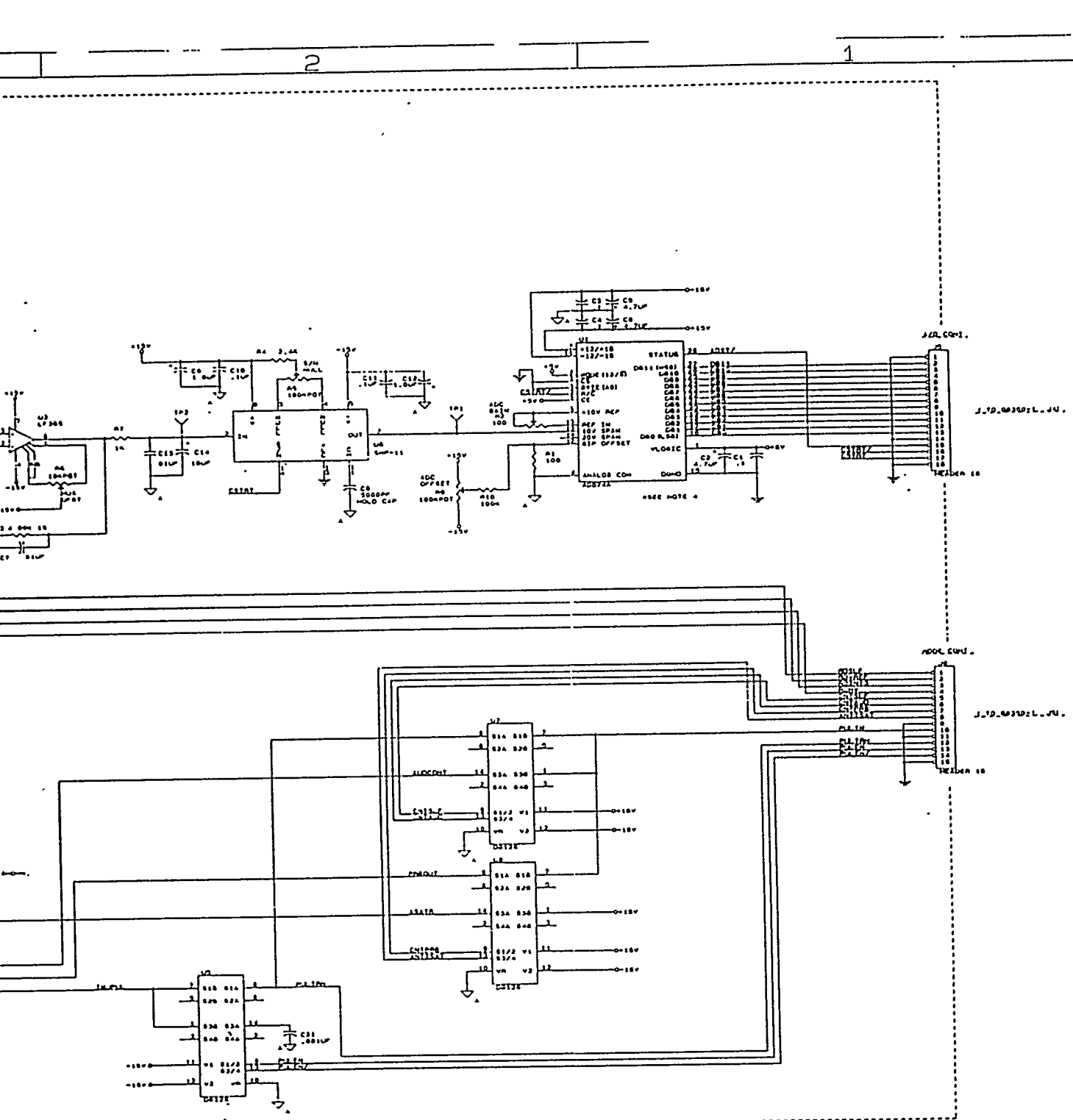
CONTROLLER BOARD
SCHEMATIC DIAGRAM

BICRON INSTRUMENT TESTER

FILE DATE	FILE POINT	SCALE	SHT	OF
6/6/94	HC-1 EC-1	3500	1	1

SCALE	IS	I & C	0-6350-001	REV 1


ANALOG PRINTED CIRCUIT BOARD 06350-4
(MOUNTED ON BACK PANEL)
4
3
D
C
B
A



NOTES:

1. ALL RESISTORS ARE 1/4W, 1% MF
2. ALIGNMENT PROCEDURE
   - A. SHORT TP HI0 AND TP HI1 BY JUMPERING UP ALH
   - B. ADJUST HV ZERO FOR 0 Vdc AT TP HIMTP HV0.
   - C. APPLY 10.0000 AT INPUT BY MOVING UP V4 TO THE TEST POSITION.
   - D. ADJUST HV TRIM FOR 50.00Vdc AT TP HIMTP HV0.
   - E. MOVE UP ALH BACK TO THE NORMAL POSITION.

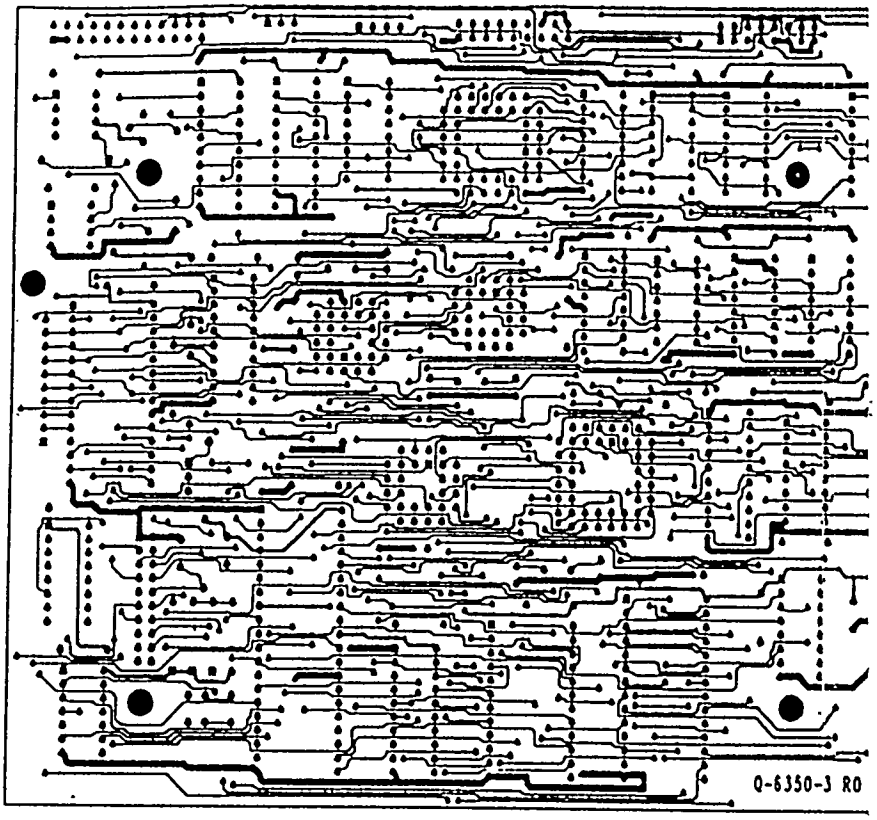
MARTIN MARIETTA — MARTIN MARIETTA ENERGY SYSTEMS, INC.
OAK RIDGE, TENNESSEE

ANALOG BOARD
SCHEMATIC DIAGRAM

DICON INSTRUMENT TESTER

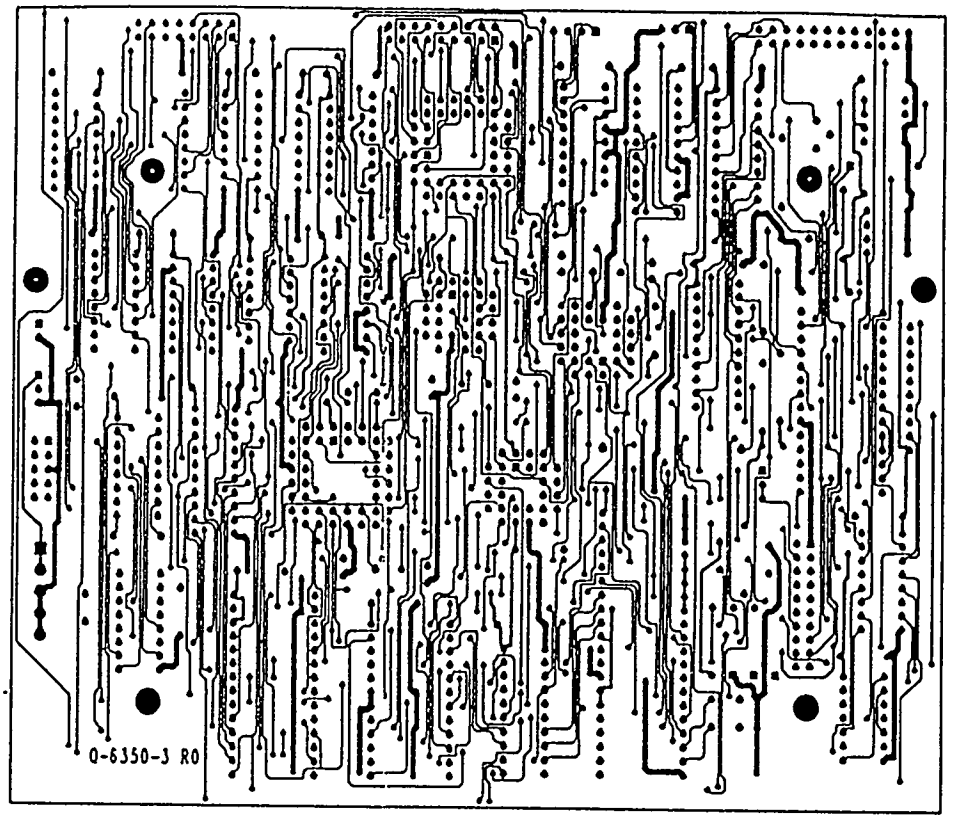
FILE DATE	FILE POINT	BLOB	LAY	SD.
6/6/94	HC-1 EC-1	3500	1	1

SCALE			REV
NTS	I & C	0-6350-002	1



FRONT

REV.	HF	DESCRIPTION	FILE DATE	BY	ENGR	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			



REAR

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE: MAR 21 '91
INITIAL: SS

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky Portsmouth, Kentucky

CONTROLLER PRINTED CIRCUIT BOARD

BICRON INSTRUMENT TESTER

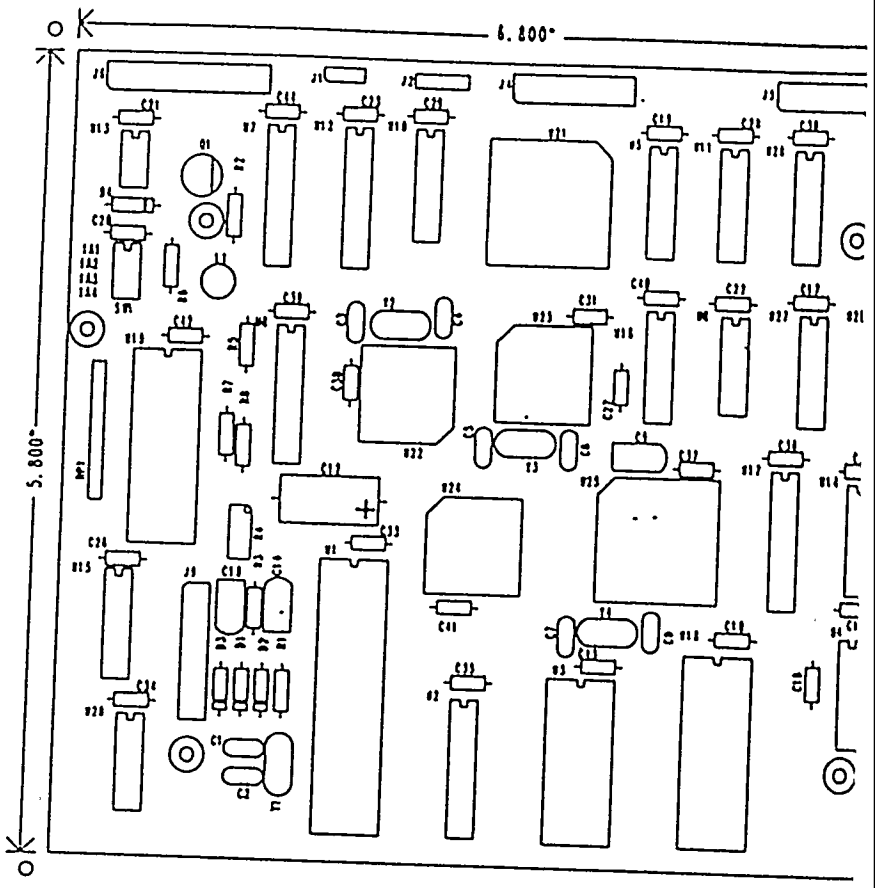
TYPE: PC

TOLERANCES UNLESS OTHERWISE SPECIFIED			
FRACTIONS	±	1/64"	
DECIMALS	±	.010"	
DECIMALS	±	.005"	
ANGLES	±	0° 30'	
BREAK SHARP EDGES			

DRAWN BY	A.A. MAPLES	3-91
ENGINEER	G. W. Turner	9-24-91
APPROVED	E. Maddox	9-24-91
DRAWING APPROVALS		DATE

FILE DATE	FILE POINT	BLOG	SHT.	OF	REV
9-25-91	HC-1 EC-1	3500	1	2	::

SCALE	IS		REV
1 : 1	I&C	Q-6350-3	Ø

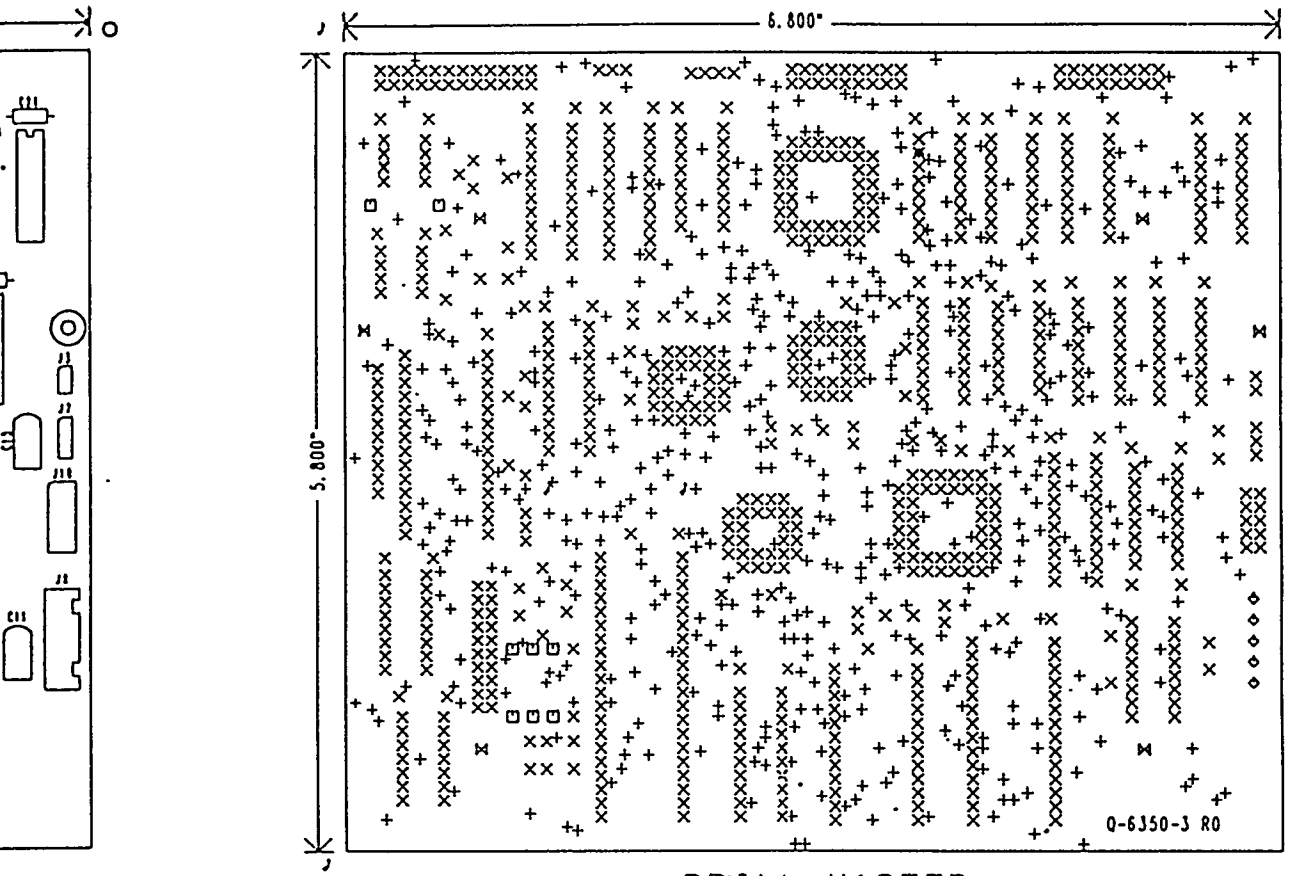


SILK SCREEN

NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/1-B-2-B.
2. ALL HOLES WITH LANDS ON BOTH SIDES OF THE BOARD SHALL BE 100% PLATED THRU. COPPER PLATING INSIDE THE HOLES SHALL BE .001" MINIMUM IN ACCORDANCE WITH MIL-C-14550. FRONT TO REAR REGISTRATION TO BE WITHIN .005".
3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728. THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.
4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.
5. DO NOT USE THIS DRAWING AS A WORK SHEET. MASTER POSITIVE(S) OF WIRING WILL BE SUPPLIED BY ORNL.
6. LIMIT ON ALL DIMENSIONS .015 IN UNLESS OTHERWISE NOTED.
7. HOLE LOCATION TOLERANCE .003 IN.
8. DIMENSIONS OF ALL HOLES NOTED BELOW ARE FINISHED HOLE DIAMETERS AFTER PLATING THRU.

REV.	MF	DESCRIPTION				
BRAND SIM PADS-PCB		REVISION OR ISSUE PURPOSE	FILE DATE	BY	ENGR	APVD
			APPROVALS			



DRILL MASTER

SIZE	QTY	SYM
28	490	+
35	838	×
48	8	□
52	5	◇
55	2	2
108	3	3
125	1	⋈

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE:
INITIAL: MAR 21 '91

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

CONTROLLER
PRINTED CIRCUIT BOARD

BICRON INSTRUMENT TESTER

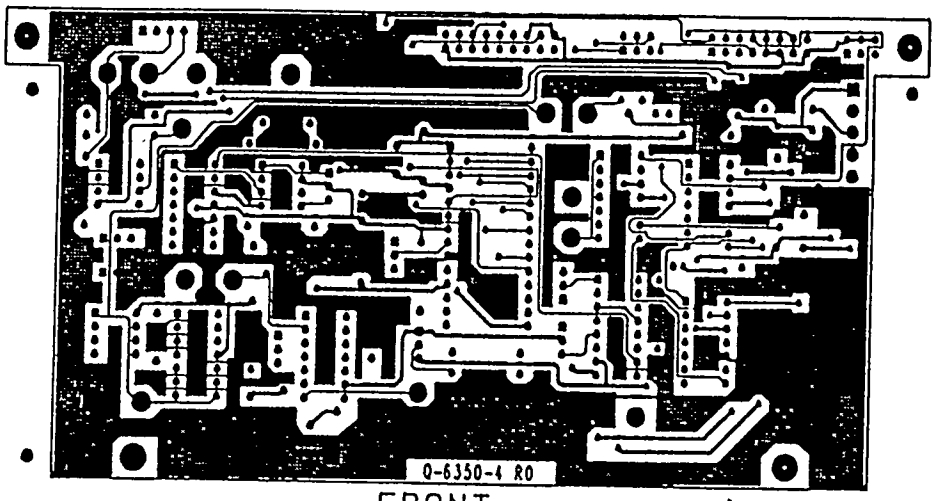
TYPE: PC

TOLERANCES UNLESS OTHERWISE SPECIFIED		
FRACTIONS	±	1/64"
x DECIMALS	±	.010"
xx DECIMALS	±	.005"
ANGLES	±	0° 30'
BREAK SHARP EDGES		

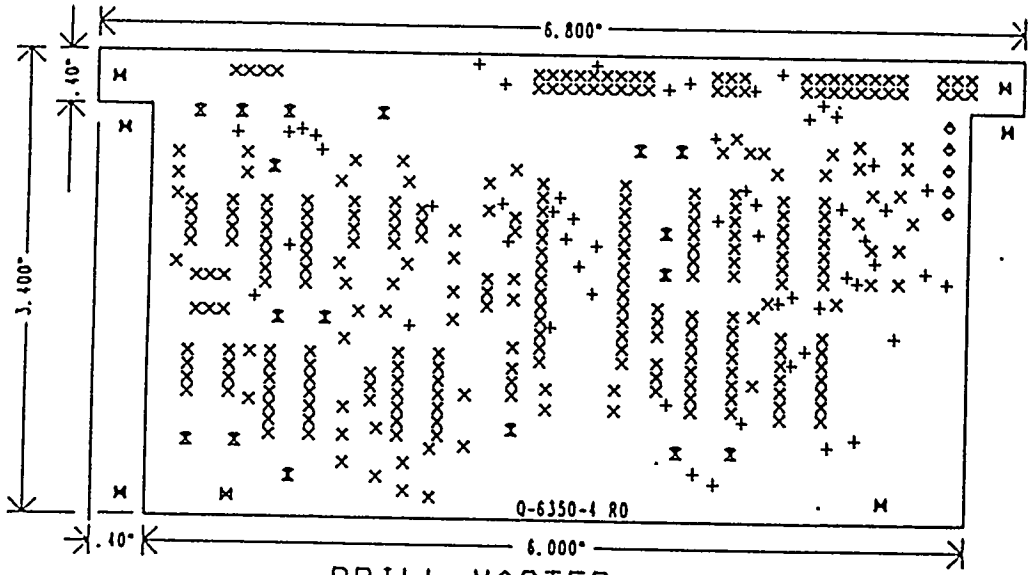
DRAWN BY	R.A. MAPLES	3-91
DRAWING CHECK		
ENGINEER	B. W. Turner	9-24-91
APPROVED	E. Madden	9-24-91
DRAWING APPROVALS		DATE

FILE DATE	FILE POINT	BLOCK	SHT. OF	REV
9-25-91	HC-1 EC-1	3500	2 2	A

SCALE	IS		REV
1 : 1	I&C	Q-6350-3	A



FRONT



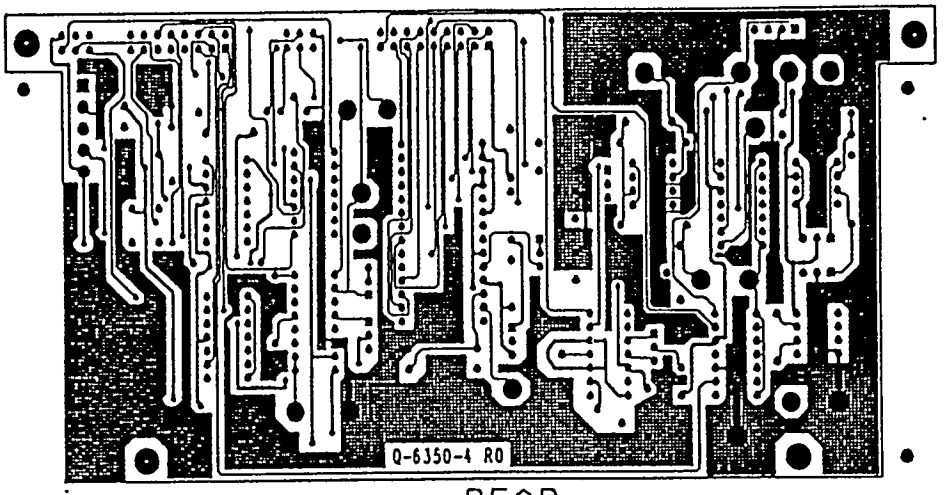
DRILL MASTER

NOTES:

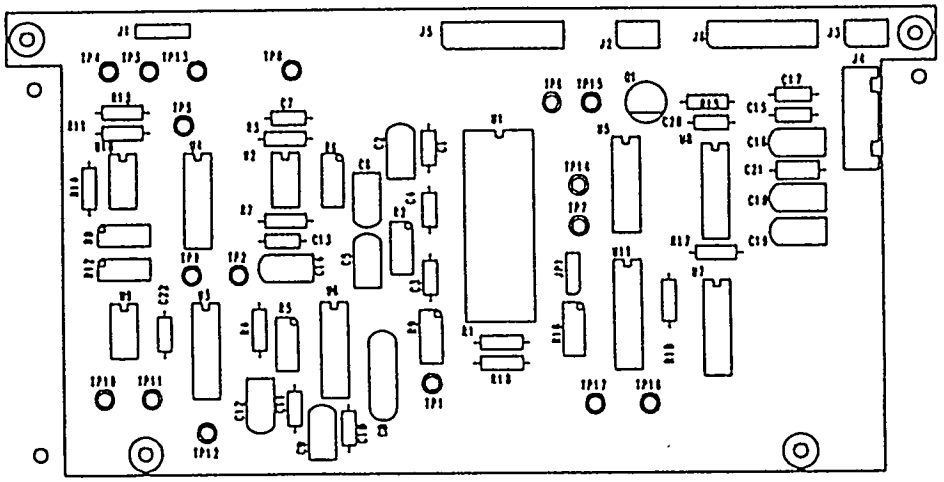
1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/1-8-2-8.
2. ALL HOLES WITH LANDS ON BOTH SIDES OF THE BOARD SHALL BE 100% PLATED THRU. COPPER PLATING INSIDE THE HOLES SHALL BE .001" MINIMUM IN ACCORDANCE WITH MIL-C-14550. AMOUNT TO BEAR REGISTRATION TO BE WITHIN .005".
3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728. THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.
4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.
5. DO NOT USE THIS DRAWING AS A FINAL SHEET. OF VIAINS WILL BE SUPPLIED BY DANL.
6. LIMIT ON ALL DIMENSIONS .015 IN UNLESS
7. HOLE LOCATION TOLERANCE .003 IN.
8. DIMENSIONS OF ALL HOLES NOTED BELOW ARE AFTER PLATING THRU.

REV.	MF	DESCRIPTION	FILE DATE	BY	ENGR	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			

UNLESS OTHE
TOLERANCES
ON DECIMALS
XXX DECIMALS
ANGLES
BREAK SHARP



REAR



SILK SCREEN

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE:
INITIAL: MAR 21 '91 S.S

SIZE	DIM	TOL
.XX	.XX	+
.XXX	.XXX	×
ANG	X	◇
FIN	XX	⧗
.XXX	X	⋈

ASTER POSITIVE(S)

ERWISE NOTED.

OLDER MASK DIMENSIONS

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operating for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

ANALOG PRINTED CIRCUIT BOARD

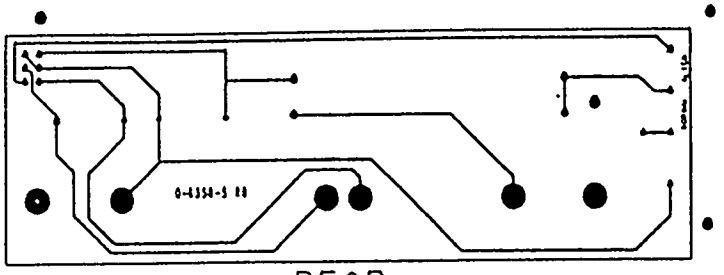
BICRON INSTRUMENT TESTER

TYPE: PC

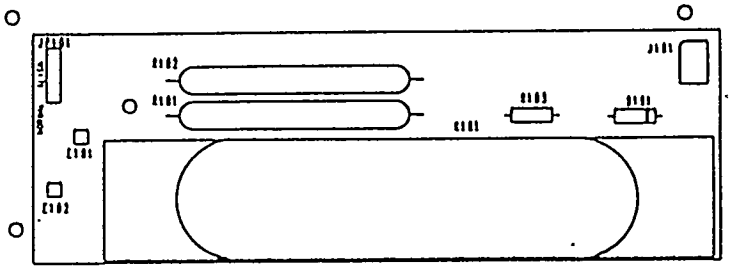
DRAWN BY		DATE
	R.A. MAPLES	3-91
	L.W. Swan	9.24.91
	E. Mallon	9.24.91
DRAWING APPROVALS		DATE

FILE DATE	FILE POINT	BLOCK	SHT. OF	REV
9-25-91	HC-1 EC-1	3500	1 1	

SCALE	ID		REV
1 : 1	I&C	Q-6350-4	0



REAR



SILK SCREEN

NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-E-1/0-0-2-0.

2. THIS BOARD ONLY HAS CIRCUIT ON THE BACK WITH A SILK SCREEN ON THE FRONT.

3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728. THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUM.

4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.

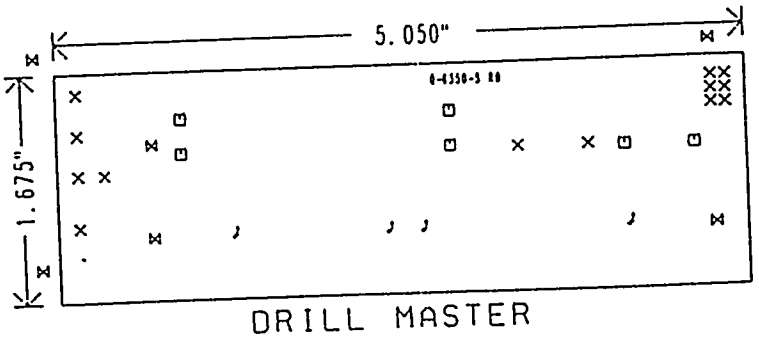
5. DO NOT USE THIS DI
OF WIRING WILL BE

6. LIMIT ON ALL DIMEN

7. WHERE LOCATION TOL

8. DIMENSIONS OF HOL

REV.	MF	DESCRIPTION	FILE DATE	BY	CHKD	APVD
DASH NUM PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			



DRILL MASTER

SIZE	QTY	SYM
35	13	X
40	6	□
93	2	2
125	8	⋈
136	2	Y

MASTER POS.
DO NOT USE IN
P. C. SHOP
DATE: MAR 21 '91
INITIAL:

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

HIGH VOLTAGE PRINTED CIRCUIT BOARD

BICRON INSTRUMENT TESTER

TYPE: PC

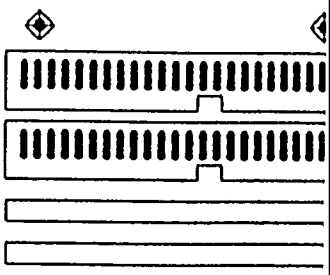
FILE DATE	FILE POINT	BLDG	SHT.	OF
9-25-91	HC-1 EC-1	3500	1	1

SCALE	ID		REV
1 : 1	I&C	Q-6350-5	0

INE AS A WORK SHEET. MASTER POSITIVE(S)
PPLIED BY DRWG.
0mg .015 IN UNLESS OTHERWISE NOTED.
NCE .003 IN.
LES NOTED BELOW AND BROKEN ARE BROKEN.

TOLERANCES
SS OTHERWISE SPECIFIED
[illegible] ± .760"
[illegible] ± .010"
[illegible] ± .005"
[illegible] ± 0° 30'
SHARP EDGES

DRAWN BY	B.A. MAPLES	3-91
[illegible]		
[illegible]	B W Turner	9.27.91
[illegible]	E. Mallon	9-27-91
DRAWING APPROVALS		DATE



CIRCUIT

. NOTES:

1. MATERIAL TO BE .[illegible]
FR4, PER MIL-P-1[illegible]
FL-GFP-0625-C1/8-

2. REGISTRATION TO [illegible]

3. CLEAN AND SOLDER
MIL-P-81728.THE [illegible]
THICK MINIMUN.

4. AFTER PROCESSING,
AND SCRATCHES.

	HOLE
◆	#48 .
●	#51 .
::	1/8 .

NOTES:

1. MATERIAL TO BE .062 IN THICK EPOXY FIBERGLASS TYPE FR4, PER MIL-P-13949/4. SPECIFIC DESIGNATION SHALL BE FL-GFP-062-C-1/8-8-2-8.

2. THIS BOARD ONLY HAS CIRCUIT ON THE BACK.

3. CLEAN AND SOLDER PLATE ALL COPPER IN ACCORDANCE WITH MIL-P-81728.THE FUSED TIN-LEAD SHALL BE .0003 INCH THICK MINIMUN.

4. AFTER PROCESSING, ETCHED BOARDS TO BE FREE OF VOIDS AND SCRATCHES.

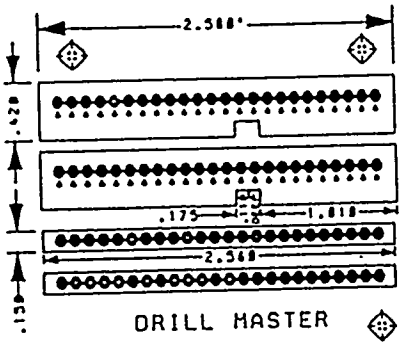
5. DO NOT USE THIS OF VIAING WILL

6. LIMIT ON ALL DI

7. HOLE LOCATION [illegible]

8. DIMENSIONS OF [illegible]

REV.	MF	DESCRIPTION	FILE DATE	BY	CHKR	APVD
DRAWN WITH PADS-PCB		REVISION OR ISSUE PURPOSE	APPROVALS			



IN THICK EPOXY FIBERGLASS TYPE
L. SPECIFIC DESIGNATION SHALL BE
C.

ITHIN .003".

E ALL COPPER IN ACCORDANCE WITH
TIN-LEAD SHALL BE .0003 INCH

CHED BOARDS TO BE FREE OF VOIDS

E	QUAN.
IN DIAMETER	48
IN DIAMETER	96
IN DIAMETER	3

RAWING AS A WORK SHEET. MASTER POSITIVE(S)
SUPPLIED BY ORNL.
NSIONS .010 IN UNLESS OTHERWISE NOTED.
RANCE .002 IN.
HOLES NOTED BELOW ARE FINISHED HOLE DIAMETERS.

TOLERANCES UNLESS OTHERWISE SPECIFIED			
FRACTIONS ± 1/64"	Drawn By	R.A. MAPLES	3-91
2 DECIMALS ± .010"	Checked By		
3 DECIMALS ± .005"	Engineer		
ANGLES ± 0° 30'	Approved		
BREAK SHARP EDGES			
	DRAWING APPROVALS		DATE

REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

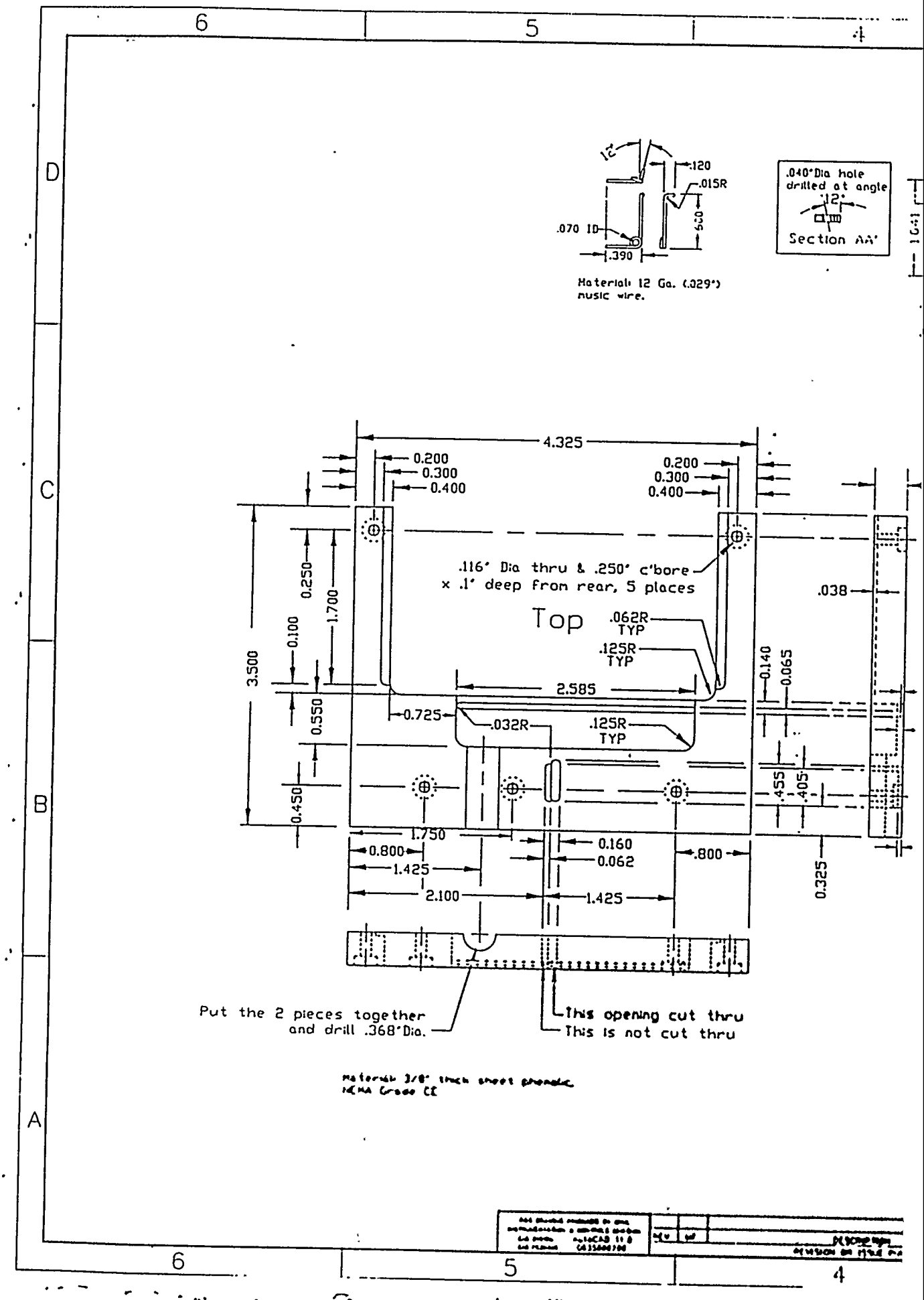
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

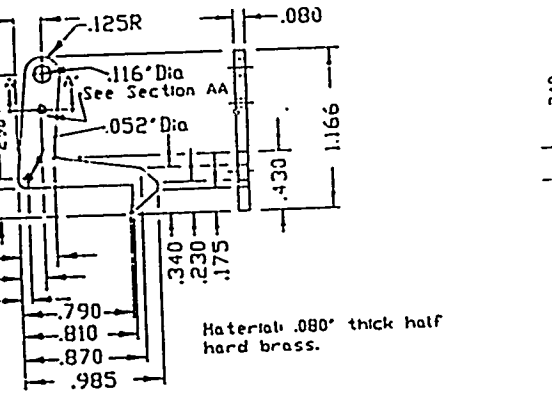
PROBE
PRINTED CIRCUIT BOARDS

BICRON INSTRUMENT TESTER

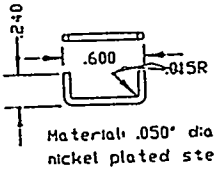
TYPE	FILE DATE	FILE POINT	BLOCK	SHT. OF	REV
PC		HC-1 EC-1	3500	1 1	

SCALE	ID		REV
1 : 1	I&C	Q-6350-6	0

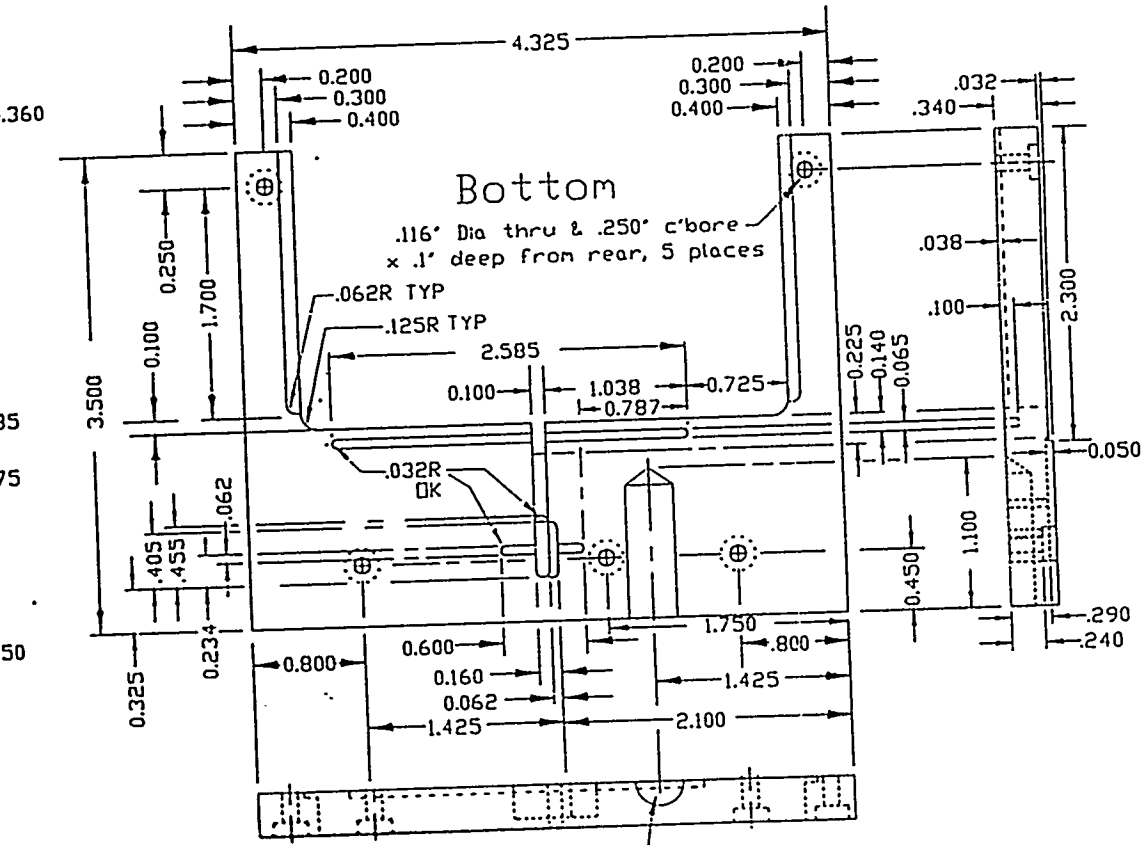

6
5
4
D
C
B
A
12°
.120
.015R
.070 ID
.390
.650
.040" Dia hole drilled at angle
12°
Section AA'
Material 12 Ga. (.029") music wire.
4.325
0.200
0.300
0.400
.116" Dia thru & .250" c'bore x .1" deep from rear, 5 places
Top
.062R TYP
.125R TYP
.038
3.500
0.250
1.700
0.100
2.585
0.140
0.065
0.550
0.725
.032R
.125R TYP
.455
.405
0.450
1.750
0.800
1.425
2.100
0.160
0.062
.800
1.425
0.325
Put the 2 pieces together and drill .368" Dia.
This opening cut thru
This is not cut thru
Material 3/8" thick sheet phenolic NEMA Grade CE



Material .080" thick half hard brass.



Material .050" dia nickel plated steel. Use a large paper clip.



Put the 2 pieces together and drill .368" Dia.

Material 3/8" thick sheet phenolic. NEMA Grade CE

REFERENCE DRAWINGS | MARKER

MARTIN MARIETTA | MARTIN MARIETTA ENERGY SYSTEMS, INC

PROBE HOUSING MECHANICAL DETAILS

BICRON INSTRUMENT TESTER

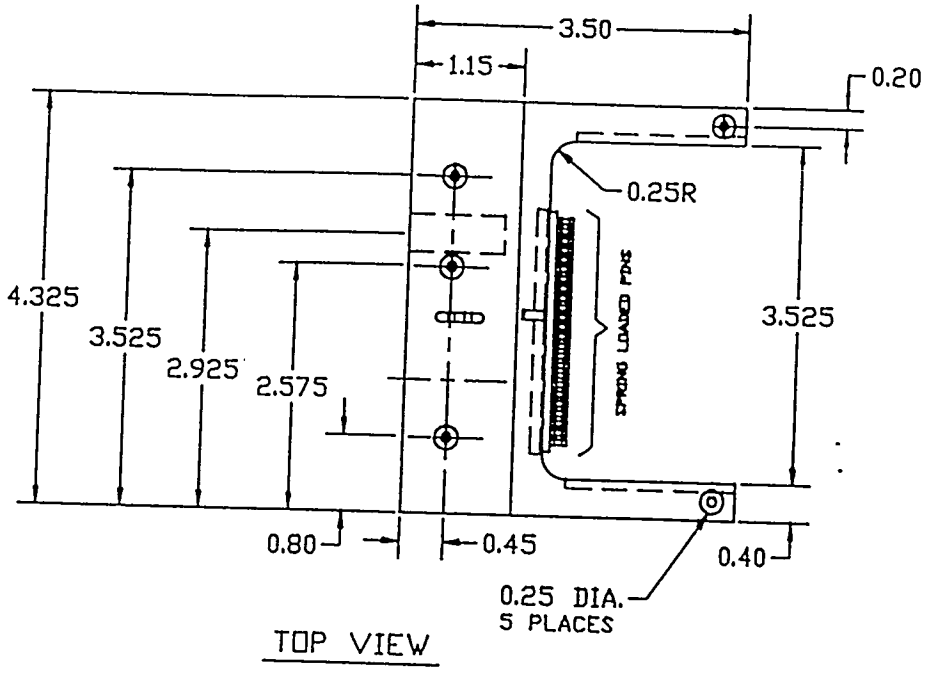
FILE DATE	FILE POINT	MDG	SHT	OF
MAR 1 9 1992	HC-1 EC-1	3500	1	1

SCALE	TO	DRAWING NO.	REV
	I & C	0-6350-7	0

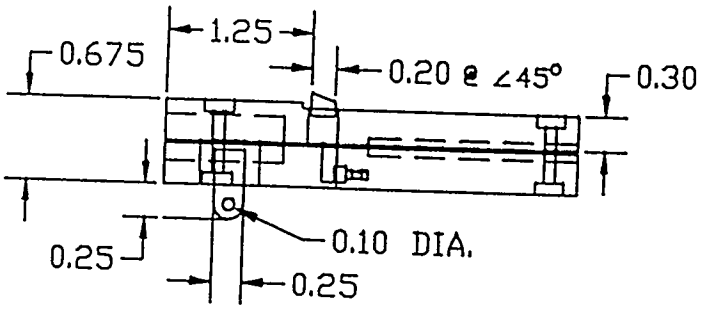
TOLERANCES UNLESS OTHERWISE SPECIFIED

DRAWING APPROVALS		DATE
R. A. MAPLES		3-91
		3-19-92
		3-92

TYPE MD

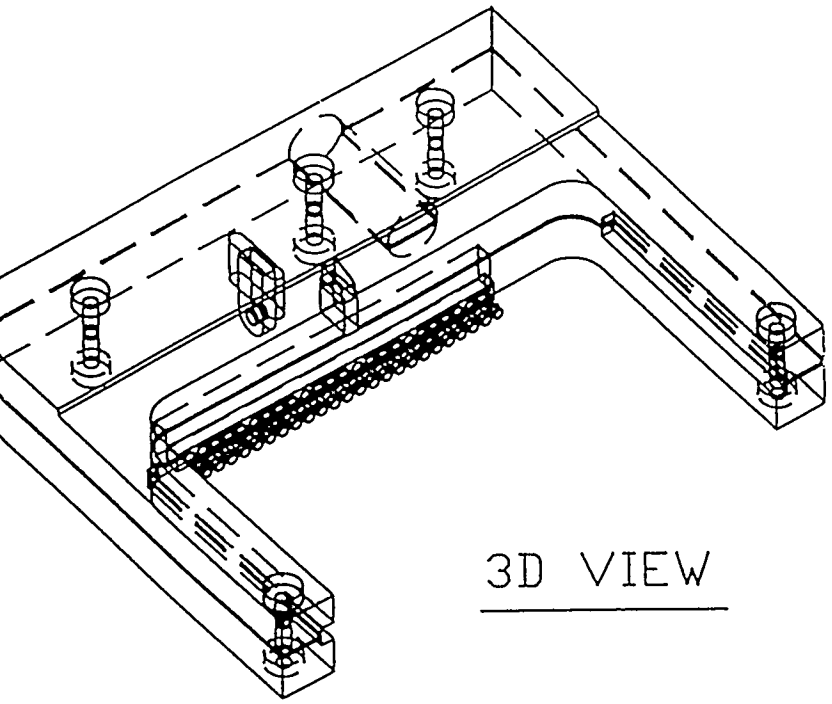


TOP VIEW

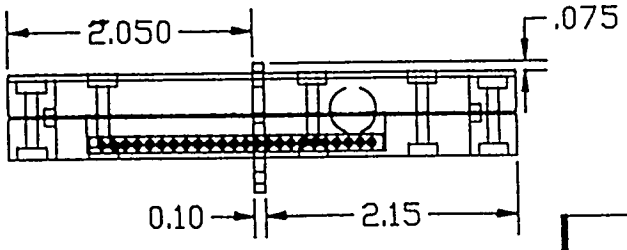


END VIEW

REV.	MF	DESCRIPTION OR MEMO NO.	FILE DATE	BY	ENGR	APVD
			APPROVALS			



3D VIEW



FRONT VIEW

REFERENCE DRAWINGS	NUMBER

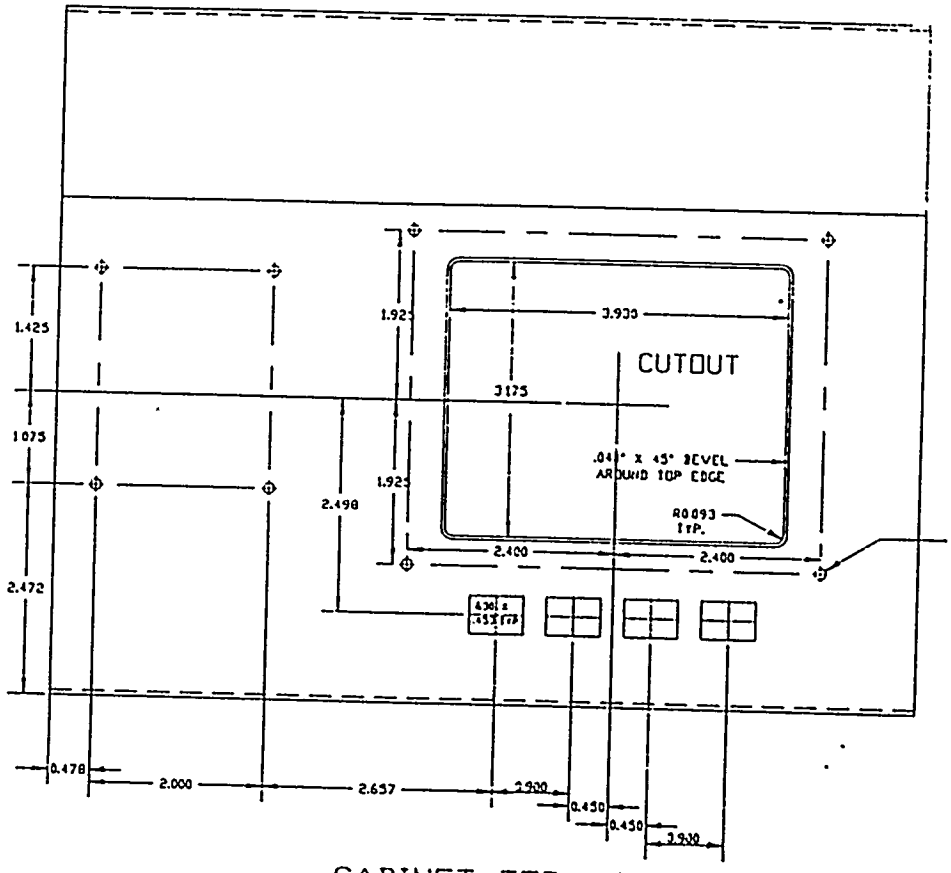
MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

TEST HARNESS HOUSING MECHANICAL DETAILS

BICRON INSTRUMENT TESTER

FILE DATE	FILE POINT HC- EC-	BLDG	SHT. 2 OF 2	TYPE MD	MF
SCALE	ID I&C	Q-6350-007			REV Ø

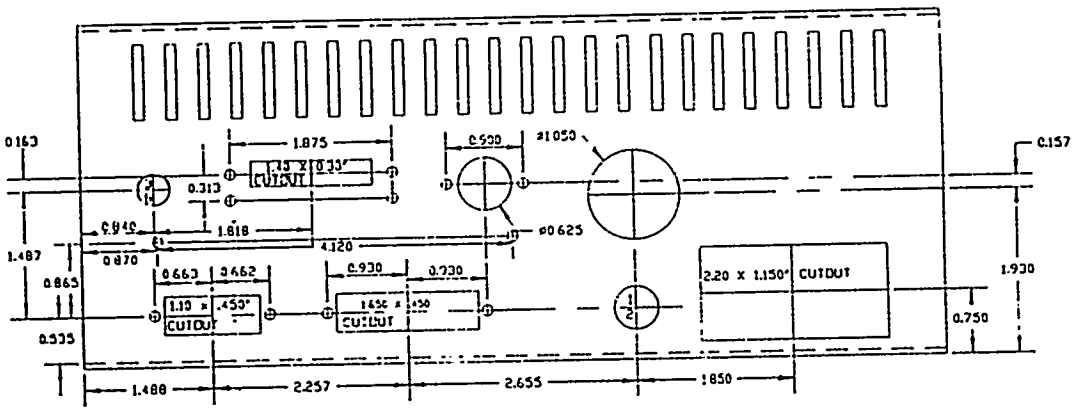
DRAWN BY		
	R.E COOPER	9/95
DESIGN CHECK		9/95
ENGINEER		9/95
APPROVED	D. a. Clayton	9/95
	RT Rushing	9/95
DRAWING APPROVALS		DATE



CABINET TOP VIEW

NOTES:
1. ENCL
4.0"
2. ENC
4.0"

REV.	BY	DESCRIPTION
		REVISION OR ISSUE



CABINET REAR VIEW

URE TOP IS HAMMOND #1456KG4 PVH: 10" WIDE, 7.2" DEEP
GH.
SURE CASE IS HAMMOND #1456KG4CWV, 10" WIDE, 7.2" DEEP.
GH.

REFERENCE DRAWINGS	

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.

HOUSING TOP
MECHANICAL DETAILS

BICRON INSTRUMENT TESTER

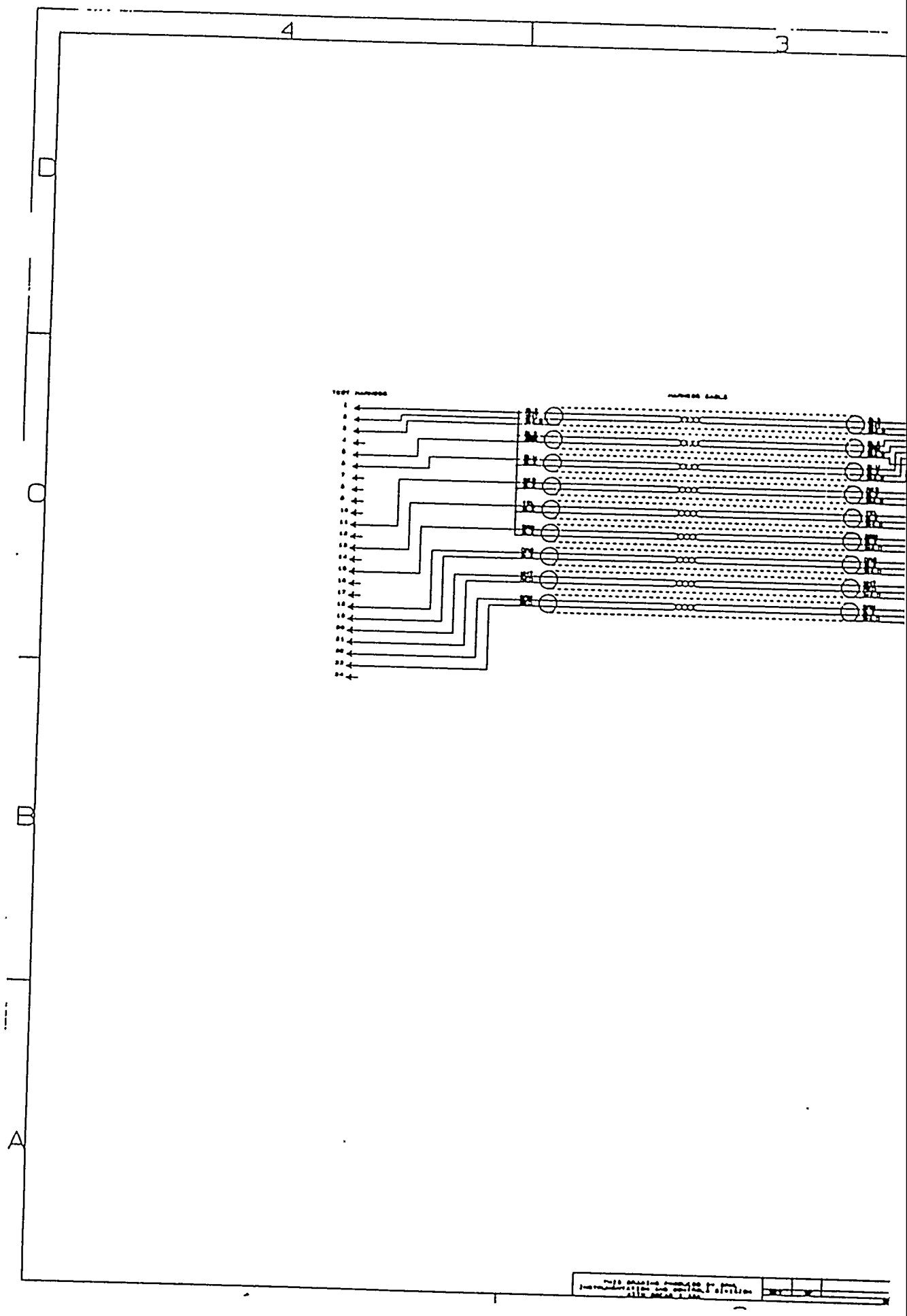
FILE DATE	FILE POINT	BLDG	SHT	OF
MAR 1 6 bsc	HC-1 EC-1	3500	1	1

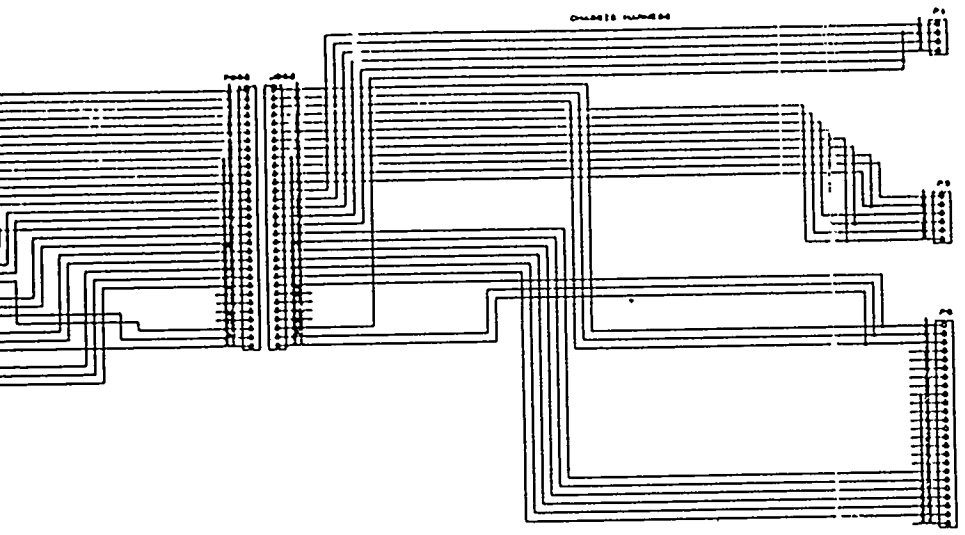
SCALE	ID		REV
FULL	1 & C	Q-6350-8	D

TOLERANCES UNLESS OTHERWISE SPECIFIED		R. A. MAPLES	3-91
FRACTIONS	± 1/64"		
.XX DECIMALS	± .015"		
.XXX DECIMALS	± .005"		3-11-92
ANGLES	± 0° 30'	E. Mattes	3-92
BREAK SHARP EDGES			
		DRAWING APPROVALS	DATE

	FILE DATE	BY	ENGR	APVD

APPROVALS


4
3
D
C
B
A



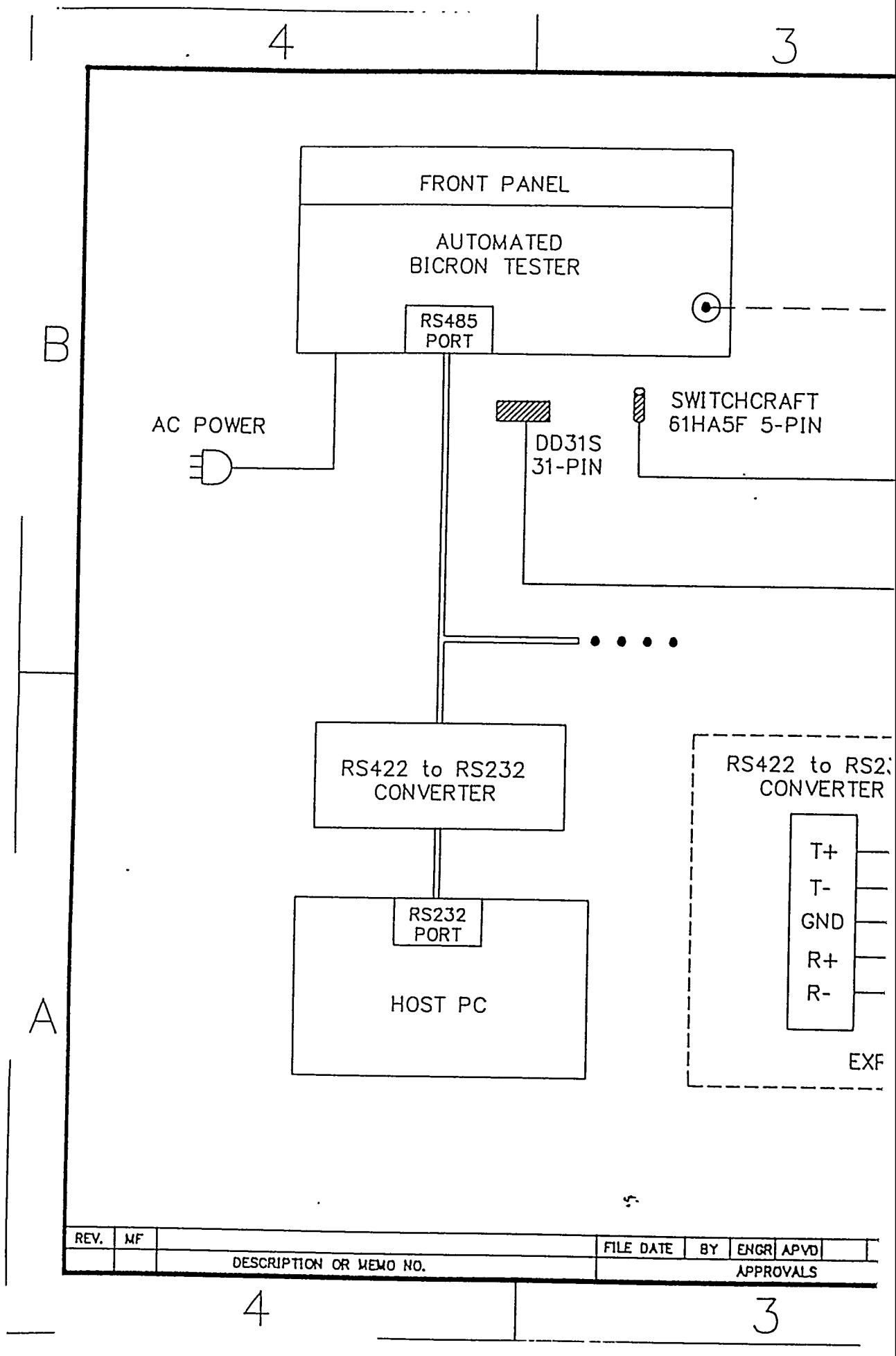
MARTIN MARIETTA

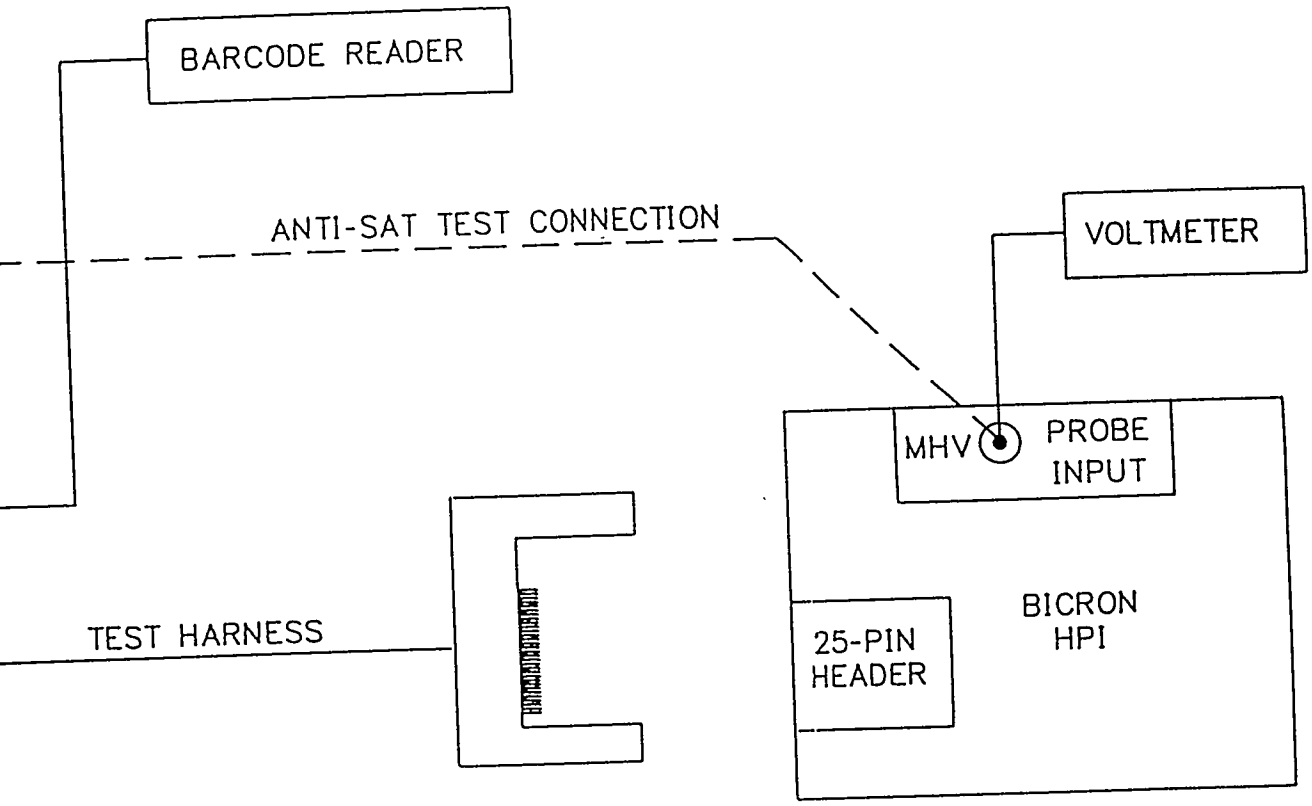
HARNESS CABLING
SCHEMATIC DIAGRAM

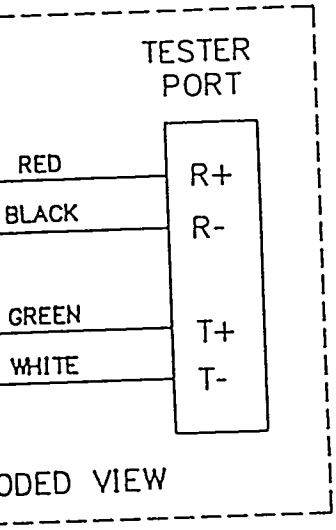
BICRON INSTRUMENT TESTER

HC-1 EC-1 3500

NIS I E C 0-6350-010 0


1
4
3
B
A
FRONT PANEL
AUTOMATED
BICRON TESTER
RS485
PORT
AC POWER
DD31S
31-PIN
SWITCHCRAFT
61HA5F 5-PIN
RS422 to RS232
CONVERTER
RS232
PORT
HOST PC
RS422 to RS2
CONVERTER
T+
T-
GND
R+
R-
EXF
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3





REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

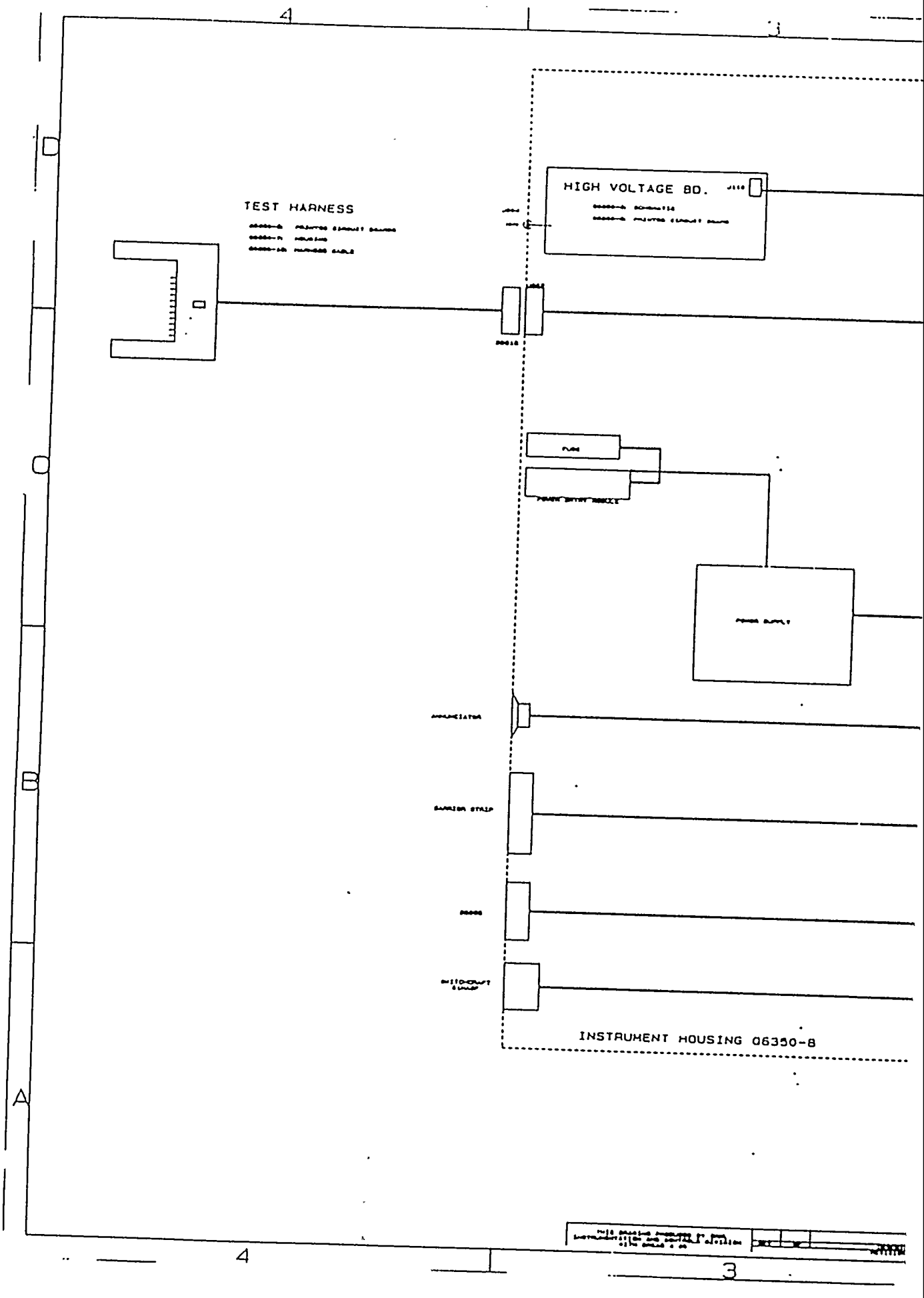
EQUIPMENT & RS-485 CONNECTION DIAGRAM

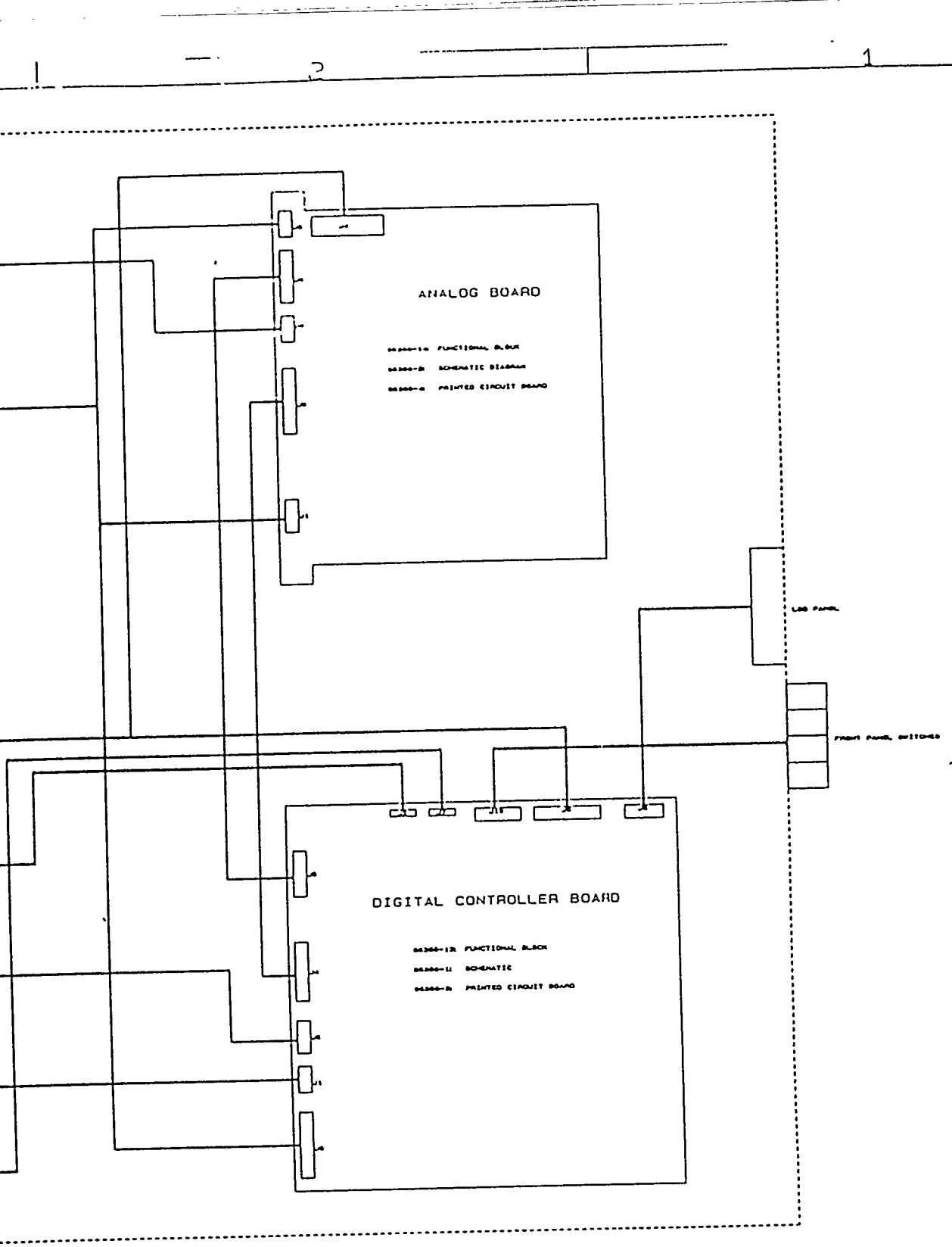
BICRON INSTRUMENT TESTER

TYPE: BD

DRAWN BY	R.E. COOPER	7/95
DRAWING CHECK		9/95
ENGINEER		9/95
APPROVED	J. A. Clayton	9/95
	R T Roseberry	9/95
DRAWING APPROVALS		DATE

FILE DATE	FILE POINT	BLDG	SHT. OF	M/F
	HC- EC-		1 1	
SCALE	ID	Q-6350-011		REV
	I&C			Ø


4
D
C
B
A
TEST HARNESS
HIGH VOLTAGE BD.
INSTRUMENT HOUSING 06350-8
4
3

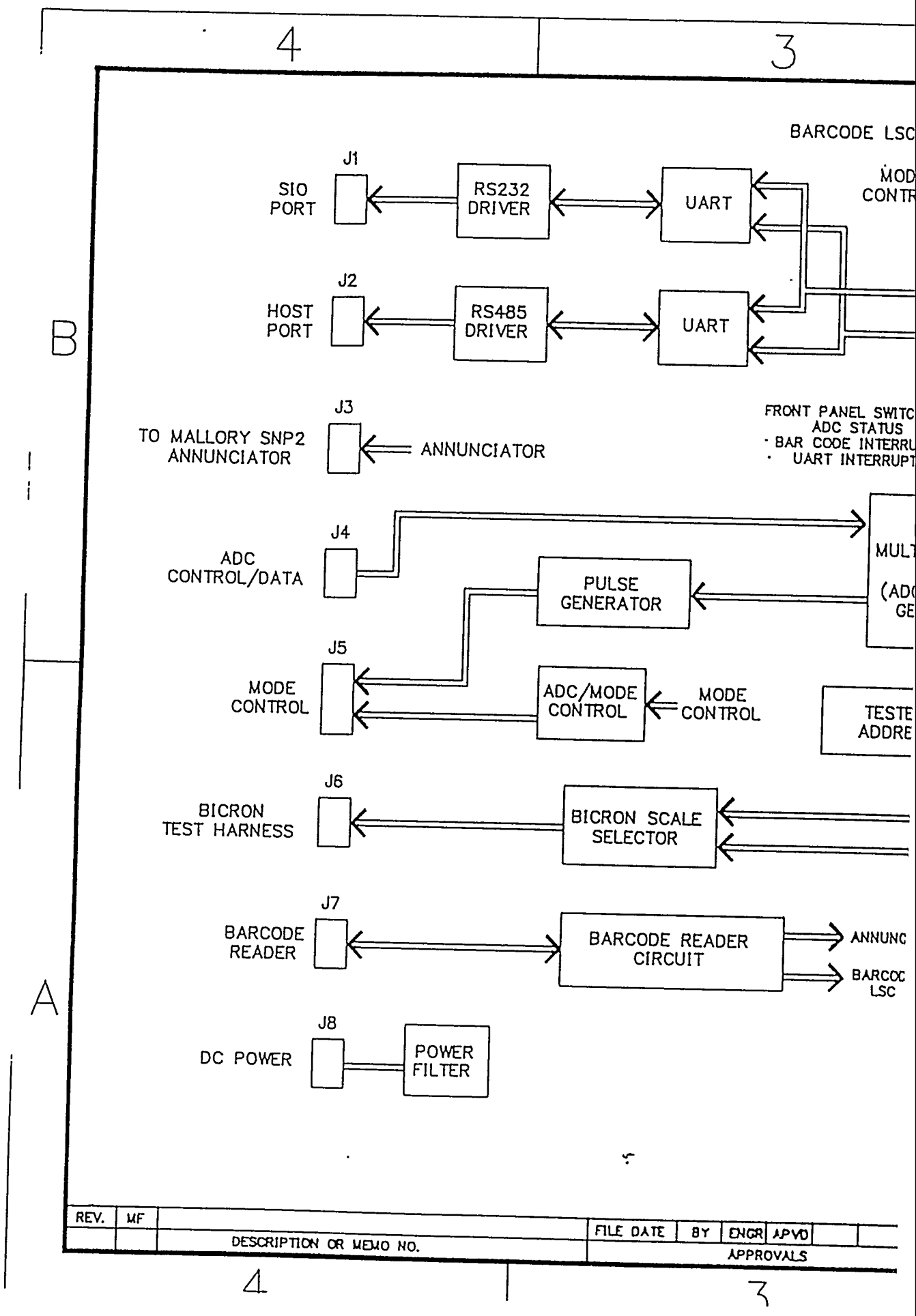


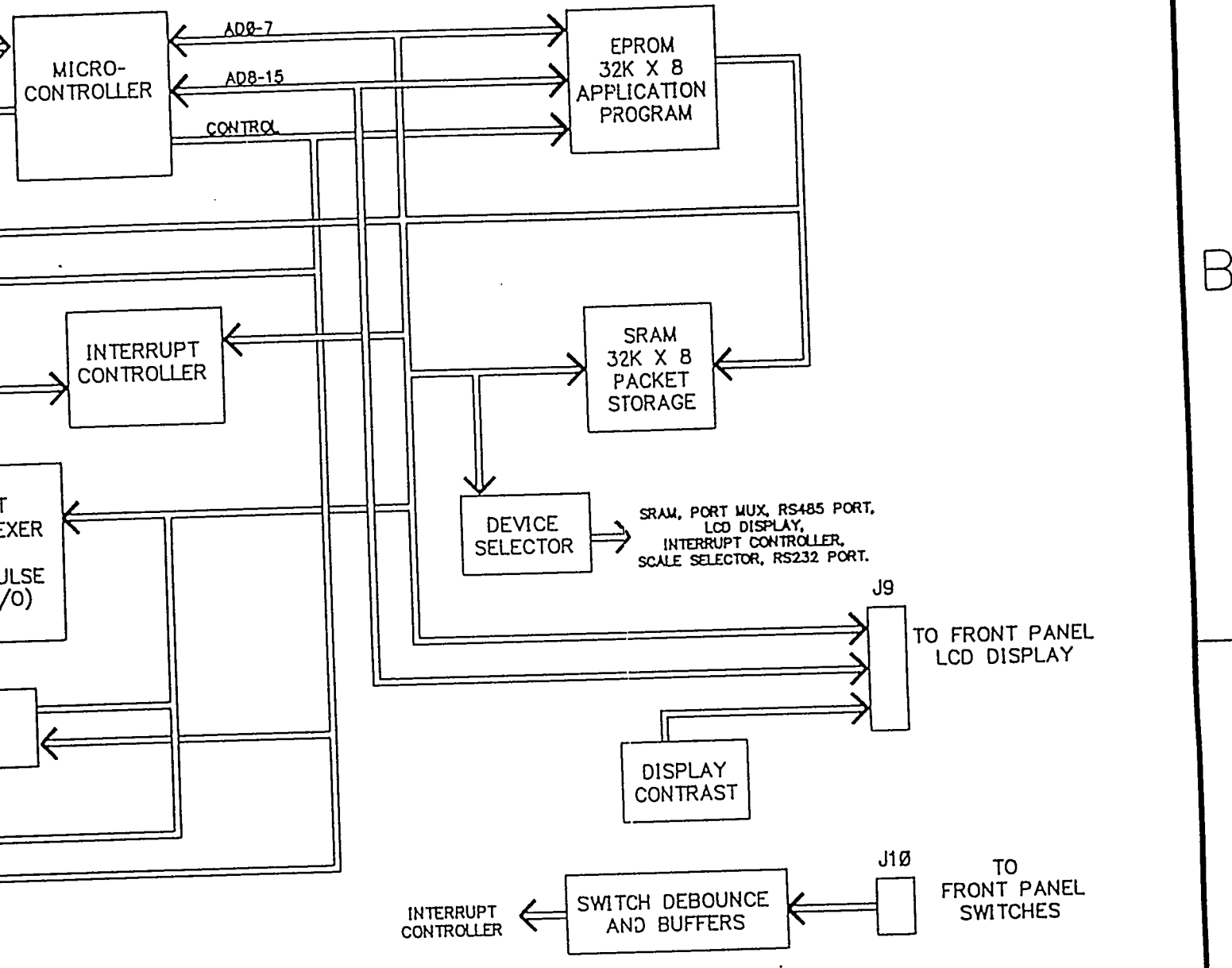
MARTIN MARIETTA

BOARD INTERCONNECTION
SCHEMATIC DIAGRAM

BICRON INSTRUMENT TESTER

SIZE	ID	0-6350-012	REV
NTS	I & C		0


4
3
B
A
BARCODE LSC
J1
SIO PORT
RS232 DRIVER
UART
MOD
CONTR
J2
HOST PORT
RS485 DRIVER
UART
J3
TO MALLORY SNP2 ANNUNCIATOR
ANNUNCIATOR
FRONT PANEL SWITC
ADC STATUS
· BAR CODE INTERRU
· UART INTERRUPT
J4
ADC CONTROL/DATA
PULSE GENERATOR
MULT
(ADC
GE
J5
MODE CONTROL
ADC/MODE CONTROL
MODE CONTROL
TESTE
ADDRE
J6
BICRON TEST HARNESS
BICRON SCALE SELECTOR
J7
BARCODE READER
BARCODE READER CIRCUIT
ANNUNC
BARCOC
LSC
J8
DC POWER
POWER FILTER
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3



REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA — MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

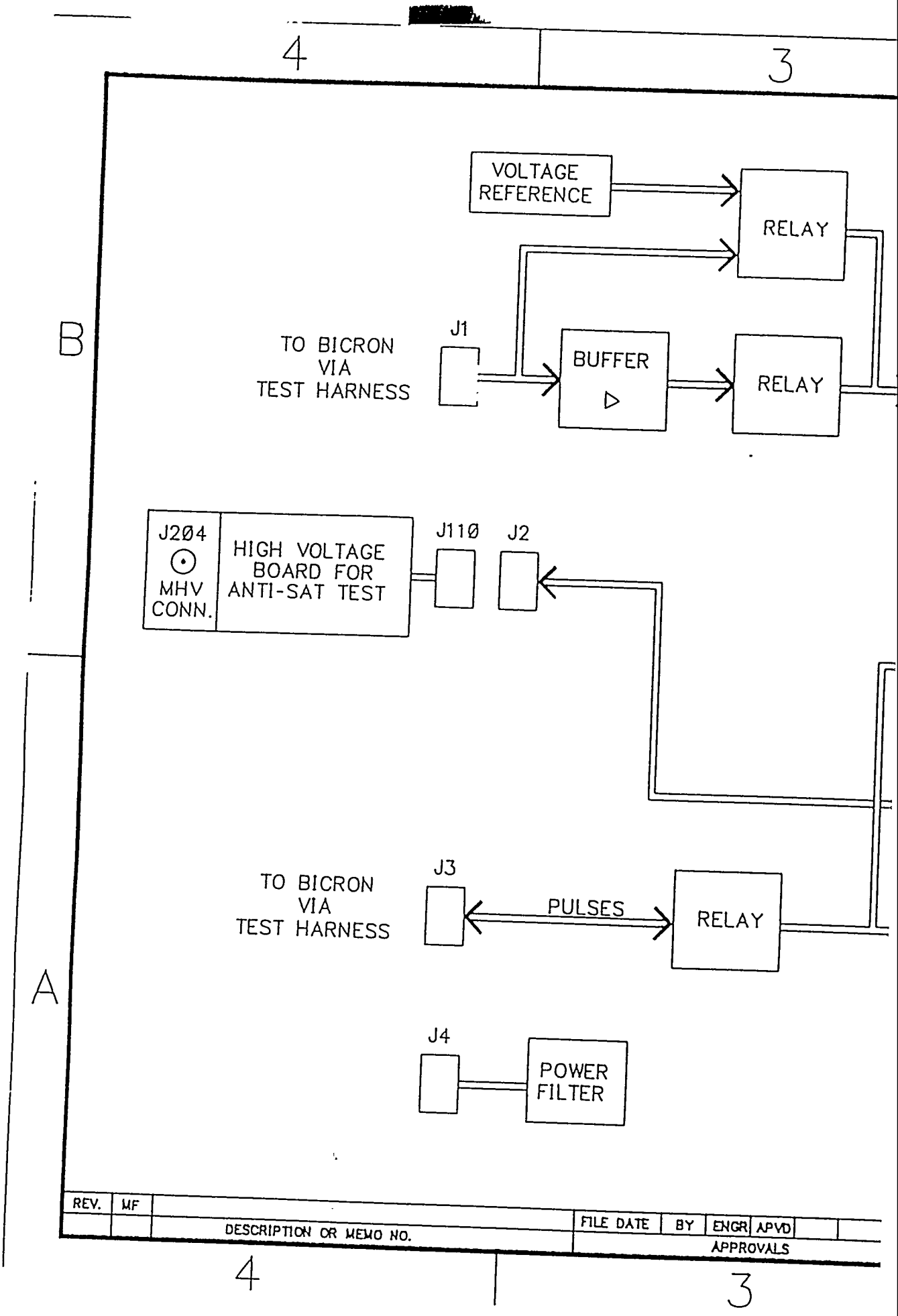
DIGITAL CONTROLLER BOARD FUNCTIONAL BLOCK DIAGRAM

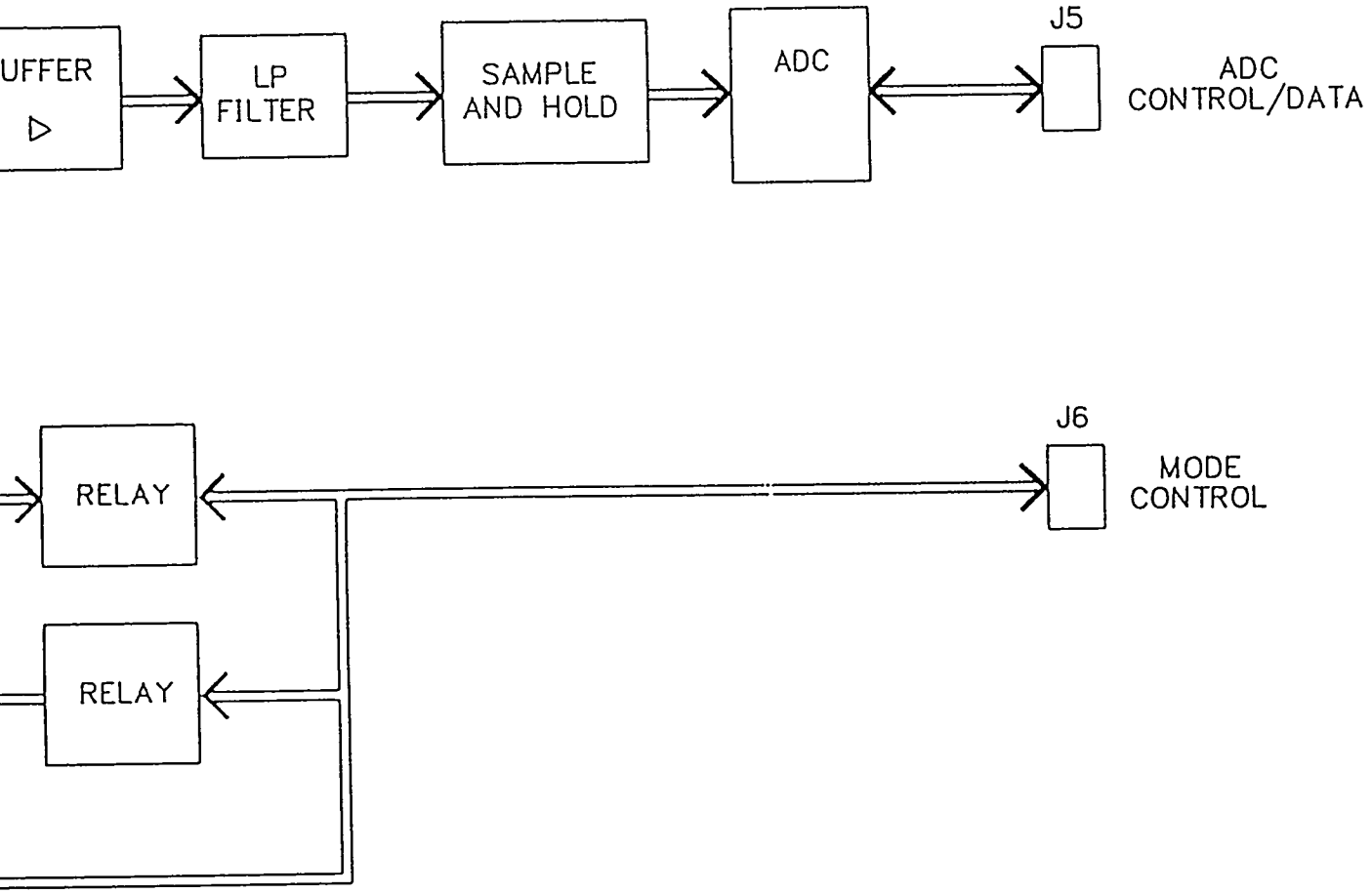
BICRON INSTRUMENT TESTER — TYPE: BD

DRAWN BY	R.E. COOPER	7/95
DESIGNER DIRECT		9/95
ENGINEER		9/95
APPROVED	D. A. Clayton	9/95
	R T Bracken	9/95
DRAWING APPROVALS		DATE

FILE DATE	FILE POINT	BLDG	SHT. OF	MF
	HC- EC-		1 1	

SCALE	ID		REV
	I&C	Q-6350-013	Ø


4
3
B
A
VOLTAGE REFERENCE
RELAY
TO BICRON VIA TEST HARNESS
J1
BUFFER
RELAY
J204
MHV CONN.
HIGH VOLTAGE BOARD FOR ANTI-SAT TEST
J110
J2
TO BICRON VIA TEST HARNESS
J3
PULSES
RELAY
J4
POWER FILTER
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3



REFERENCE DRAWINGS	NUMBER

MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC.
operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400
Oak Ridge, Tennessee Paducah, Kentucky

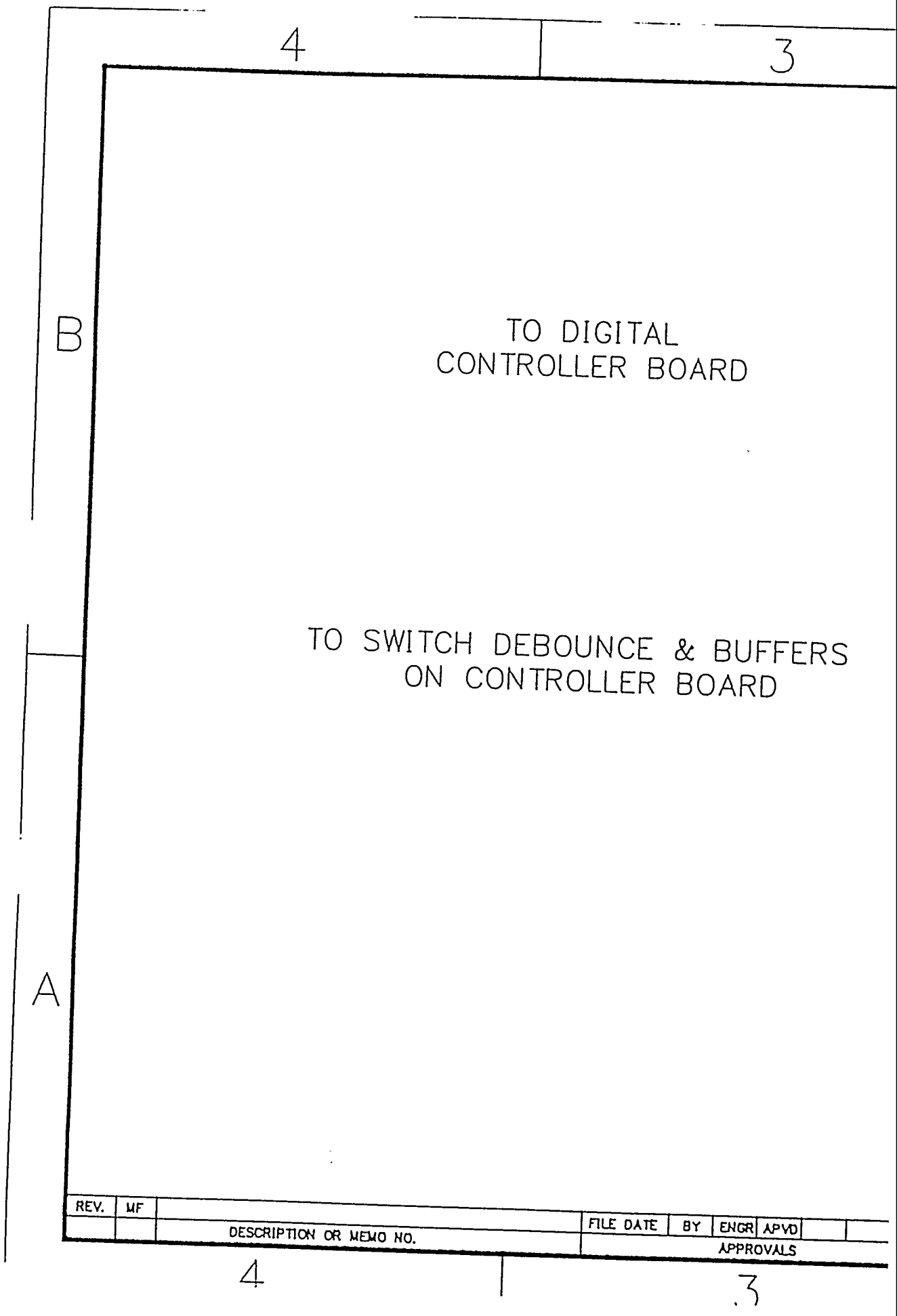
ANALOG BOARD FUNCTIONAL BLOCK DIAGRAM

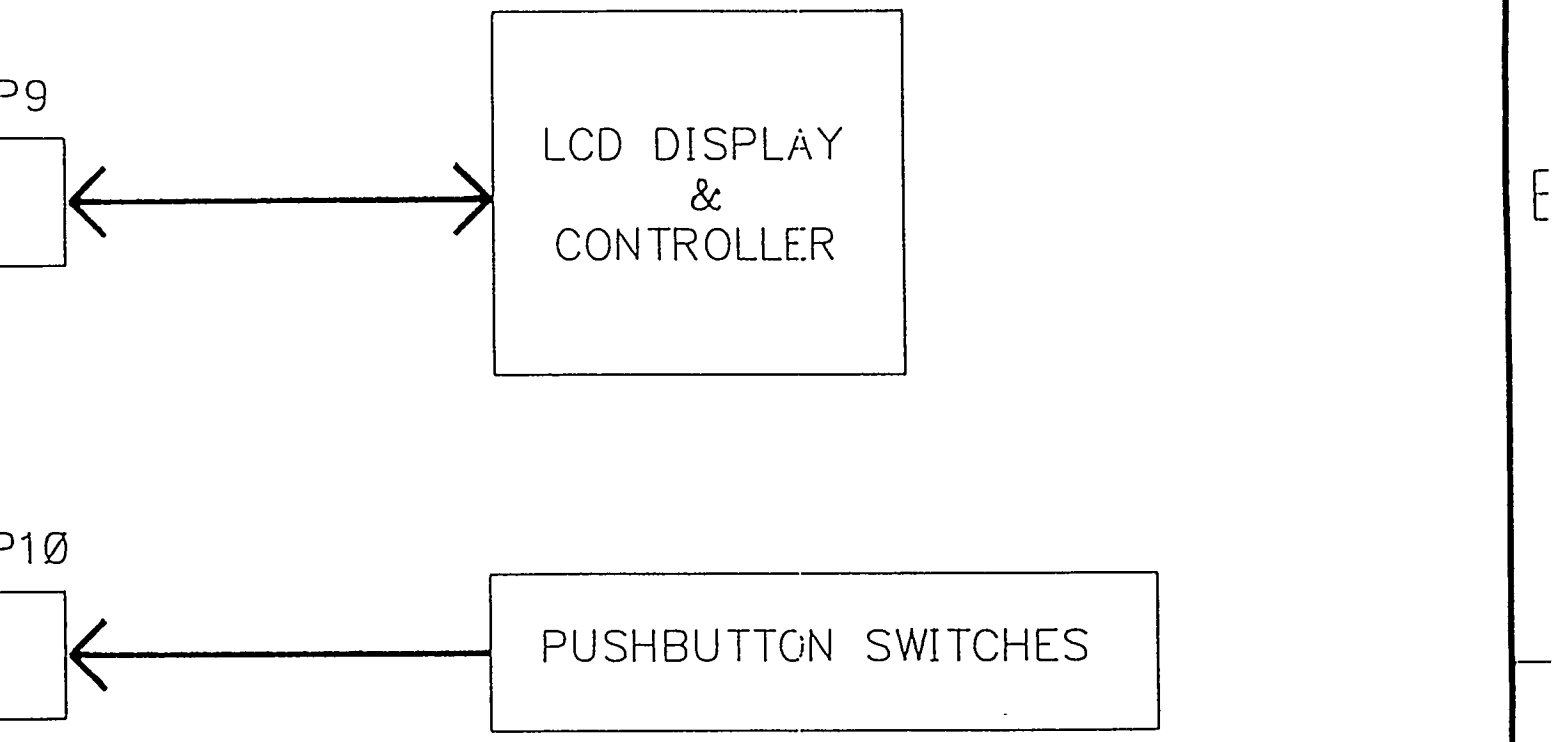
BICRON INSTRUMENT TESTER — TYPE: BD

FILE DATE	FILE POINT	BLDG	SHT. OF	WF
	HC- EC-		1 1	

SCALE	ID		REV
	I&C	Q-6350-014	Ø

DRAWN BY	R.E. COOPER	7/95
DESIGN CHECK		9/95
CONCUR		9/95
APPROVED	A. Clayton	9/95
		9/95
DRAWING APPROVALS		DATE


4
3
B
TO DIGITAL
CONTROLLER BOARD
TO SWITCH DEBOUNCE & BUFFERS
ON CONTROLLER BOARD
A
REV.
MF
DESCRIPTION OR MEMO NO.
FILE DATE
BY
ENGR
APVD
APPROVALS
4
3



REFERENCE DRAWINGS	NUMBER
MARTIN MARIETTA MARTIN MARIETTA ENERGY SYSTEMS, INC. operated for the DEPARTMENT OF ENERGY under U.S. GOVERNMENT contract DE-AC-05-84OR21400 Oak Ridge, Tennessee Paducah, Kentucky	

OPERATOR INTERFACE FUNCTIONAL BLOCK DIAGRAM

BICRON INSTRUMENT TESTER					TYPE BD
FILE DATE		FILE POINT HC- EC-	BLDG	SHT. 1 OF 8	WF
SCALE	ID I&C	Q-6350-015			REV Ø

DRAWN BY	R.E. COOPER	7/95
DRAWING CHECK		9/95
ENGINEER		9/95
APPROVED		9/95
		9/5.5
DRAWING APPROVALS		DATE