

To the Graduate Council:

I am submitting herewith a thesis written by Richard H. Cox entitled "Development of a Final Manufacturing Test for a Resolver Position Tracker." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science with a major in Electrical Engineering.

Donald W. Bouldin

D. W. Bouldin
Major Professor

We have read this thesis
and recommend its acceptance:

Robert E. Balducci

Marshall Pace

Accepted for the Council:

Cuminski

Vice Provost
and Dean of The Graduate School

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at The University of Tennessee, Knoxville, I agree that the Library shall make it available to borrowers under rules of the library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgement of the source is made.

Permission for extensive quotation from or reproduction of this thesis may be granted by my major professor, or in his absence, by the Head of Interlibrary Services when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Signature

Date

Richard O. H. Cooper
March 14, 1990

DEVELOPEMENT OF A FINAL MANUFACTURING TEST FOR A
RESOLVER POSITION TRACKER

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Richard H. Cox, Jr.

May 1990

ACKNOWLEDGEMENTS

The author wishes to thank the members of the thesis committee for their time and consideration of this thesis. The author wishes to express additional appreciation to Dr. Bouldin for his efforts in scheduling the final presentation and interfacing with others on campus.

ABSTRACT

Before delivery of an electronic assembly to a customer, a means of testing must be developed to insure that the assembly is working as specified prior to shipment. At Texas Instruments's (TI) Industrial System Division, electronic assemblies are tested on a Final Test Station to verify the functionality of the device before shipment. The Resolver Position Tracker (RPT) is a microprocessor based device which is used to monitor resolver positions on a robot arm. This device is being manufactured at TI's plant in Johnson City, Tennessee.

The test station that is used for the final testing of RPT is also used for the final test of other products in a family of programmable robot controllers. For this reason, a custom interface is required to interface the device being tested to Final Test Station. In addition to providing connections from general purpose commercial test equipment such as digital voltmeters, function generators, and power supplies, this interface has to generate signals that are particular only to the RPT. Control of the commercial test equipment and circuitry in the interface is done by software designed specifically for the test of the RPT. This test station is the final operation in the manufacturing process of the RPT module before shipment to the customer.

TABLE OF CONTENTS

CHAPTER	PAGE
1. INTRODUCTION	1
A. PURPOSE OF TESTING	1
2. DESCRIPTION OF THE RESOLVER POSITION TRACKER . . .	4
A. RPT IN A ROBOT SYSTEM	4
B. RPT FUNCTIONS	6
C. RPT COMMANDS	9
3. TEST PHILOSOPHY	14
A. GOAL OF TESTING	14
B. TEST SEQUENCE	15
4. TEST STATION HARDWARE	26
A. EQUIPMENT RACK	26
B. RPT TEST HEAD	28
C. RS232 TO RS485 CONVERTOR	31
D. RESOLVER SIMULATOR	33
5. TEST SOFTWARE	43
A. TEST STATION SOFTWARE	43
B. INDIVIDUAL TEST PROGRAMS	44
6. CONCLUSION	57
A. REVIEW OF GOALS	57
B. THE RESOLVER SIMULATOR AS A MOTOR SIMULATOR . .	60
LIST OF REFERENCES	64
APPENDIX: SCHEMATICS, SOFTWARE	66
VITA	91

LIST OF TABLES

TABLE	PAGE
2-1. ASCII commands and their function	9
4-1. Bit weight in degrees	38
6-1. Binary quadrant values	60

LIST OF FIGURES

FIGURE	PAGE
2-1. RPT in a robot system	6
2-2. Binary message format for the RPT	10
2-3. Binary commands for the RPT	10
2-4. System level status indications	11
2-5. RPT position data	13
4-1. Commercial test equipment	27
4-2. Bus requirements	30
4-3. RS232 to RS485 convertor	32
5-1. Individual test programs	44

CHAPTER 1

INTRODUCTION

A. PURPOSE OF TESTING

The sole purpose of testing electronic assemblies should be to determine if the unit performs according to established specifications.[1] An important reason for testing is to assure that the customer receives a quality product. The cost of quality due to insufficient testing can be lost manufacturer credibility. The cost of quality at the time of test development is more than just the development cost of the test hardware and software. The actual usage of the test equipment, and the maintenance of the test equipment must be considered. A balance between a reasonable cost in testing and the delivery of a quality product must be considered the development of the final test of any assembly.

A primary goal of this final test will be to keep test time to a minimum. This goal will be obtained by scrutinizing each test to verify its necessity, combining tests of different functions where possible, and reduce test coverage if test time is excessive. Incremental increases in test coverage may not increase the overall product quality enough to justify the additional test time.

One way for the test engineer to reduce costs is to work in conjunction with design engineers during early prototype testing. By working on test development early in the design

phase, errors in the design can be found and corrected while it is relatively inexpensive to do so.[2] Early involvement in the design phase also aids in the transition from prototype to pilot-build to final production of a product by allowing for a quicker means of determining the functionality of newly built units.

Testing of RPT assemblies will be accomplished in three stages. After the circuit boards have been built, they are placed on a "bed of nails" fixture for individual component testing using a GenRad model 2270 tester. Passive components are tested to the extent possible against their expected values. Active components, such as logic devices, will be tested against a truth table for the particular device. More complicated devices, such as microprocessors, are checked only for proper insertion. The GenRad tester is also used to check for improper opens and shorts on the circuit board. Testing the RPT on the GenRad tester also helps diagnose problems that could possibly damage the final test head. The second stage of testing involves running the assembled circuit board in a powered-up condition in an oven that cycles temperature between 0 and 60 degrees Celsius over a period of time. This test is used to induce premature failure of any marginally functioning components. The third stage of testing involves the final test which this thesis details.

Chapter 2 of this thesis explains the purpose of the RPT and its primary features. Chapter 3 details the reasoning for

what needed to be tested on the RPT and the method of testing the board. Chapters 4 and 5 are concerned with the actual implementation of the final test hardware and software for the RPT. Chapter 6 consists of a conclusion about the project with a brief discussion of other uses of equipment developed for this project.

CHAPTER 2

DESCRIPTION OF THE RESOLVER POSITION TRACKER

A. RPT IN A ROBOT SYSTEM

The main purpose of the RPT module is to monitor changes in resolver position during power loss on a robot arm.[3] A resolver is a device used to measure shaft position of devices such as motors and radar dish gimbals. The customer desired this function so that if, for instance, the plant were to be shut down or momentarily lose power, restoration of the robot to the position before the shut down could be done automatically. The servo controller controls and monitors the position of the robot while power is on. During this time, the RPT monitors only the resolver signals for position information and actual integrity. Upon loss of main power, the RPT monitors the resolver position and stores any changes that occur while the module itself is powered from a battery. When main power is restored, the servo controller interrogates the RPT via serial data lines to determine if any movement occurred during the power loss and if any faults are detected. The servo controller can take this new robot position along with the position before power loss and move the servo motors on the robot to their proper position.

A secondary function of the RPT module is to allow the servo controller to determine to which type of device it is connected. Data giving the characteristics of a given robot

system is previously stored in nonvolatile memory(EEPROM) on the RPT module. The servo controller can interrogate the RPT to determine the particular characteristics of the robot and execute motion control instructions accordingly.

The RPT is installed on the robot whose resolvers it will be monitoring. The main power for the module is provided by a DC power supply with backup power coming from a 6 volt battery. Each servo controller in the product line can control a robot with up to eight axis of movement. In turn, each RPT can monitor eight resolvers. The reference output from the servo control module, which is a sine wave, is input to the reference input of the RPT. The reference signal passes through detection circuitry on the RPT module and is output to the reference inputs of the resolvers on the robot. The sine and cosine outputs of each resolver act as inputs to the RPT module. These signals are monitored by the RPT and feed through the RPT to the servo controller. Figure 2-1 shows a block diagram of the basic robot control system with the RPT installed. The circles with the letter "R" and a number represent resolvers on a robot arm.

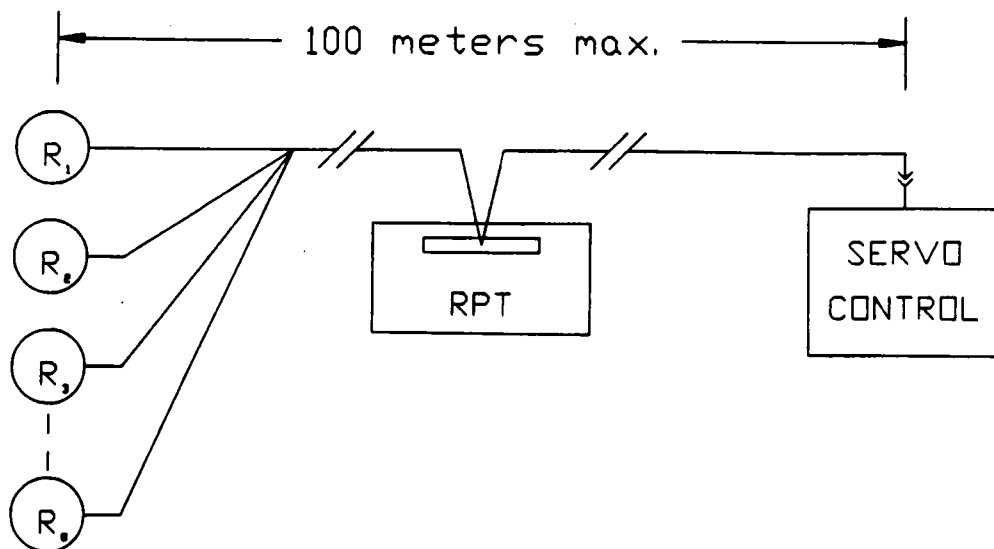


Figure 2-1. RPT in a robot system.

B. RPT FUNCTIONS

Initial conditions for the resolver positions can be programmed either by ASCII commands from a nonintelligent terminal or by binary commands from the servo control module. The RPT can be interrogated for the status of the RPT itself, the resolvers, or any illegal initialization that might occur. Characteristics of the robot can be programmed into the RPT's EEPROM memory by binary commands for later access by the servo controller. Characteristics may include the number of resolvers mounted on the robot, the type of robot, or the type of resolver used in the robot system.

The RPT constantly monitors resolver movement while power

is supplied by the main power source. The RPT module has resolution capability only to the quadrant. It tracks changes in quadrants and revolutions of the resolvers from the programmed initial conditions.

Upon loss of main power or loss of the resolver reference signal from the servo controller, the RPT starts generating the reference for the resolvers in the form of a periodic pulse and continues to monitor the sine and cosine returns from the individual resolvers. The pulse repetition rate(PRT) of the pulse depend on whether or not motion is detected during the loss of power or reference. If no motion is detected, the module goes into a "sleep" mode to conserve battery power and pulses only the resolvers at a PRT set by the user. This is performed periodically to determine if any movement has occurred. If a quadrant change is detected, it will go into a fast pulse mode and stay in this state until a period of no motion is again detected. When main power is restored to the RPT and a reference pulse is detected from the servo control module, the module will return to a passive mode of monitoring the movement of the resolvers under control of the servo controller. At this point, the servo control module can interrogate the RPT to determine whether any movement occurred during power loss. This information from the RPT is used to return the robot to the last position before the power loss.

The RPT can also monitor the status of the resolvers for

opens and shorts, illegal quadrant movement, illegal initialization, or the condition of its power sources and report this status to the servo controller or an operator with a nonintelligent terminal.

The actual features of the module play an important part in the definition of the test equipment requirements. The equipment rack for the test station contain general purpose test equipment that is used to test the RPT module as well as other products associated with the robot control equipment being manufactured. Any hardware that is required only for the RPT should be placed in the test head.

The RPT receives its main power from a fixed DC power supply. The unit is capable of operating from 17 volts up to 36 volts with a nominal operating level of 25VDC. Battery power is supplied by a 6VDC rechargeable battery. The RPT monitors the battery voltage level and will report a battery low condition if the voltage drops below 5.8VDC. If the battery voltage drops below 5.4VDC, the RPT will shut down to avoid completely draining the battery. Testing of the RPT requires at least one fixed power supply and one variable supply.

The RPT uses the IEEE 485 serial communications protocol. Each module has one transmit line pair and one receive line pair with termination resistors that can be switched in if needed. The RPT uses two positions of a DIP(dual inline package) switch for the four baud rates (1200, 2400, 4800,

9600). In the robot control system, various units have a unique address on the IEEE 485 communications lines. The address of RPT modules can be set to a hexadecimal address of 1E or 1F by means of a DIP switch. Communications to and from the unit under test(UUT) will be provided by a serial communication card in the 990 computer with a circuit in the test head to convert to and from the RS232 and RS485 protocols.

C. RPT COMMANDS

The RPT uses a separate group of commands for binary and ASCII communication. ASCII commands allow for a simple method to initialize and monitor the RPT once it has been installed via a nonintelligent terminal. Table 2-1 lists the valid ASCII commands for the RPT.

Table 2-1. ASCII commands and their function.

COMMAND	FUNCTION
PPOS <argument>	Preset Positions
SPOS	Send Positions
SSTA	Send Status
CSTA	Clear Status
SETR <argument>	Set Sample Rate
EDIA	Execute Diagnostics

The PPOS and SETR commands require an argument data string along with the command itself. The PPOS command is used to initialize the positions of the eight resolvers. If a position is not used, it is initialized with an "X". The SETR

command has an accompanying argument that varies from 2.5 to 320.

The second communications protocol for the RPT is the binary protocol. This protocol is used for communications between the servo controller and the RPT and provides a maximum throughput between the two devices. Figure 2-2 gives the binary protocol format.

<DLE-STX> <ADDRESS> <LENGTH> <DATA> <BCS> <DLE-ETX>

DLE	data link escape
STX	start of text
ADDRESS	address on the RS485 multiple drop line
LENGTH	length of transmission in bytes
DATA	binary command plus associated data
BCS	block check sum
DLE	data link escape
ETX	end of text

Figure 2-2. Binary message format for the RPT.

The binary protocol contains all the commands used in the ASCII plus additional commands only used by the servo controller. Figure 2-3 lists the valid binary commands for the RPT.

Command Character	Command
D (ASCII \$44)	Preset Positions
P (ASCII \$50)	Send Positions
S (ASCII \$53)	Send Status
C (ASCII \$43)	Clear Status
U (ASCII \$55)	Clear Positions Presets
R (ASCII \$52)	Set Sample Rate
A (ASCII \$41)	Write EEPROM Data
B (ASCII \$42)	Read EEPROM Data
E (ASCII \$45)	Execute Diagnostics

Figure 2-3. Binary commands for RPT.

Binary commands are only sent out one at a time. Reading and

writing to EEPROM memory and verifying the RS485 addressing of the RPT can only be accomplished using binary communication. This will require the use of binary communications during the final test.

There are several error and status conditions that are monitored by the RPT and reported by the Send Status commands. The error and status indications are broken down into two groups. The first group shown in Figure 2-4 give an overview of the system level status.

RESOLVER PRIMARY SHORT
180 DEGREE PRESET ERROR
RESOLVER SECONDARY OPEN/SHORT
BATTERY LOW
MOTION DURING POWER LOSS
RPT HARDWARE ERROR
ILLEGAL QUADRANT SEQUENCE
RPT RESET AFTER LAST PRESET

Figure 2-4. System level status indications.

Additional channel status is provided for the RESOLVER SECONDARY OPEN/SHORT and ILLEGAL QUADRANT SEQUENCE conditions to aid in debugging a system.

The common resolver reference from the servo controller passes through a detection and switching circuit on the RPT and then to each resolver on the robot arm. This circuit monitors the reference for proper voltage and frequency as well as for shorts between the RPT and the primary windings on the resolver. The RPT is capable of monitoring a 5.6 volt peak to peak, 4880 Hz resolver reference signal. If the reference signal falls out of the acceptable range in either

voltage or frequency, the RPT switches from the servo controller's reference to a reference signal generated by the RPT. The reference signal generated by the RPT in the active mode is a 50 microsecond(μ S) pulse with a PRT of 1 millisecond(mS). The PRT of the pulse during the inactive or "sleep" mode is dependent on initial conditions set up by the user (2.5mS to 320mS). This RPT generated signal provides the stimulus to the primary windings of up to eight resolvers. If the RPT detects motion during the loss of the resolver reference, it pulses the resolver at a 1mS PRT and monitors the sine and cosine signals from each of the attached resolvers. At a maximum tracking rate of 7000 RPM, which gives a elapsed time of 2.14ms in each quadrant, this guarantees that the RPT will not miss movement from quadrant to quadrant. If no quadrant movement is detected the RPT will revert to an inactive "sleep" mode. In this mode, the RPT will pulse only the resolver according to the preset value. It is up to the robot system designer to determine an appropriate inactive pulse PRT for a given situation. When motion is again detected, the RPT will go into the active mode again. Testing of the various pulse widths and PRTs will be accomplished by a timer/counter in the test equipment rack.

The RPT is capable of monitoring the sine and cosine returns from up to eight resolvers. The RPT is capable of resolution only down to the quadrant. The initial quadrant location and rotation information is entered by a

nonintelligent terminal or by the servo controller. An input channel is enabled by programming an initial quadrant and rotation condition. The RPT is initialized with quadrant locations 0 through 3 and rotational counts from 0 to 16384. The format of the position of each of the resolvers is given in Figure 2-5. Position tracking is verified under various test conditions.

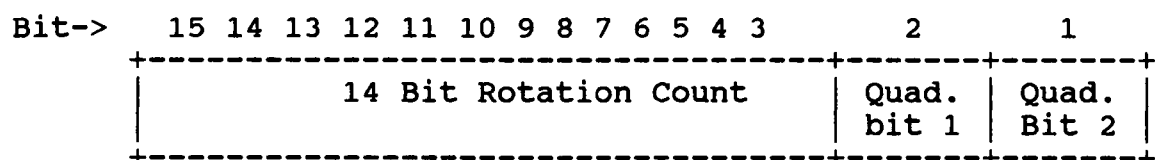


Figure 2-5. RPT position data.

CHAPTER 3

TEST PHILOSOPHY

A. GOAL OF TESTING

Ideally, it is desirable to perform a 100% functional test on the UUT before shipment. For economic reasons, this is not always practical. After a certain point, the cost for additional test coverage does not justify the incremental step in total system quality. To achieve a goal of maximum test coverage in a minimal amount of test time, each test is scrutinized during the definition and design phase. During the early stages of the design cycle, the test engineer and design engineer must work together and agree on a level of testing. This is formally accomplished during periodic design reviews.

To avoid wasting test station time and possible damage to a UUT or test head, consideration must be given as to how the RPT will be tested. If the power supply is nonfunctional, then other functions that will be tested will probably fail also and proceeding with further testing would be a waste of time. An input signal shorted to ground may cause damage to the test head circuitry in addition to causing misleading test results.

The lowest level of functionality of the board is the power supply. By using a current-limited DC power source and measuring the voltage drop across a sensing resistor,

excessive variances in the expected current draw of a UUT can be checked with limited risk of damage to the UUT or test station. After it has been determined that the unit is drawing a proper amount of current from the power source, calibration will be performed. Most of the remaining testing of the RPT will require interactive communications between the UUT and the test station so that this function will be tested next. The user accessible memory test is performed before any further testing is done so if the operating system is inadvertently written to, the normal functioning of the code should indicate a problem with the EEPROM. A status check of the checksum is also performed as a double check. Verifying the proper operation of the UUT while supplied from the main power source and the battery is done after the communications section is deemed to be functional. The load carrying capability of the resolver primary is tested while making sure that the circuit does not switch to the pulsing mode operation. The ability of the module to track changes in quadrant movement and resolver rotations will also be checked while the UUT is operating from the main power supply. After functionality of the board has been verified using the main power source, the ability to detect motion during operation on battery power will be verified.

B. TEST SEQUENCE

The power for the RPT is supplied primarily from a 24VDC

service power supply during normal operation. In a typical application, if 24VDC is lost, the RPT switches over and operates from a 6VDC battery. If the RPT is at least functioning marginally, it will draw approximately 150mA from the 24VDC power supply. The RPT is also expected to operate over a range of 17VDC to 36VDC from the 24VDC service power supply so the current draw at both extremes is tested. Two options for testing the current draw of the RPT under test are measuring the voltage drop across a resistor that is in series with the RPT and the power supply or by taking a direct current reading using the digital multimeter (DMM) in series with the RPT and power supply. The meters commonly used in test stations at TI can be damaged by a RPT with a short in the DC to DC convertor so a power resistor with a high enough wattage rating to withstand a dead short at 36VDC is used. The battery input circuitry first required calibration of the charging section. The calibration of the battery circuit proves both the initial proper operation of the circuit as well as setting the output battery charging voltage of the RPT. At this point, the ability of the RPT to supply sufficient current to charge a low battery will be tested.

To prolong the life of relays in the relay matrix box, consideration is given as to how and when these relays are used. The general rule is not to open or close these relays with power applied to them. For example, a current measurement is performed by closing all applicable relays,

powering up the unit, then taking a reading. Using this same procedure after the reading is made, the UUT is powered down, then relay connections are broken. When a relay must be opened or closed with power applied, power relays external to the relay matrix box are used whenever possible. These power relays have sturdier contacts and can withstand the arcing that occurs when they are switched with power present.

Communications circuitry will be the next part of the RPT to be tested. The IEEE-485 serial port communication needs to be verified since the RPT basically acts as a monitoring device and it must be interrogated for information to be used by the servo control module or by a user interface to a nonintelligent terminal.

There are two DIP switches used in the communications circuit. One DIP switch is used to switch in and out 120 ohm termination resistors. The second DIP switch is used to select one of four baud rates and the address of the RPT on the IEEE-485 serial channel. The DIP switch used to select termination resistors has four switches and will be tested in four positions: ALL OFF, ALL ON, and two alternating OFF and ON patterns. This is done because two switches are needed to switch in a termination resistor. A DMM is used to measure the resistance between the transmit and receive wires which are an open circuit except when all switches are ON. These measurements are made with power removed from the RPT. For the second DIP switch that controls the baud rate and the RPT

address the four switches are set to ALL ON and ALL OFF only. In the ALL OFF position, the RPT communicates at 9600 baud at address 1F on the IEEE-485 serial line. In the ALL ON position, the RPT will reside at address 1E and communicate at 1200 baud. There is one position of the DIP switch that is not used. If there are any stuck at one or zero conditions, the board will not respond to the command due to improper addressing or to an incorrect baud rate. In the case of the DIP switch for the termination resistors, a shorted position would be missed by an ALL ON and ALL OFF only test. The status of the module will be checked at each baud rate. If the module communicates and the RPT operating system diagnostics pass the checksum test, it can safely be assumed that it will communicate at the 2400 and 4800 baud rates selected by the second DIP switch. The reason for such a detailed test of the DIP switches is that they are historically problem devices.

If the RPT passes the communications test, the microprocessor is functioning at a level that allows further testing of the unit. The operating system of the RPT is resident on a single EEPROM at addresses hexadecimal 0 through 7FFF. There is also 16384 bytes of user-accessible memory in this same EEPROM which can only be written to or read from using the binary communications protocol. This user memory is located at addresses hexadecimal 8000 through BFFF in the EEPROM. The RPT has a time out feature on the communications

operation which prevents sending a binary protocol message of over 53 bytes in length. Software restrictions in the communications software for the 990 computer restrict the binary protocol messages even further to 22 bytes in length. Due to time delays in the communications software for the 990 and the number of messages required to send enough data to fill the RPT's EEPROM memory, only one address will be written to. This memory location is address 8000. The most significant address bit of the EEPROM is the only bit not used by both the RPT's operating system software and the user-accessible area. All other address lines will be toggled during the memory accesses done by the microprocessor in normal operation. If the most significant address bit is stuck low, the test data is written on top of the operating system. Further functional testing of the RPT should catch this failure.

The RPT is placed in line between the servo controller and the resolvers on the robot arm. The board is designed so the reference signal generated by the servo controller is passed through detection circuitry on the RPT. Since this signal is used to generate the sine and cosine outputs of the resolver simulator, the functionality of the detection circuitry will be verified next. The RPT detection circuit is designed to pass a 4880 hertz, 5.6 volt peak to peak reference or generate a pulse to be sent to the resolvers capable of delivering 800mA if the reference from the servo

is lost or drops below acceptable levels. A power resistor is used to provide a load for the RPT's drive circuitry. A DMM is used in the AC volts range to verify the amplitude of the reference signal on the output (resolver) side of the RPT. A DMM is used instead of a frequency counter because if the RPT is intermittently switching between the servo reference and pulse output mode, the counter may be locked onto the reference at the time it is polled by the 990 computer, however, the DMM can be programmed to sample multiple times and give one response to one poll from the 990. The voltage amplitude reading of the pulse output of the RPT is substantially lower than the reference from the servo control module. A minimum and maximum reference voltage and frequency test are also done at this time. The DMM is used to verify that the RPT does not go into the pulse mode with a proper reference signal.

The primary function of the RPT is to monitor motion of the robot arm both under power from the user supplied 24VDC and the 6VDC battery. Under operation from the user 24 volt power supply, the module's ability to track movement a quadrant at a time is verified. Using the resolver simulator, eight positions are used to check the RPT's ability to track motion in the clockwise and counter clockwise direction. Two positions on either side of the 4 quadrant thresholds (0, 90, 180, and 270 degrees) are selected to verify proper quadrant tracking. By doing this, both quadrant tracking and quadrant

threshold circuitry are tested simultaneously. The decoding circuitry is verified by initializing all channels then moving only a selected few channels. Selecting some channels to be fixed and some variable is done to highlight stuck LOW and stuck HIGH conditions. During the clockwise test, four channels are connected to a fixed angle and the other four are connected to the resolver simulator. The resolver simulator is an electronic circuit that is used to generate sine and cosine signals for testing the RPT. The resolver simulator is used instead of a mechanical resolver to increase the speed and repeatability of the final test of the RPT. In the counter-clockwise test the connections are reversed so that all channels are verified to track movement. The RPT is capable of tracking and reporting 16384 rotations before rolling over to 0. By presetting the RPT to a near maximum rotation count and using the resolver simulator to cause an overflow, stuck-bit conditions as well as number of rotations can be monitored with very little actual rotation. It is desirable to reduce the number of rotations used in testing because even at the maximum 7000 RPM, it would take over 2 minutes to move from 0 rotations to 16384 rotations. As part of the rotation test, the RPT was initialized to a minimum rotation count and an underflow condition was induced.

There are two error flags which the RPT reports that are related to quadrant movement. The first error flag tested at this point is the illegal quadrant preset test. If the user

attempts to enter a preset position to the RPT that is 2 quadrants (180 degrees) out of phase with the actual resolver position, the preset position will be discarded and an error flag will be set for each corresponding channel that is out of phase. The test for this is to set the same angle for every channel on the RPT, preset all channels incorrectly, and verify an error flag for all channels. The second error flag is set for illegal quadrant movement. The RPT sets this flag when quadrant movement does not occur between adjacent quadrants. This test is accomplished by setting up the resolver simulator to a quadrant and properly presetting the RPT to the particular quadrant. The resolver simulator is then set to a new angle 180 degrees out of phase with the preset quadrant without the simulated rotation. The circuitry on the resolver simulator is fast enough to fool the RPT into thinking that an illegal quadrant movement had occurred and set the error flag for the test to verify. In actuality, an overspeed condition is simulated. The next step in the RPT testing is to verify rotation tracking.

Once the movement error detection circuitry has been verified, an OVER SPEED test will be performed. The RPT is designed to track a moving resolver up to 7000RPM. To verify this, the function generator is used to clock the resolver simulator to represent this rotational speed. After spinning a number of rotations, the resolver simulator stops and the RPT is interrogated to verify it tracked the correct number

of rotations and that no error flags were set. The RPT reports illegal quadrant movement if it is not tracking properly at this speed.

After functionality during operation on user 24VDC is established, operation on battery power is verified. The RPT has eight sampling rates while it is in the "sleep" mode on battery power. For this test, only the minimum and maximum rates are tested. During this part of the test the PRT and pulse width of the reference pulse output by the RPT are checked as the resolver simulator is used to simulate motion. Current measurements will be made from the simulated battery to verify the appropriate amount of current is drawn for a given pulsing rate. It is critical to verify battery life that is expected during operation in the field. In actual operation, the RPT will be expected to operate for weeks on battery power. During simulated motion, the RPT will come out of its sleep mode and pulse the resolver at 1 ms intervals with a 50mS pulse. When the simulated motion is stopped, the RPT reverts to the programmed sampling rate established during initialization. A timer counter is used to verify both active and sleep mode PRT's. In addition to verifying PRT's and pulse width of the RPT generated pulse, this also verifies that the UUT will properly switch between active and sleep modes. Tracking motion under battery power is checked by initialilizing the unit operating on user power then removing power and simulating motion with the resolver simulator. This

proves that motion can be properly tracked in the event of lost user power and that the UUT properly interprets the pulse shaped sine and cosine signals returned from a resolver.

The RPT is capable of detecting opens and shorts under certain conditions on the resolver primary and secondary. On the resolver primary, it is also imperative that the short circuit detection circuit be functional to avoid destroying the module during operation in the field. The RPT reports shorts or opens only while operating on battery power. It is assumed that if user power is present, the servo control module can handle any problems due to detected opens and shorts on resolver lines. The resolver secondary opens and shorts tests are basically the same in that the RPT is initialized with user power on and rotation is initiated before user power is removed. While operating on battery power, the open or short condition is applied to all eight channels simultaneously then removed before user power is restored. After user power is restored, the module open/short error flags are checked to verify that the open or short was detected on every channel. The shorted resolver primary error flag is checked by switching in a resistor of a low enough value resistance across the primary output of the RPT so that the pulse generating circuit shuts down. A resistor is used instead of a dead short to prevent severe damage to the UUT during testing.

There are functions of the RPT that do not necessarily

fall into one of the above tests nor do they affect the operation of any other part of the circuitry on the UUT. During this test all remaining error flags that have not been previously checked are set.

CHAPTER 4

TEST STATION HARDWARE

A. EQUIPMENT RACK

In order to optimize utilization and reduce maintenance cost, the test station equipment rack for the RPT final test has been configured to very closely match existing equipment racks that are presently in use for the final manufacturing test of products that are already in production at TI. Test station utilization is enhanced because there is the opportunity to test products on different test stations if production volume is unexpectedly high for a short period of time on one particular device or if a test station breaks down and requires an extended period of time to repair, product testing can be moved to another test station. Maintenance costs due to repair are reduced because once a technician is familiar with one station, very little additional training is needed to repair other equipment racks. By standardizing on the instruments that are used in the test stations, the cost of maintaining spares is reduced. Most of the test stations at the TI plant in Johnson City, Tennessee have the same digital volt meters, function generators, and programmable power supplies.

The test station equipment rack that is used for the final test of the RPT is made up of three basic components. The test station itself is under control of a TI model 990

computer. The interface between the 990 computer and the rest of the test equipment as well as the UUT is a TI model 990 expansion chassis. The expansion chassis provides buffered logic level signals that will drive the IEEE-488 instrument bus and the resolver simulator. The only other signals from the 990 computer are the RS232 lines required for communication with the RPT. The second major component of the test equipment rack is the commercial test equipment. Figure 4-1 gives a list of test equipment used in the equipment rack.

Hewlett Packard 8116A Pulse/Function Generator
Kikusui DP02112A/PLZ 72W GPIB Programmer/Electronic load
Systron Donner P1 Programmer
Systron Donner PQ50-1 Programmable DC Power Supply
Systron Donner PHR50-8 Programmable DC Power Supplies
Fluke 1953A Counter/Timer
Data Precision 8200 Precision Voltage and Current Supply
Fluke 8502A Digital Multimeter
California Instruments 846CM Control Module
California Instruments 1001TC Programmable AC Power Supply
A. D. Data Systems 56E Relay Matrix

Figure 4-1. Commercial test equipment.

The third component of the test station is the signal chassis. The signal chassis is a custom interface between the 990 expansion chassis, commercial test equipment, and a bulk head connector where the RPT test head is attached. The signal chassis contains wiring from connections to the commercial test equipment and 990 expansion chassis to the 1248 contact bulk head connector. The signal chassis also has four fixed DC power supplies provided to the bulk head connector. A safety interlocking relay is also located in the signal chassis to prevent raw 110VAC power from being present on the

bulk head connector pins when no test head is present.

B. RPT TEST HEAD

The test head is the interface between the equipment rack and the UUT. It provides the custom interconnections between the common inputs and outputs of devices in the equipment rack and the unique connection points of the device to be tested. This allows the equipment rack to be customized for each device to be tested relatively inexpensively. The RPT test head contains wiring from the bulk head connector to the connectors that mate to the UUT and to internal circuitry that is unique to the testing of the RPT. There are two wire wrap circuit cards in the RPT test head. One card has circuitry to perform the RS232 to RS485 and RS485 to RS232 conversion required for the 990 computer to communicate with the RPT. The second card in the test head is the resolver simulator board. This board generates the sine and cosine signals used in testing the RPT. The RPT test head also contains other components that are needed specifically for testing the RPT. There are five resistors used for current sensing and providing loading to the RPT outputs. To increase the longevity of relays in the equipment rack relay matrix, three power relays are located in the test head to switch the main power to the UUT, the AC reference signal from the equipment rack AC power supply, and to use in the shorted resolver test. These relays are used instead of matrix relays because of

sturdier contacts and they are easier to replace in the test head. Terminal strips are provided in the test head for common connection points for busses and to provide easy access points to these busses for hardware and software debugging efforts. Even though the programmable power supplies can be initialized for current limiting, a fuse in the RPT's power supply input is provided for additional safety. A schematic of the test head can be found in the appendix.

There is very little mechanical consideration needed in the test head design due to the standardization of the chassis used for test heads at TI's Industrial Systems Division. The RPT is physically small (2.5" wide x 7" long x 1.5" tall) so a chassis that was surplus from a previous project is used. Connectors that mate to the UUT are a wire wrap style so they can be stacked. This allows for a worn connector in the test head to be replaced without disturbing the internal wiring. The UUT connectors are mounted on a hinged lid for easy access to the test head's internal circuitry. Edge card connectors are used to facilitate easy removal of the two circuit boards in the test head. The three power relays used in the test head are mounted in sockets for easy removal and replacement.

The first electrical consideration is the signal bus structure in the test head. These signal busses are used as common connection points for the commercial test equipment in the equipment rack, circuitry in the test head, and signals

from the UUT. Figure 4-2 gives a list of considerations used in determining the number of signal busses needed.

- 1) Commercial test equipment that provides stimulus to the UUT and the resolver simulator board.
- 2) A bus that commercial test equipment connect to make readings.
- 3) Each sine and cosine channel of the RPT has two bus connections where signals can be provided.
- 4) The source of the reference signal for the RPT and resolver simulator.
- 5) The output reference of the RPT.

Figure 4-2. Bus requirements.

A general purpose stimulus bus is used primarily to drive the resolver simulator and make timing measurements of the reference output of the RPT. A bus which is referred to as the sense bus is used for measuring voltage drops across load and current sense resistors as well as a secondary bus for timing measurements. The RPT is used to complete the path between the resolvers on the motor and the servo controller so there are two connections for the sine and cosine of each channel. The test head will contain a separate sine and cosine bus combination each connection and the sine and cosine output of the resolver simulator. Two busses were used so one group of sine and cosine channels could be connected to a signal that represented a fixed shaft position while the other channels could be connected to the resolver simulator for motion tracking. The reference signal input for the RPT under test and the resolver simulator forms another bus in the

test head. This allows for the AC power supply to provide the reference signal to the RPT and the resolver simulator. The last major bus in the test head is the bus that connects the reference output of the RPT to load resistors and as another reference input for the resolver simulator board.

The resistors used for testing the RPT are located in the test head. The selection of the values that are used is based on what parts were available in house and what the values of resistance were needed to provide the correct load for a particular test.

There are two circuit boards in the RPT test head. These two circuit boards provide functions that are needed specifically to test the RPT and are not producible by any of the commercial test equipment in the equipment rack or the 990 computer. A major concern in the selection of components for use in the two circuit boards is availability of the components. Wherever possible, components that are used in other products manufactured at TI in Johnson City are used. This allows for easy access to replacement parts should the circuit boards need repair.

C. RS232 TO RS485 CONVERTOR

The RPT requires serial communications to be of the IEEE 485 protocol. This protocol is not standard in the 990 computer serial communications card so a circuit had to be designed. The RPT has only differential transmit and receive

lines. Figure 4-3 shows the basic block diagram of the circuit used to convert the RS232 protocol data that is sent and received by the 990 computer to the RS485 protocol data that is received and sent by the RPT.

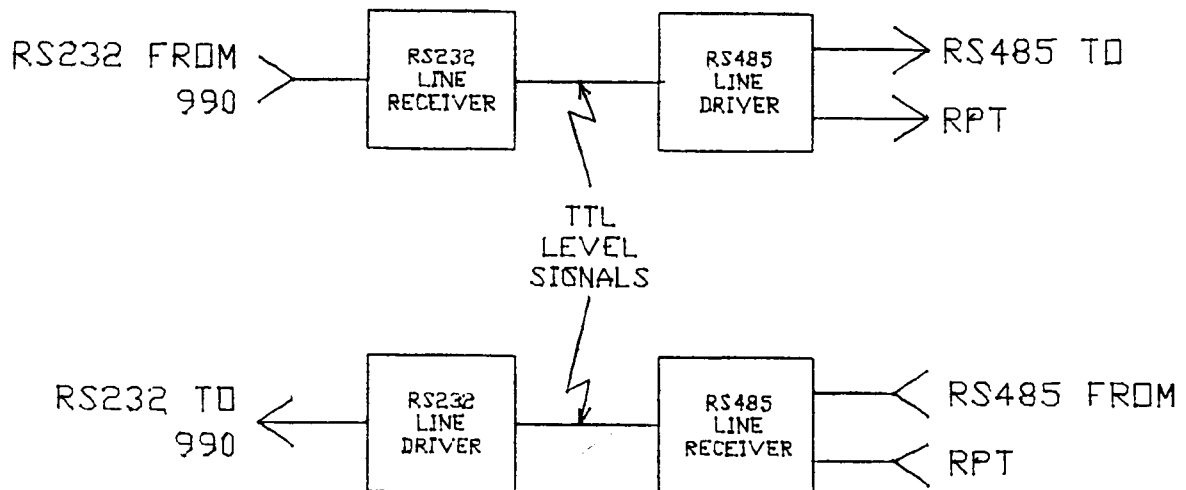


Figure 4-3. RS232 to RS485 convertor.

This circuit is implemented using standard RS232 and RS485 driver and receiver integrated circuits. A RS232 receiver is used to convert the RS232 signal to a TTL-level which was used by a RS485 driver to convert it to a signal the RPT can receive. Similarly, a RS485 receiver was used to take the signal from the RPT and convert it to a TTL-level signal. A RS232 driver is used to take this TTL-level signal and convert it to the RS232 protocol to be sent to the 990 computer. A schematic of this conversion board is included in the Appendix.

D. RESOLVER SIMULATOR

The other circuit board in the RPT test head is the resolver simulator. The primary function of this board is to convert a reference signal that is generated by an AC power supply or the pulse output of the RPT into angular information in the form of a resolver sine and cosine output. The functions of the board are computer controlled and connections to the RPT being tested and the commercial test equipment are done by the relay matrix. The schematic of the resolver simulator is located in the Appendix.

The resolver simulator has several features incorporated to aid in the testing of the RPT module. It also has a number of features that are not required, or even used, in the testing of the RPT. During the definition of the requirements for the resolver simulator a decision was made to add features to the board that would not be required in the testing of RPT modules. These features were added to allow the resolver simulator to be used in testing servo control modules and to aid in troubleshooting.

The RPT was originally designed to work with three different resolver reference signals. During the design phase of the RPT, the requirement for operating at 2666Hz and 3000Hz was dropped after the design and construction of the resolver simulator was completed.

What is referred to as second generation operation is a 16 volt peak reference with a frequency of either 2666 hertz

or 3000 hertz. Third generation operation is a 5.6 volt peak reference with a frequency of 4880 hertz. The selection of generation and reference is done by two-bit control. One bit selects generation and, in the case of second generation, the second bit selects the reference frequency. Three-to-one multiplexing is used before the square wave output of the desired reference frequency is converted to the required sine wave of appropriate amplitude. The square wave to sine wave conversion is done by a circuit that is used in the servo controller that is part of the same family as the RPT. Even though this solution required discrete component implementation, all parts are readily available at the manufacturing site. At this point, an analog switch is used to select between one of three sources of a reference signal that will be used to generate the resolver sine and cosine signals. The first two sources are the second and third generation references generated on the resolver simulator board. The third source is an externally selected source such as an AC power supply or the output of the RPT. The output of the analog switch is connected to the multiplying DACs and is also available as an output pin on the edge card connector. The simulated rotation speed of the resolver simulator is controlled by either an analog voltage or a TTL-level clock input. An analog voltage is used for gross speed settings or for connections from the velocity drive voltage of a servo controller. A clock input is used as a more precise means to

generate a desired simulated RPM. The direction of spin is controlled by one control bit to the up/down counters which form the basis of the sine and cosine signal generation. The outputs from the up/down counters are used to generate an address for EPROMs which contain a map of data that gives a proportional digital value of sine and cosine for a given count from the up/down counters. This scaled data provides the digital input for the two multiplying digital to analog converters(DACs). The resolver simulator is designed with the capability to start and stop at a given angle and rotations by means of digital data that represents an initial position and rotations count and a final position and rotations count. This feature was incorporated so that motion can be started and stopped under computer control.

For convenience and debugging the resolver simulator, three crystal-based oscillator circuits are incorporated into the resolver simulator.[4] Each oscillator circuit is divided down to form the base frequency of each resolver reference signal. The primary concern in the design of each of the oscillators is part availability at the manufacturing site. The crystals are the most difficult parts to standardize.

The method used to select a crystal frequency was to take the reference frequencies of 2666 Hz, 3000 Hz, and 4880 Hz and multiply by two until the value of the frequency obtained could be found in the crystal part drawings used to build standard products. After the value of these crystals were

found, a simple divide circuit was used to divide the crystal frequency down to the desired reference frequency.

The conversion from the square wave from the oscillators to a sine wave reference is done by a square wave to sine wave converter circuit which is part of the servo control module that is part of the same family as the RPT.[5] This circuit is implemented with discrete components and has sufficient bandwidth to properly convert all three reference frequencies. There are ICs that can generate an acceptable sine wave but they are not readily available in house. The standard output of this square-to-sine wave convertor circuit is 5.6 volts peak so an additional amplifier circuit is used to generate the 16 volt peak signal required for the second generation reference. To get a 16 volt peak signal, the plus and minus 15 volt supply voltages must be increased. Since testing at the 16 volt peak is not done, this feature is not used.

An analog switch IC is used to select between the two different amplitudes generated by the square-to-sine wave convertor circuit. The TTL logic signals that control the analog switch also control the three-to-one multiplexer chip that feeds the sine wave convertor. This two bit data reduces the required control signals by selecting the proper output amplitude for a given reference frequency. The third input to the analog switch is for an external reference signal. The output of the analog switch IC is available on a connector pin and is also connected to the reference inputs of the two

multiplying DACs.

The digital data used by the multiplying DACs is generated by the scaled output of up/down counters. The clocking signal for these up/down counters is generated by either a voltage controlled oscillator (VCO) or from an external TTL-level square wave generator. The selection of the input that will be used to provide the clock for the up/down counters is done by one control bit. The main consideration in the VCO design was part availability. The VCO circuit that is part of the resolver simulator board comes from the Raytheon linear integrated circuits data book.[6] The input voltage can be varied from 0 to 10 volts DC. The output of this circuit can vary up to about 250 KHz.

The output of the clocking circuit is used by up/down counters. These counters provide the raw digital data that eventually is used to amplitude modulate the reference signal into a sine and cosine signal proportional to a desired angle. Up and down counting gives the simulated clockwise and counterclockwise motion. The up/down counters also provide a number of rotations count. The RPT is capable of counting up to 14 bits so the most significant 14 bits of the up/down counters represent this rotation count. The least significant 14 bits of the up/down counters represent the actual angle data. Table 4-1 gives a list of the least significant angle data bits and their weighting.

Table 4-1. Bit weight in degrees.

<u>Bit Number</u>	<u>Weight(Degrees)</u>
0 (LSB)	0.022
1	0.044
2	0.088
3	0.175
4	0.351
5	0.703
6	1.406
7	2.813
8	5.625
9	11.25
10	22.5
11	45
12	90
13 (MSB)	180

The clocking for the rotations count is provided by the MSB of the angle data. The count enable for the up/down counters is controlled by one bit and can either be externally generated or the output of a comparator circuit. The externally generated count enable is generally used to allow the resolver simulator to rotate continuously in the selected direction. A 28-bit comparator circuit is incorporated into the resolver simulator to allow for presettable start and stop movement. In this case, the start angle and rotation is loaded into the up/down counters and the finish angle and rotation is loaded into a separate data latching circuit. The output of the up/down counters and the final position latching circuit are sent to a comparator circuit which generates a one-bit EQUAL signal that disables the up/down counters when the desired final position is reached.

Sine and cosine are not linear functions so taking the data from the up down counters and using it with the multiplying DACs to generate the resolver sine and cosine would give incorrect results except at 0, 90, 180, and 270 degrees. A method is needed to convert the linear count output from the up/down counters to digital data that can be sent to the DACs so that a more accurate amplitude of sine and cosine can be generated for the angle vector. The resolver simulator board uses EPROMs programmed with scaling data that is used to provide the multiplying DACs with proper data for generating the sine and cosine resolver signals.

For example, for a 45 degree angle the output of the up/down counters would be hexadecimal 800. Both the sine and cosine for 45 degrees is 0.7071. In order for the DAC to generate a resolver sine and cosine that is 0.7071 times the reference, it has to be loaded with hexadecimal 16A0. The output data from the up/down counters is used as the address for the EPROMs. At address 800, the data will be 16A0. The equation for the voltage output of the DACs is given as

$$V_{out} = V_{ref} * (Input\ Data / 8192)$$

V_{out} represents the output voltage of the DAC, V_{ref} represents the resolver reference voltage, and Input Data is the decimal value of the data generated by the scaling EPROMs. The four least significant bits of the output of the scaling data are logically ANDed with a control. This eliminates problems with output settling times of the DACs when high speed rotation is

simulated.[7]

Two multiplying DACs were used in the resolver simulator to generate the resolver sine and cosine signals from the resolver reference and the scaled digital data from the EPROMs. The DACs each have a 14-bit data input and were operated in the bipolar mode. Fourteen bits of resolution was more than what was required to test the RPT but it was decided during the definition phase of the project to use 14 bits so the resolver simulator could be used to test the servo controller. The servo controller used with the RPT had 12-bit resolution incorporated into its resolver-to-digital circuitry. These devices were not standard products at the manufacturing site but the need for high resolution necessitated their selection. The data from the scaling EPROMs was loaded into the DACs and then latched in the internal DAC register. The DAC had a transparent mode of operation, but it was found that the transition of the outputs of the EPROMs after an address change caused glitches in the resolver sine and cosine outputs. The op amps used to generate the sine and cosine signals from the DACs were powerful enough to drive all eight sine or all eight cosine inputs of the RPT simultaneously.

A 20L10 PAL is used in the resolver simulator to decode the control signals that select the mode of operation of the board. The PAL is used to reduce the number of ICs required to decode signals needed to operate the resolver simulator.

There are three decoding functions needed to select modes of operation. The square wave that forms the basis of the resolver reference is the output of a three-to-one multiplexer implemented in the PAL and is selected by two external control lines. A single control line is used to select between the two sources of a clock for the up/down counters. The last decoding function of the PAL is the ENABLE control for the up/down counters. Either the output of the comparators for final angle determination or an external disable are used for enable control.

Two software packages were used in the design of the resolver simulator circuit. The fuse plot drawing for the PAL was developed using a commercial software package called CUPL. This program took a file that contained the logic equations and test vectors and verified the expected results from the equations and vectors as well as generating the fuse plot for use with a PAL programmer.

The second software package used for test hardware development was BASIC. The data that was programmed into the EPROMs was too voluminous to be calculated by hand and then entered into an EPROM programmer. The first requirement of the program was that the scaled data calculated from the input data from the up/down counters be stored in a file that could be down loaded to an available EPROM programmer. The other major requirement was that this data file be in a format that could be downloaded to the EPROM programmer.[8]

The listings of the programs used to generate the PAL fuse plot and the data programmed into the EPROM can be found in the Appendix.

CHAPTER 5

TEST SOFTWARE

A. TEST STATION SOFTWARE

The test station software that controls the instrumentation, the resolver simulator, and communication to the RPT is written in PASCAL. PASCAL is the standard language used to write test software that is resident on 990 controlled test stations at TI in Johnson City. Using a standard language allows for utilization of routines already written for other products and test stations presently in use. Examples of standard routines are programs that control the commercial test equipment, operator interface menus, and batch programs that link different test programs together.

A test executive is used to handle the calling of specific test routines, parameter files, and communication command files. It also handles operator prompts and interprets the response to these prompts. This test executive is relatively standard on all 990 based test stations with only minor alterations that are necessary for a particular product. The test executive controls the operation of the individual functional tests that are part of the RPT's final test.

The ten major test routines written for the RPT are listed in Figure 5-1 along with brief description of the function they test. The breakdown of functions tested by each

routine is based on the functional blocks of the RPT and on the 990 memory available for loading a particular test program.

POWERDC	RPT DC power supply tests
COMM	ASCII communications tests
COMMB	Binary communications and EEPROM tests
POWERBAT	Battery backup tests
OPENSHT	Resolver opens and shorts tests
QUADRANT	Quadrant threshold tests
REV	Revolutions tracking tests
BUMODE	Back up mode resolver pulsing tests
REF	Reference drive tests
ALBOE	Other minor tests
CALRPT	RPT calibration routines

Figure 5-1. Individual test programs.

The tests are written to be executed individually without results from previous tests. Most of the tests also have a number of subtests incorporated into them. When the test detects a failure, a message is printed out that contains information on what failed, any applicable parameters that are part of the test, and the number of the test.

B. INDIVIDUAL TEST PROGRAMS

The first test preformed on the RPT is the DC power supply test. A programmable DC power supply is connected to the UUT by a power relay in the test head with a series resistor for current measurement. The DMM is programmed to measure DC volts and is connected across the series resistor by means of the relay matrix. Two measurements are made, both at the minimum and maximum operating voltages of the RPT. The programmable DC power supplies also have a current limiting

feature which is used during this test to reduce the chance of a malfunctioning UUT destroying itself. The current limit is set slightly above the maximum expected current draw of the RPT with no connections made to the resolver inputs or outputs. Any failures during this test are considered fatal and further testing is halted.

The RS485 line termination test is performed after the basic functionality of the power supply has been verified. The test verifies the proper functionality of the four position switch that selects the termination resistors for the RS485 communications lines. Two switches are required to be ON for each termination resistor to be connected into the circuit. The operator is prompted to set these switches to alternating ON and OFF patterns. The DMM is programmed to read resistance and the program verifies that no switches are in a shorted ON condition by checking for an over range resistance reading (open circuit indication). The last pattern set by the operator is ALL ON. This pattern verifies that all the switches can be turned on by reading the 120 ohm termination resistors, which are also checked at this point. This test also verifies a good connection from the test head to the RPT for communications testing.

At this point, the UUT has been proven to power up properly and the serial lines to the board are viable. The address of the RPT on the RS485 serial bus can either be hexadecimal 1E or 1F depending on the switch setting. The

testing of this switch position can be done only by communicating to the RPT using the binary protocol with the proper address in the message sent to the RPT. Data files which contain command and expected response information are used by the calling program. The calling program takes a line from the command data file and sends it out on the RS232 card in the 990 computer. The response to this command, if any, is verified against expected response data file. All four positions of this switch are set to ON and the board is interrogated at address 1F at 9600 baud. If no response is returned, the board is deemed bad. The switches are then set to ALL OFF and the board is addressed at 1E at 1200 baud to verify proper communication. Only these two patterns are used since a shorted switch would cause the test to fail because the RPT would not respond.

The binary communications test is also used to verify other functional aspects of the RPT being tested. The message returns all status indications which include a power-up self-test and the revision level of the RPT's own software. The power up self test performs an internal processor check, ram test, and checksum test on the program in the RPT. If the checksum works, there is a high level of confidence that testing functions such as communications using ASCII characters is unnecessary since the serial data lines have already been proved to be functional and the software is correct. If the wrong version of RPT software is installed,

the test will abort and fail the unit. ASCII communication is not rigorously tested because the serial lines are verified during binary communication and a proper checksum indicates that the software that provides ASCII communication is viable.

The next function to be tested on the RPT is the 16384 bytes of EEPROM memory that is user accessible. In an operating robot system this memory would be used to store information about the particular robot that the RPT has been installed in for the servo controller to interrogate. The memory can only be written to or read from using two commands in the binary protocol communications. The command to write includes the start address, number of bytes of data, and the actual data. A major restriction in programming the EEPROM data is that only 22 bytes can be programmed in one binary command. With this restriction, it takes 745 commands to fill the entire EEPROM memory with data. The 990 takes about 5 or 6 seconds to send and process the return of each command. Because of the excessive time required to fill memory, a rigorous pattern test is not practical. The method of testing that is used is based on the mapping of the user memory within the EEPROM. By choosing the single unique address line to write to, all address lines to the EEPROM are tested in both states. As a final check, a status check is performed to verify the checksum on the EEPROM.

After it has been established that the test station can communicate properly with the UUT, the ability of the module

to track motion can be tested. Communication is required for presetting positions and reading back results after movement. The first test performed on the RPT will check for the proper tracking of movement within a quadrant and quadrant-to-quadrant.

The quadrant test verifies four aspects of operation with respect to motion tracking by using the resolver simulator to move through eight different positions, two in each quadrant. These functions include proper quadrant indication for a given angle of rotation, quadrant-to-quadrant tracking, rotation overflow, and channel-to-channel decoding. The RPT is polled by the test station after each of the eight discrete steps by the resolver simulator to verify proper rotation and quadrant tracking. The positions in each quadrant are selected to be near the 90 degree thresholds of each quadrant. This is done to assure proper tracking at the RPT's highest level of resolution. The eight positions, when combined, simulate movement through all four quadrants. This verifies tracking in all four quadrants as well as quadrant-to-quadrant movement. The RPT increments or decrements the rotation count each time the 0 degree boundary is crossed. By initializing the resolver simulator and the RPT so that the 0 degree boundary is crossed, rotations counting is tested. By initializing the module to a maximum rotation count, the crossing of the 0 degree boundary causes a overflow condition, which is indicated by zero rotations. Three digital bits are

used by the microprocessor to scan the sine and cosine inputs. If one or more bits are stuck, position information from the sine and cosine channels is not valid. A decode test is performed by simulating a fixed angle on half the channels while using the resolver simulator to "move" the other channels. This process is repeated by connecting the resolver simulator to previously fixed channels while holding the other channels at a fixed angle and simulating movement.

There are two error flags associated with quadrant movement, 180 degree preset error and illegal quadrant movement. To verify 180 degree preset error, the resolver simulator is set to a angle in a given quadrant. The RPT is then preset to a quadrant that is 180 degrees different from this angle. The response of the RPT is checked for an error condition. The illegal quadrant movement error flag is checked by initializing the resolver simulator and the RPT to the same quadrant. The resolver simulator is then set to a new angle that is 180 degrees without actually rotating. All related status bits are checked during both of these tests.

After it has been verified that the UUT can properly track movement, the functionality of the open and shorted resolver detection circuitry is tested. The test is done at this point because a high level of operation is verified with the quadrant test and if the shorted resolver test damages the RPT, further tests that check other functions of the board will fail the unit. This helps the repair technician to

troubleshoot problems more readily with the RPT.

It is assumed that the servo controller handles all open and short conditions during a powered up condition, the error flags on these conditions are only set when the RPT is working on battery power. The RPT and the resolver simulator are both initialized to the same position. The resolver simulator is then put into a free rotation mode of operation at a speed that will guarantee that a properly functional RPT will detect an open resolver before main power is removed from the RPT. After power is removed, all connections to the resolver simulator are broken and then restored. When main power is restored to the UUT, the open resolver error flags are verified to be at their proper states. The shorted resolver test is accomplished by switching in a resistor by means of a power relay located in the test head. This resistor is of a low enough value as to cause the RPT reference short detection circuit to sense a short and set the shorted resolver error flag. A resistor is used instead of a dead short in an attempt to prevent major damage to the board if this circuit reference drive current sense circuit is not functioning properly.

The next test tests aspects of revolution tracking of the RPT. In order for the RPT to track motion while operating under battery power, the unit is initialized while operating on the 24VDC power supply and the simulated battery. For this test, the main power is removed from the UUT after it has been

initialized and the resolver simulator starts rotation. After the simulated rotation stops, main power is restored to the module and the new rotation count is verified. For the other part of revolution tracking, rotation counting is performed under an overspeed condition of 7000 rpm and on battery power. These tests were kept separate to help the repair technician more readily isolate problems, especially since this test is the first verification of motion tracking under battery power. For rotation tracking in an overspeed condition, the frequency generator is used to generate the clocking for the resolver simulator instead of using the on board VCO. This is done because the maximum clocking rate of the VCO is about 250KHz and the function generator is capable of outputting a more accurate and repeatable signal. Illegal quadrant error flags are checked to verify that the RPT was able to track motion properly at this speed.

One of the most critical functions of the RPT is the ability of the module to go into a "sleep" mode if no movement is detected for a certain period of time. When the RPT first loses power and is operating on battery power only, it will pulse the resolver once every millisecond. If no motion is detected it will reduce the pulsing frequency to a user selectable preset value that varies from 2.5mS to 320mS. Since this time is a software function of the RPT, only the two extremes are verified at the final test to confirm proper current draw from the simulated battery. The resolver

simulator has one bit that can control the free rotation mode of operation which is used in this test to start and stop the simulated motion. The module is first initialized to the 2.5mS sampling time. After power is removed from the module the rotation of the resolver simulator is put into the free running mode. The frequency of the pulse train is verified to be approximately 1 KHz and the width of the pulse is verified to be roughly 50uS wide. After this has been verified, the resolver simulator is disabled which should send the RPT into the sleep mode of operation. At this point the frequency of the pulse reference signal should be about 400 Hz. If this test passes, main power is restored to the module and the sampling rate is reprogrammed to 320mS. Main power is once again removed and the resolver simulator is set to a free running mode and then halted. The frequency of the pulse train is then verified to be approximately 3 Hz. The lower sampling frequency is not as accurately measured as the maximum sampling frequency because of the slow sampling rate of the frequency counter for such a low frequency signal and the desire to have the test execute as quickly as possible. A current measurement from the battery is made at each of these steps.

Since the test station does not include an oscilloscope, the DMM is used to verify that the UUT is not constantly operating in the backup pulsing mode. If the RPT detects the loss of a valid resolver reference signal, it switches to the

pulsing mode and monitors resolver movement itself. To detect this failure, the DMM is used to read the reference output from the RPT under a maximum load condition. The AC power supply provides a 16 and 5.6 volt peak signal at the appropriate frequency to simulate a servo controller supplying a reference signal to the resolvers through the RPT. If the module is pulsing, the amplitude of the reference output of the RPT will be too low and this will be indicated by the DMM reading. A secondary function of this test is to test the load handling capability of the switching circuitry on the RPT. At 16 volts peak in the second generation mode, the RPT is expected to be able to pass 300mA with a minimal loss. For this part of the test, a 100 ohm power resistor is used to provide the simulated maximum resolver load. For the third generation mode of operation, the RPT must be capable of handling a 5.6 volt reference providing 800mA to a load. A 7.5 ohm resistor is used for loading the RPT in the third generation mode of operation.

The ALBOE test is a general test that verifies any functions that have not already been checked. These tests are not part of any other function that has already been tested. At this point, only status flags that have not been verified earlier are checked.

The final program that is part of the RPT final test is used to calibrate the UUT. The only part of the RPT that requires calibration is the battery charging circuit. To

calibrate the circuit, the DMM is connected to the battery input terminals using the relay matrix, and the person running the test is prompted to adjust the potentiometer that controls the output voltage on the battery charging circuitry.

The RPT test head is a relatively simple device except for the resolver simulator. Common problems that occur with test heads include broken wires and interface connectors. The resolver simulator card is complicated enough to warrant a self test program that gives the test station repair technicians confidence that the circuitry is functioning properly and the associated wiring is sound. To isolate the resolver simulators function as much as possible, relay connections and commercial test equipment setups are made to try to reduce the chance of extraneous failures. Relay connections are made in parallel paths whenever possible as part of this effort. The first part of the test is a walking-bit test to check for stuck-bit conditions. Data is loaded into the up/down counters and the final positions latches so the comparator circuit would indicate equal data. The equal bit is tested at the output of the PAL so the maximum circuitry can be tested at once. After all equal conditions are verified, data that would cause the comparators to indicate unequal data was latched and the equal bit is checked for a NOT EQUAL condition. This part of the circuitry is verified first since the latches and comparators are used in the next part of the self test to generate fixed and moving

sine and cosine signals. Since the resolver simulator can generate its own reference as well as sine and cosine signals, an external reference with a frequency and voltage different from any reference generated by the resolver simulator is used. The sine and cosine outputs are first tested for a frequency that matches the external reference frequency. After the frequency is verified, seven data patterns are written to the angle up/down counters which latch this data. One bit is changed at a time for the first six most significant bits with the other outputs held low, then an all low data pattern is latched. Using only the most significant six bits verifies the highest level of resolution that is needed to test the RPT simulator without over testing the accuracy of the resolver simulator.

There are two means of clocking the up/down counters. To verify the decoding circuitry that controls the source of the clocking signal as well as checking the count direction bit, a rotation test is performed. The resolver simulator is initialized with unique start and stop information and the counters are enabled. By initializing the board to allow only a few revolutions and sampling the amplitude of the sine and cosine within a short enough period of time to allow motion to stop, the fact that the board stopped rotating can be verified since the stop angle is different from the start angle. If the direction bit is not functioning, the board will still be "rotating" when the DMM samples the sine and

cosine signals. It is not likely that the sine and cosine amplitudes will both be exactly correct at the sampling time if the resolver simulator is still "rotating". If the clocking circuit is not functioning, no rotation occurs and the initial position settings are read by the DMM. This program is in the appendix as an example of a typical pascal program.

CHAPTER 6

CONCLUSION

A. REVIEW OF GOALS

The primary goal in the design of the final test of the RPT was to provide the highest level of test coverage in a minimal amount of time. The purpose of the final test of this module was to simulate, as closely as possible, the actual environment that the unit would operate in, not prove the viability of the design of each unit being tested.

By operating the RPT in an environment that simulates its actual use, a significant amount of test coverage can be obtained. By tracking motion on all eight channels under main power and battery backup power, a high level of confidence in the functionality of the circuitry such as the power supply, resolver signal sensing, resolver reference switching, serial communications and microprocessor circuitry can be achieved. Error conditions are introduced to further check for the proper operation of the RPT. Even with this level of test coverage, time constraints prevented the thorough testing of some parts of the circuitry. The final test time is approximately 10 minutes. This may seem like a significant amount of time, but the amount of communication required to the UUT and the desire to break down tests to a point that would make it relatively easy for a technician to troubleshoot a failed unit necessitated it.

Performance items that were altered from the original design specification also affected the test development. The requirement for the RPT to work with a 2666 Hz to 3000 Hz reference was dropped. This change allowed for the dropping of one test and a calibration procedure involving a phase lock loop adjustment. Design problems have temporarily restricted the RPT from tracking motion over approximately 4500 RPM. The test used for overspeed tracking is written so that when the maximum tracking speed is increased, modifications to that particular test can be implemented easily. The final design change was the increase the amount of user accessible memory from 512 bytes to 16384 bytes.

When the tests were originally written, little consideration was given to the final test time or if a particular test or subtest was actually needed. Before the consolidation of tests, the final test time was approximately 15 minutes. The justification aspect of this thesis forced a deeper consideration of the test software which resulted in the deletion of a number of subtests with a subsequent reduction in test time. A second reason for reduction in test time was the change in the EEPROM test. The significant increase in user memory and the time required to test this additional memory necessitated reconsideration of how to test it. The most significant contribution of this thesis was a reduced test time for the RPT. Another important contribution was a very thorough documentation of the test and software for

others to use.

The most glaring example of minimal test coverage was the user memory of the EEPROM. Programming the EEPROM required communicating the data to the UUT by binary commands. Due to the restricted length of the binary commands and the relatively lengthy overhead processing time of the 990 computer, only a stuck-most-significant-address-bit condition was checked. Each command took about 5 seconds to process and the amount of data that could be sent at any one time was only 22 bytes. With 16384 bytes of user memory, this would require about 62 minutes to write in just one pattern of data into the EEPROM.

During the process of debugging the final test software a number of design errors were found. Most of the design errors were related to RPT operating system software problems. As an example, one of the first errors found was that the system software was interpreting the sine position bit as the cosine position bit and the cosine position bit as the sine position bit. The error flags had not been implemented yet and the design group was only interested that the position count changed with resolver movement. During the quadrant test, thresholds near the four major axes are verified by rotating through each of the four quadrants with two angle checks in each quadrant. The least two significant bits of the rotation count indicate the quadrant and are shown in Table 6-1. When the resolver simulator moved across the 90,

180 or 270 degree axes, an improper quadrant value occurred.

Table 6-1. Binary quadrant values.

QUADRANT	ANGLE REPRESENTED IN DEGREES	BINARY VALUE
1	0 TO 90	00
2	90 TO 180	01
3	180 TO 270	10
4	270 TO 360	11

After finding this problem, the design team became very diligent in providing the latest revision of the operating system software for check out on the test station.

The RPT test head was the first device to use the test station. This meant that problems found early in the test debug effort could be due to problems with the test head or the test station itself.

B. THE RESOLVER SIMULATOR AS A MOTOR SIMULATOR

Other uses were identified for the resolver simulator once it was proved to be useful in testing the RPT. These uses centered around modifying the resolver simulators circuitry so it could simulate an actual motor and resolver combination in a servo control system. A servo controller (model 500-5024) was introduced into production at TI a few years ago. To test the motion control capability of this device, a motor with a resolver for position feedback and a servo amplifier were installed into a test head made

specifically for this unit. The approximate cost of this equipment was \$2000. By adding a few simple op amp circuits to the VCO and up/down control, the resolver simulator could be made to imitate the action of a motor/resolver combination for less than \$200. This was verified by connecting the modified resolver simulator circuit to the model 500-5024 servo controller and executing a simple motion profile program. Present plans call for the use of the resolver simulator to test the servo controller that is part of the same system as the RPT.

The modification of the circuit involved translating the drive signals of the 500-5024 into voltage levels that were compatible with the resolver simulator. The servo drive and resolver signals from the 5024 were differential bipolar analog voltages. The drive output analog voltage indicated direction by polarity and speed by amplitude. The VCO used on the resolver simulator was designed to handle only positive voltages and the up/down control was a TTL-level signal. The final consideration of the interface circuit was inertia. For all practical purposes, the resolver simulator had no inertia. If the state of the up/down control bit was changed, the resolver simulator would change direction within the limits of the propagation delays of the TTL circuitry. The first function of the interface circuit was to convert the differential drive and reference signals to single ended signals by using differential op amp circuits whose gain were

one. The single ended reference signal was used as is by the resolver simulator. The circuit that converted the drive output to voltages for the VCO and a TTL level up/down control signal consisted of an integrator driving a zero cross detector and absolute value circuit. The integrator created the effect of inertia by slowing the rate of change of the drive signal from the servo controller. The zero cross detector generated the TTL level up/down control signal and the absolute value circuit converted the bipolar drive signal to positive only voltage for the VCO.

Test and design engineering were involved with the concept phase of the design of the resolver simulator. Design engineering's interest in the project was a method to test prototype servo controllers in an office environment as soon as possible. The servo controllers were able to control eight motor/resolver combinations simultaneously. Eight motors operating at one time could generate undesirable levels of audible noise. The other reason for their interest was the uncertainty of if and when they could obtain eight actual motors and resolvers for prototype checkout.

The main inspiration for modifying the resolver simulator to more truly resemble an actual motor/resolver combination to a servo controller came from the design verification group. As previously mentioned, actual motors and resolvers are substantially more expensive than the motor/resolver simulator. Their function is to run extended operational

tests in conditions that could be expected to be found in a typical factory environment. The only major shortcoming of the unit for testing purposes is that the user could not actually look at the device and tell it is functioning, like a user can observe a motor spinning.

LIST OF REFERENCES

LIST OF REFERENCES

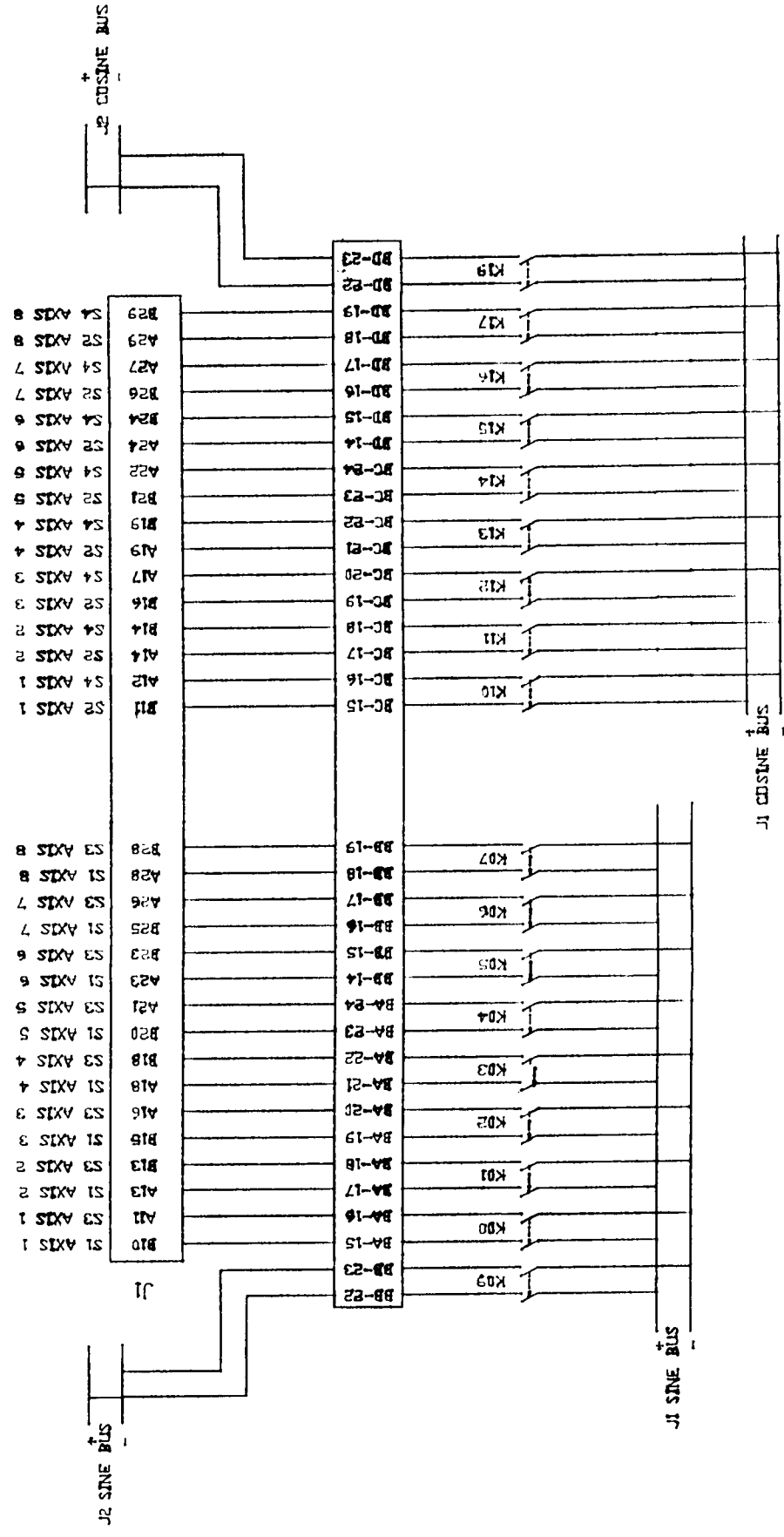
- (1) P. B. Crosby, Quality is Free, First Edition, Mentor, 1980
- (2) W. E. Perry, A Structured Approach to Systems Testing, First Edition, Prentice-Hall, 1983
- (3) Resolver Position Tracker Unit Specification, Texas Instruments Incorporated, 1988
- (4) D. Lancaster, TTL Cookbook, First Edition, Howard W. Sams & Co., Inc., 1974
- (5) Servo Control Unit Specification, Texas Instruments Incorporated, 1989
- (6) Linear Integrated Circuit Data Book, Raytheon Semiconductor Division, 1978
- (7) Data Conversion Products Data Book, Analog Devices Incorporated, 1988
- (8) Data I/O 29B Universal Programming System Operator's Manual, Data I/O Corporation, 1985

APPENDIX

RPT TEST HEAD

11.057.00

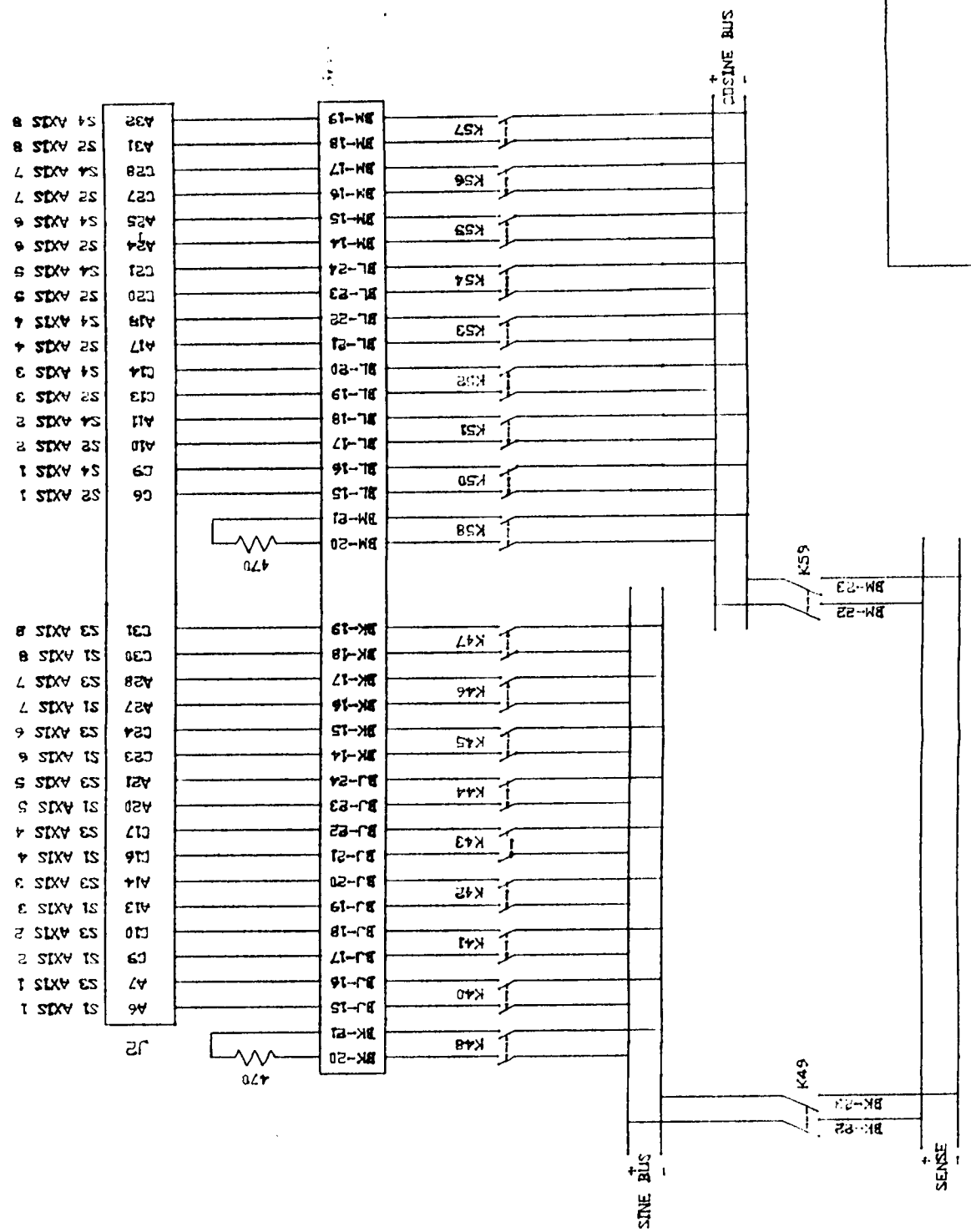
STANDARD CITY

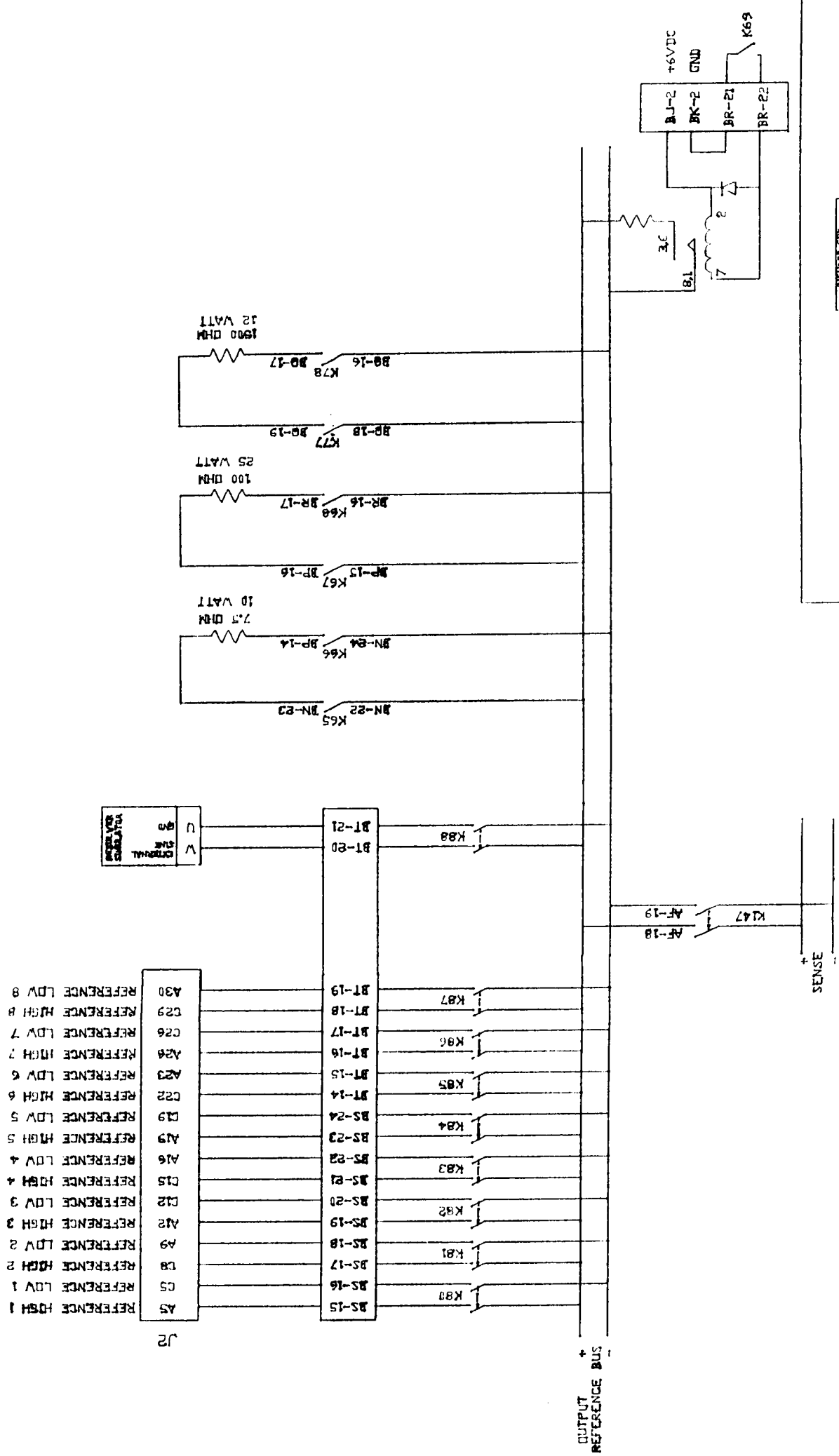


RPT TEST HEAD

11 OCT 68

WOMAC 201



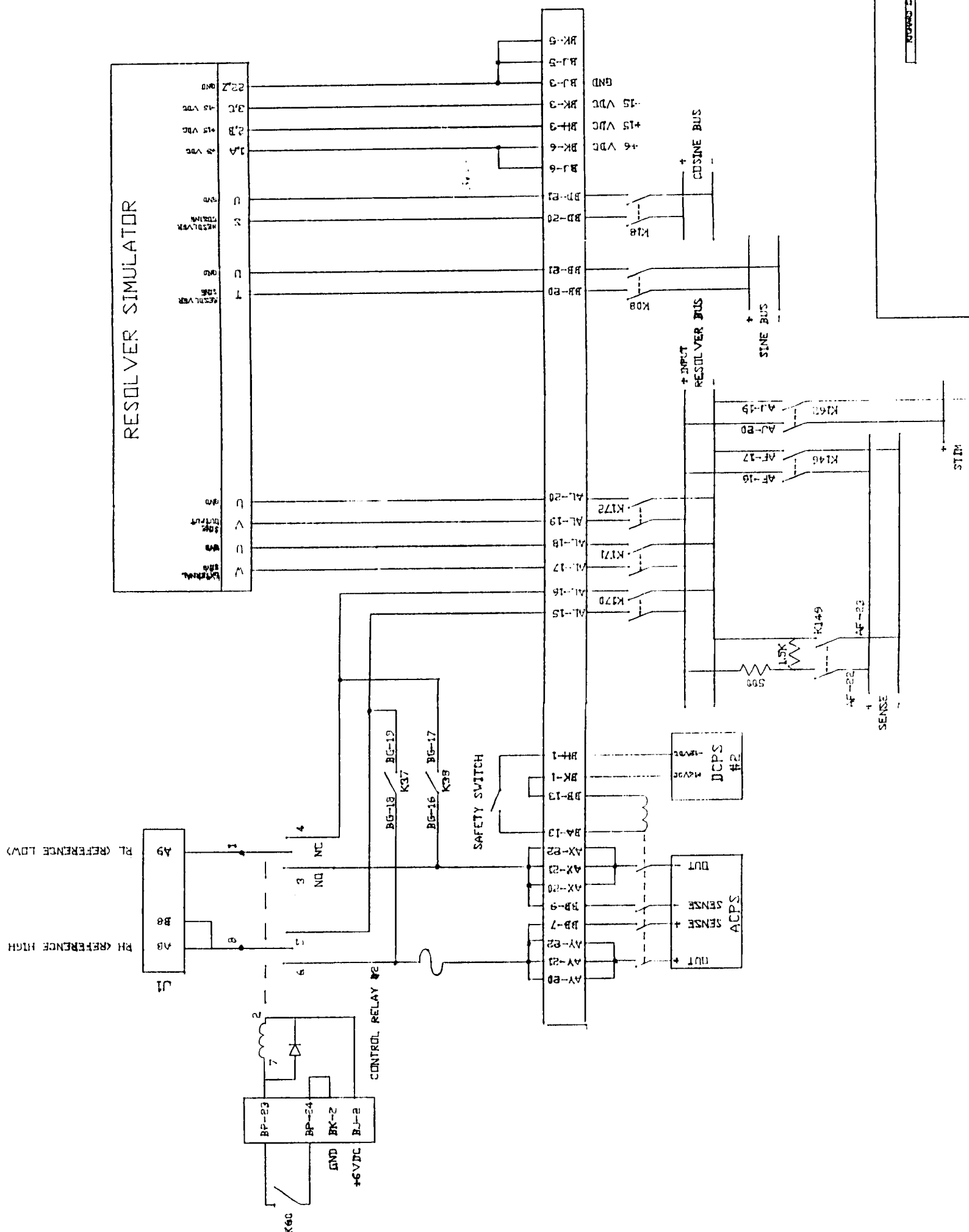


RPT TEST HEAD

REPORT

K35	BE-18	A32	S4 AXIS 8
	BE-19	A31	S2 AXIS 8
K34	BE-20	U28	S4 AXIS 7
	BE-21	U27	S2 AXIS 7
K33	BG-22	A25	S4 AXIS 6
	BG-22	A24	S2 AXIS 6
K32	BG-23	C21	S4 AXIS 5
	BG-24	C20	S2 AXIS 5
K31	BH-14	A18	S4 AXIS 4
	BH-15	A17	S2 AXIS 4
K30	BE-14	C14	S4 AXIS 3
	BE-15	C13	S2 AXIS 3
K29	BH-21	A11	S4 AXIS 2
	BH-22	A10	S2 AXIS 2
K28	BH-16	C7	S4 AXIS 1
	BH-17	C6	S2 AXIS 1
K27	BF-15	C31	S3 AXIS 8
	BF-16	U30	S1 AXIS 8
K26	BE-24	A28	S3 AXIS 7
	BF-14	A27	S1 AXIS 7
K25	BE-22	C24	S3 AXIS 6
	BE-23	C23	S1 AXIS 6
K24	BH-23	A21	S3 AXIS 5
	BH-24	A20	S1 AXIS 5
K23	BF-17	C17	S3 AXIS 4
	BF-18	C16	S1 AXIS 4
K22	BF-19	A14	S3 AXIS 3
	BF-20	A13	S1 AXIS 3
K21	BF-21	C10	S3 AXIS 2
	BF-22	C9	S1 AXIS 2
K20	BF-23	A7	S3 AXIS 1
	BF-24	A6	S1 AXIS 1

J2

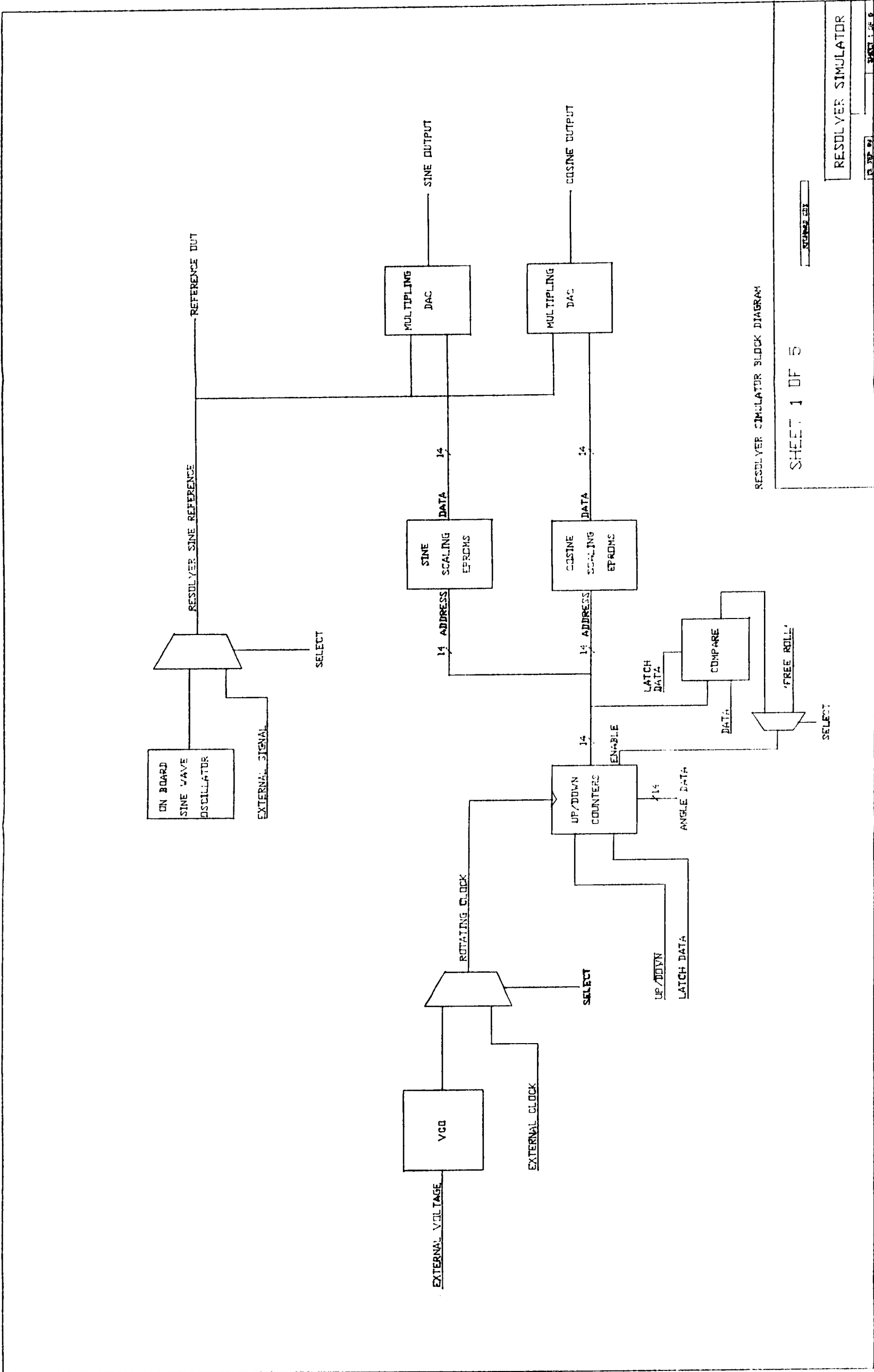


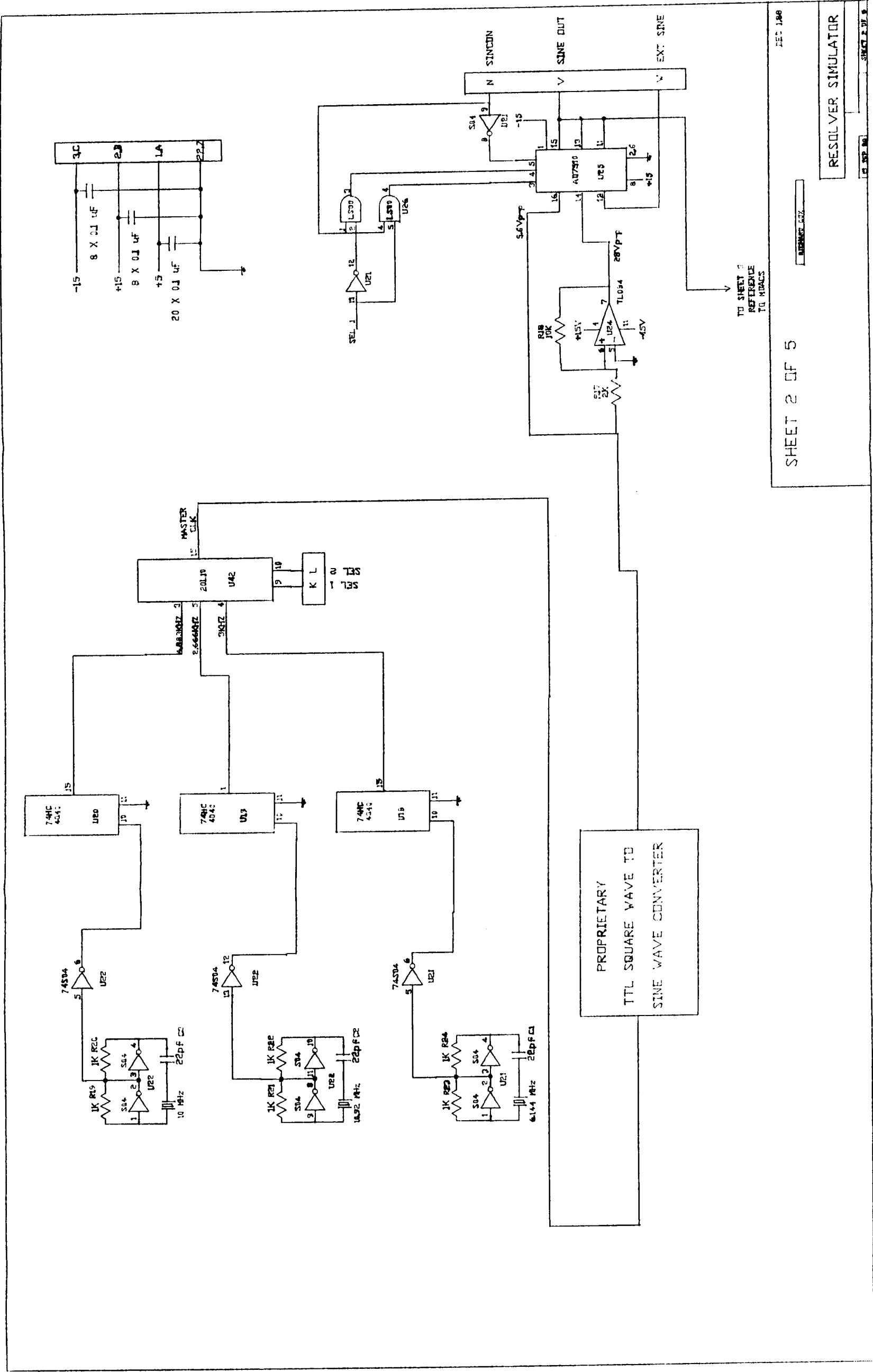


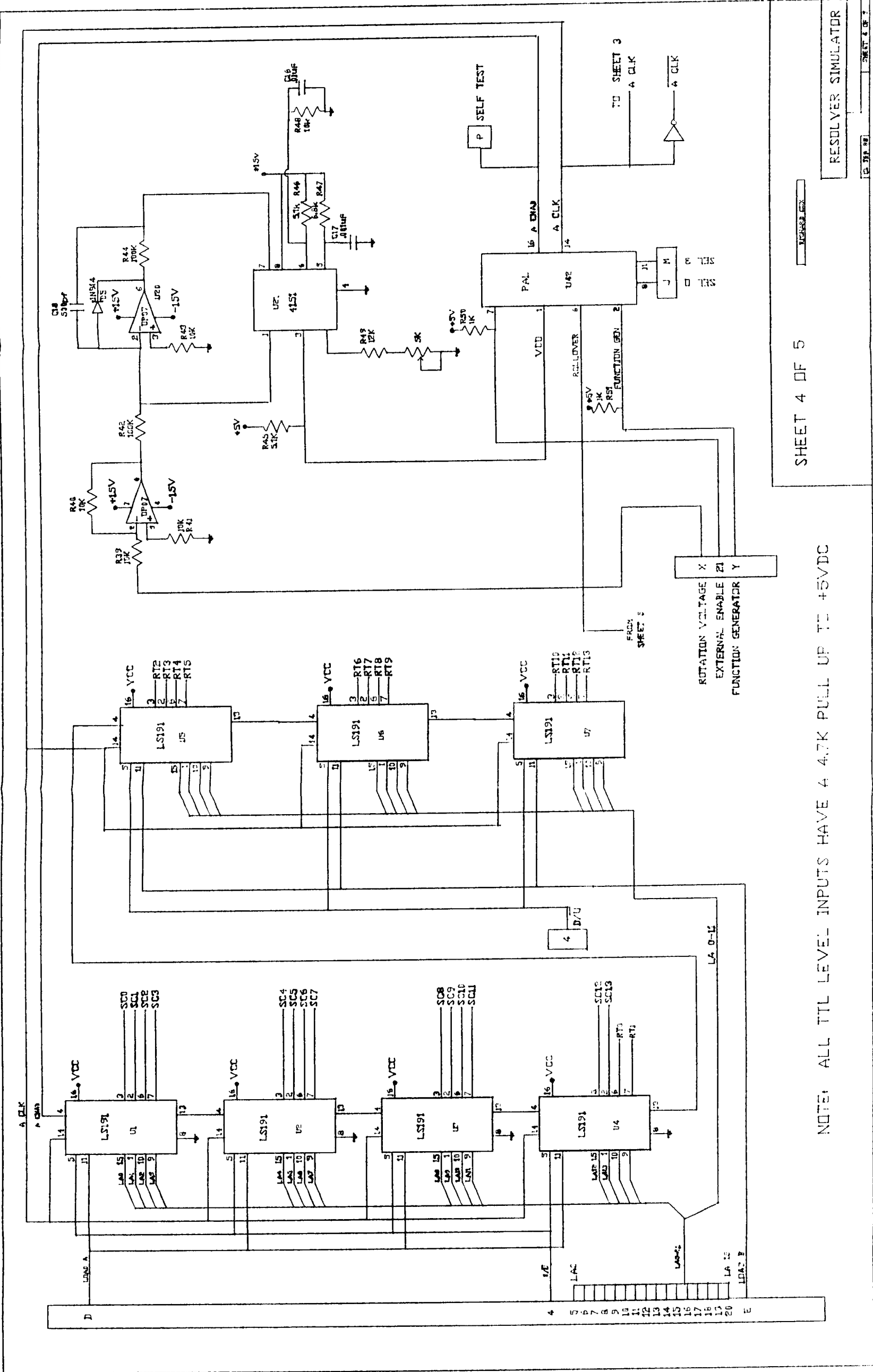
RS232/RS485 BOARD

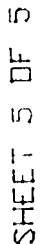
15 JUNE 57

10. 1987









RESOLVER SIMULATOR

9 40 5 1985	88 JUL 88
-------------	-----------

```

10 REM      SCALING EPROM CODE GENERATING ROUTINE FOR COMAU RPT MODULE
20 REM
30 REM      SINE EPROMS
40 REM
50 REM      AUTHOR: RICHARD COX      OCTOBER 14, 1988
60 REM      THIS PROGRAM GENERATES THE BINARY CODE FOR THE SCALING EPROMS
70 REM      THAT ARE IN THE RESOLVER SIMULATOR BOARD.  THESE EPROMS MUST
80 REM      TAKE THE 0 TO 3FFF RAMP OUTPUT OF THE 191 COUNTERS AND CONVERT
90 REM      IT INTO A PROPORTIONAL 0 TO 360 DEGREE SWEEP OF SINE AND COSINE
100 REM     FOR THE MULTIPLYING DAC'S.
110 REM
120 A$ = "000"
130 B$ = "00"
140 C$ = "0"
150 F$ = CHR$(2)
160 G$ = CHR$(3)
170 OPEN"O",#1,"DATA"
180 PRINT #1,USING "&";F$;
190 PRINT
200 FOR N= &H0 TO &H3FFF
210 REM
220 REM   THE DIVIDE BY 4096 INSTEAD OF 4095 WAS NEEDED TO ASSURE PROPER
230 REM   TRANSITION FROM 359 TO 0 DEGREES
240 REM
250 O = (8191 * (SIN(N * (1.570796/4096)))) + &H2000
260 P = (8191 * (COS(N * (1.570796/4095)))) + &H2000
270 O$ = HEX$(O)
280 A = LEN(O$)
290 IF A = 1 THEN X$ = A$ + HEX$(O)
300 IF A = 2 THEN X$ = B$ + HEX$(O)
310 IF A = 3 THEN X$ = C$ + HEX$(O)
320 IF A = 4 THEN X$ = HEX$(O)
330 Z$ = RIGHT$(X$,2)
340 Y$ = LEFT$(X$,2)
350 PRINT USING "&";X$;" ";
360 PRINT #1,USING "&";Y$;" ";
370 PRINT #1,USING "&";Z$;" ";
380 NEXT
390 PRINT #1,USING "&";G$;
400 PRINT

```

```

10 REM      SCALING EPROM CODE GENERATING ROUTINE FOR COMAU RPT MODULE
20 REM
30 REM      COSINE EPROMS
40 REM
50 REM      AUTHOR: RICHARD COX      OCTOBER 14, 1988
60 REM      THIS PROGRAM GENERATES THE BINARY CODE FOR THE SCALING EPROMS
70 REM      THAT ARE IN THE RESOLVER SIMULATOR BOARD.  THESE EPROMS MUST
80 REM      TAKE THE 0 TO 3FFF RAMP OUTPUT OF THE 191 COUNTERS AND CONVERT
90 REM      IT INTO A PROPORTIONAL 0 TO 360 DEGREE SWEEP OF SINE AND COSINE
100 REM     FOR THE MULTIPLING DAC'S.
110 REM
120 A$ = "000"
130 B$ = "00"
140 C$ = "0"
150 F$ = CHR$(2)
160 G$ = CHR$(3)
170 OPEN"O",#1,"DATA"
180 PRINT #1,USING "&";F$;
190 PRINT
200 FOR N= &H0 TO &H3FFF
210 O = (8191 * (SIN(N * (1.570796/4095)))) + &H2000
220 P = (8191 * (COS(N * (1.570796/4095)))) + &H2000
230 P$ = HEX$(P)
240 A = LEN(P$)
250 IF A = 1 THEN X$ = A$ + HEX$(P)
260 IF A = 2 THEN X$ = B$ + HEX$(P)
270 IF A = 3 THEN X$ = C$ + HEX$(P)
280 IF A = 4 THEN X$ = HEX$(P)
290 Z$ = RIGHT$(X$,2)
300 Y$ = LEFT$(X$,2)
310 PRINT USING "&";X$;" ";
320 PRINT #1,USING "&";Y$;" ";
330 PRINT #1,USING "&";Z$;" ";
340 NEXT
350 PRINT #1,USING "&";G$;
360 PRINT

```

```

PARTNO      ;
NAME        20L10 ;
DATE        9/9/88 ;
REV         ;
DESIGNER    RICHARD COX;
COMPANY     TEXAS INSTRUMENTS;
ASSEMBLY    ;
LOCATION     JOHNSON CITY;

```

```

/*****
/*
/*          RPT TESTHEAD CLOCKING SELECT PAL          */
/*
/*          Target Device Type:      20L10          */
/*
*****/

```

/** Inputs (Either from Func. Gen. or SPST to GND **/

```

PIN 1      = VCOCLK      ;      /*CLOCK INPUT FROM VCO*/
PIN 2      = FGENCLK     ;      /*CLOCK INPUT FROM FUNCTION GENERATOR*/
PIN 3      = R4800HZ     ;      /*CLOCK INPUT FOR 4.8 KHZ REFERENCE*/
PIN 4      = R3000HZ     ;      /*CLOCK INPUT FOR 3 KHZ REFERENCE*/
PIN 5      = R2666HZ     ;      /*CLOCK INPUT FOR 2.66 KHZ REFERENCE*/
PIN 6      = ROLLOVER    ;      /*ROLLOVER DISABLE*/
PIN 7      = PLUS5       ;      /*FREE ROTATION*/
PIN 8      = SEL0        ;      /*SELECT LINE 0 FOR VCO/FGEN*/
PIN 9      = SEL1        ;      /*SELECT LINE 1 FOR REF*/
PIN 10     = SEL2        ;      /*SELECT LINE 2 FOR REF*/
PIN 11     = SEL3        ;      /*SELECT LINE 3 FOR AENAB*/

```

/** OUTPUT **/

```

PIN 14     = ACLK        ;      /*UP/DOWN COUNTER CLOCK*/
PIN 15     = REFCLK      ;      /*REFERENCE CLOCK FREQUENCT*/
PIN 16     = AENAB       ;      /*ENABLE LINE FOR 191 COUNTERS*/

```

/** Logic Equations **/

ACLK= VCOCLK & !SEL0 # FGENCLK & SEL0;

REFCLK= R2666HZ & !SEL1 & !SEL2 # R3000HZ & !SEL1 & SEL2
R4800HZ & SEL1 & !SEL2 # SEL1 & SEL2;

AENAB= ROLLOVER & !SEL3 # PLUS5 & SEL3;

```

/*****
/*
/*          RPT TESTHEAD CLOCKING SELECT PAL          */
/*
/*****
/*          Target Device Type:      20L10             */
/*****

```

```

VCOCLK, %2, FGENCLK, %2, R4800HZ, %2, R3000HZ, %2, R2666HZ, %2,
ROLLOVER, %2, PLUS5, %2, SEL0, %2, SEL1, %2, SEL2, %2, SEL3,
%4, AENAB, %2, REFCLK, %2, ACLK;

```

84

```

($990,GLOBALOPT)
($72COL,WIDELIST,MAP,CKPTR,CKOVER,CKINDEX,CKSET,CKTAG)
PROGRAM SELFTSTA$;

```

```

"AUTHOR: RICHARD COX

```

```

"DESCRIPTION: THIS PROGRAM TESTS THE SINE AND COSINE SIGNALS OF THE
"RESOLVER SIMULATOR AT VARIOUS AMPLITUDES.

```

```

($PAGE)

```

```

?COPY T10.TCOMAU.RPT.SRC.COMDECLR

```

```

" THESE TWO ROUTINES ARE IN THE FILE RESINFC

```

```

PROCEDURE SELECT_FUNC(SEL_NO: INTEGER); EXTERNAL;
PROCEDURE ANGLE_ROT(ANGLE, ROTATIONS: WORD); EXTERNAL;
PROCEDURE FINAL_ANG_ROT(F_ANGLE, F_ROTATIONS: WORD); EXTERNAL;

```

```

PROCEDURE SELFTSTA(SUBTEST:BYTE; VAR SUBTESTS_DONE: BOOLEAN);

```

```

VAR

```

```

    PF_FLAG: STRING_4_TYPE;
    ERR_MSG: STRING_4_TYPE;
    MSG: PACKED ARRAY[1..80] OF CHAR;
    KEY_RCVD: CHAR;
    STAT: INTEGER;
    DUMMY: MESSAGE_TYPE;

```

```

ACCESS PORT1_LUNO, INT_DATA, REAL_DATA, DATA_RECORD, MAX_FAIL;

```

```

($PAGE)

```

```

PROCEDURE BLANKMESSAGE(VAR MSG: PACKED ARRAY[1..?] OF CHAR);
BEGIN
    FOR I:= 1 TO UB(MSG) DO MSG[I] := ' ';
END;

```

```

PROCEDURE CHK_ERR(ERROR_CODE: INTEGER;
    MSG: PACKED ARRAY[1..?] OF CHAR;
    ERR_MSG: STRING_4_TYPE;
    VAR MESSAGE: MESSAGE_TYPE;
    PRINT_MSG, FATAL_ERROR: BOOLEAN;
    RPT: INTEGER);

```

```

VAR
    TMSG: CROSS_TYPE;
    NEWMSG: MESSAGE_TYPE;

```

```

ACCESS DATA_RECORD,QUIT_TEST;

```

```

BEGIN"Chk_err
    IF ERR_MSG= 'FAIL' OR DATA_RECORD THEN
        TSTERR(ERROR_CODE,MSG,QUIT_TEST);

    FOR I:= 1 TO 80 DO MSG[I]:= ' ';
    IF (FATAL_ERROR AND ERR_MSG= 'FAIL') OR

```

```

        QUIT_TEST THEN
        BEGIN "Halt path
            PWRDWN;
            ACPS(1,0,60);
            MATRLY([],CLRMAT);
            ESCAPE SELFTSTA;
        END; "Halt path
    END; "Chk_err

```

```

($PAGE)

```

```

PROCEDURE CHECK_FOR_F6(KEY_RCVD: CHAR);

```

```

ACCESS HALTED, QUIT_TEST;
BEGIN"Check for F6
    IF KEY_RCVD = F6 THEN
        BEGIN"Halt path
            PWRDWN;
            MATRLY([], CLRMAT);
            ACPS(1,0,60);
            HALTED:= TRUE;
            SUBTESTS_DONE:= TRUE;
            QUIT_TEST:= TRUE;
            DEPOSIT('TEST ABORTED BY OPERATOR');
            ESCAPE SELFTSTA;
        END;"Halt path
    END;"Check for F6
($PAGE)

```

```

PROCEDURE RES_SIM_CK;

```

"This test checks for proper operation of the resolver simulator.

```

VAR

```

```

    QUAD_ANG: INTEGER;
    DATA1: CROSS_TYPE;
    ANGLE: INTEGER;
    MSGA: PACKED ARRAY[1..50] OF CHAR;
    STRING, TIMERN_STRING, TIMERP_STRING : PACKED ARRAY[1..80] OF CHAR;
    IDEALR, SINAMP, COSAMP, AMPTOL, READING: REAL;
    OVERANGE: BOOLEAN;

```

```

ACCESS CARD1,INT_DATA,REAL_DATA,DATA_RECORD;

```

```

BEGIN"Resolver simulator test
    MATRLY([],CLRMAT);
    ACPS(1,70,1000);"Different from internally generated references
    FOR I:= 8 TO 21 DO CLRLINE(I);
    DISPLAY(9,1,'RESOLVER SIMULATOR CHECK',40,[HI_INT]);
    MATRLY([K037,K038,K170,K171],CLOSE);"Hook up resolver simulator to ACPS
    ANGLE_ROT(#F7FF,0);"Initialize resolver simulator card
    SELECT_FUNC(#055C);
    MATRLY([K093,K097],CLOSE);"Hook up resolver simulator to calibrator
    CALIB(V10V,2.0);
    BLANKMESSAGE(TIMERN_STRING);
    BLANKMESSAGE(TIMERP_STRING);
    ENCODE(TIMERN_STRING, 1, STAT,
        'CFOR4A+1LA+1.0S0HOMOT#0A#0D');"Set up for PRT check
    TIMERSET(TIMERN_STRING);

```

```

FOR I:= 1 TO UB(STRING) DO STRING[I]:= ' ';
ENCODE(STRING,1,STAT,'M1,T0,W3,CT0,DTY50%,AMP2.5V,OFS1.25V,',
      'FRQ1MHZ,D0,#0D,#0A');
FGEN_SET(STRING);
DELAY(1000);
DMMSET('VAR3F3S2',2000);
" COMMANDS TO INITIALIZE MODULE
FOR J:= 1 TO 2 DO
  BEGIN
    ANGLE_ROT(#F7FF,#FFFF);"Initialize resolver simulator card
    SELECT_FUNC(#015C);
    IF J = 1 THEN
      BEGIN
        DISPLAY(11,1,'SINE FREQUENCY CHECK',30,[HI_INT]);
        MATRLY([K049,K148,K000..K009],CLOSE);"Hook up to sine bus
      END
    ELSE
      BEGIN
        DISPLAY(11,1,'COSINE FREQUENCY CHECK',30,[HI_INT]);
        MATRLY([K059,K148,K010..K019],CLOSE);"Hook up to cosine
      END;
    DELAY(1000);
    TIMERIN(READING);
    DELAY(1000);
    IF ABS(READING - 1000.0) > 10.0 THEN
      PF_FLAG:= 'FAIL'
    ELSE
      PF_FLAG:= '    ';
    FOR K:= 1 TO UB(MSG) DO MSG[K]:= ' ';
    IF J = 1 THEN
      ENCODE(MSG,1,STAT,PF_FLAG,' SINE PRT ACT=',
            READING:8:2,' EXP 1000 +/- 10 HZ')
    ELSE
      ENCODE(MSG,1,STAT,PF_FLAG,' COSINE PRT ACT=',
            READING:8:2,' EXP 1000 +/- 10 HZ');
    CHK_ERR(2000,MSG,PF_FLAG,DUMMY,FALSE,FALSE, 0);

    MATRLY([K148],OPEN);
    MATRLY([K140],CLOSE);
    AMPTOL:= 0.10;
    CLRLINE(11);
    FOR I:= 0 TO 6 DO
      BEGIN
        CASE I OF
          0: BEGIN
              ANGLE:= #DFFF;
              SINAMP:= 0.0305;
              COSAMP:= 6.8624;
            END;
          1: BEGIN
              ANGLE:= #EFFF;
              SINAMP:= 6.9047;
              COSAMP:= 0.049;
            END;
          2: BEGIN
              ANGLE:= #F7FF;
              SINAMP:= 4.9114;
              COSAMP:= 4.8705;
            END;
          3: BEGIN

```

```

        ANGLE:= #FBFF;
        SINAMP:= 2.6639;
        COSAMP:= 6.3686;
    END;
4: BEGIN
    ANGLE:= #FDFF;
    SINAMP:= 1.3583;
    COSAMP:= 6.7113;
    END;
5: BEGIN
    ANGLE:= #FEFF;
    SINAMP:= 0.6906;
    COSAMP:= 6.7869;
    END;
6: BEGIN
    ANGLE:= #FFFF;
    SINAMP:= 0.0308;
    COSAMP:= 6.8099;
    END;
END;
ANGLE_ROT(ANGLE,#FFFF);"Set to desired angle
SELECT_FUNC(#015C);
DELAY(150);
FOR K:= 1 TO UB(MSGA) DO MSGA[K]:= ' ';
ENCODE(MSGA,1,STAT,ANGLE:1 HEX);
IF J = 1 THEN
    DISPLAY(13,5,' SINE ANGLE CHECK',
           30,[HI_INT,BEEP])
    ELSE
        DISPLAY(13,3,' COSINE ANGLE CHECK',
               30,[HI_INT,BEEP]);
    DISPLAY(13,25,MSGA,50,[HI_INT]);
    SELECT_FUNC(#055C);
    DELAY(300);
"  COMMANDS TO READ SELFTEST RESULTS
    DMMIN(READING,OVERANGE,INTERNAL_TRIGGER);
    IF J = 1 THEN
        IDEALR:= SINAMP
    ELSE
        IDEALR:= COSAMP;
    IF ABS(READING - IDEALR) > AMPTOL THEN
        PF_FLAG:= 'FAIL'
    ELSE
        PF_FLAG:= ' ';
    FOR K:= 1 TO 80 DO MSG[K]:= ' ';
    IF J = 1 THEN
        ENCODE(MSG,1,STAT,PF_FLAG,
              ' SINE DATA ',ANGLE:1 HEX,' EXP AMPLITUDE ',
              SINAMP:6:3,' +/- ',AMPTOL:5:3,' ACT= ',READING:8:4);
    IF J = 2 THEN
        ENCODE(MSG,1,STAT,PF_FLAG,
              ' COSINE DATA ',ANGLE:1 HEX,' EXP AMPLITUDE ',
              COSAMP:6:3,' +/- ',AMPTOL:5:3,' ACT= ',READING:8:4);
    CHK_ERR(3000,MSG,PF_FLAG,DUMMY,FALSE,FALSE, 0);
    END;
    CLRLINE(13);
    IF J = 1 THEN
        DISPLAY(11,1,'SINE CCW ROTATION',25,[HI_INT])
    ELSE
        DISPLAY(11,1,'COSINE CCW ROTATION',25,[HI_INT]);

```

```

MATRLY([K093,K097],CLOSE);
CALIB(V10V,2.0);
SELECT_FUNC(#015C);
ANGLE_ROT(#FFFF,#FFFF);"Start position
FINAL_ANG_ROT(#C7FF,#FFF7);"End position
DELAY(500);
SELECT_FUNC(#05DC);"Rotate
DELAY(2000);
DMMIN(READING,OVERANGE,INTERNAL_TRIGGER);
IF ABS(READING - 4.88) > 0.20 THEN
    PF_FLAG:= 'FAIL'
ELSE
    PF_FLAG:= ' ';
FOR K:= 1 TO 80 DO MSG[K]:= ' ';
IF J = 1 THEN
    ENCODE(MSG,1,STAT,PF_FLAG,
        ' SINE CCW ROTATION ACT= ',READING:8:4,
        ' EXP 4.88 +/- 0.20')
ELSE
    ENCODE(MSG,1,STAT,PF_FLAG,
        ' COSINE CCW ROTATION ACT= ',READING:8:4,
        ' EXP 4.88 +/- 0.20');
CHK_ERR(4000,MSG,PF_FLAG,DUMMY,FALSE,FALSE, 0);
CLRLINE(11);
MATRLY([K093,K097],OPEN);
MATRLY([K092,K098],CLOSE);
IF J = 1 THEN
    DISPLAY(11,1,'SINE CW ROTATION',25,[HI_INT])
ELSE
    DISPLAY(11,1,'COSINE CW ROTATION',25,[HI_INT]);
SELECT_FUNC(#015C);
ANGLE_ROT(#C7FF,#FFF7);"Start position
FINAL_ANG_ROT(#FFFF,#FFF7);"End position
DELAY(500);
SELECT_FUNC(#01CC);"Rotate
DELAY(100);
DMMIN(READING,OVERANGE,INTERNAL_TRIGGER);
IF J = 1 THEN
    BEGIN
        IF ABS(READING - 0.055) > 0.10 THEN
            PF_FLAG:= 'FAIL'
        ELSE
            PF_FLAG:= ' ';
        FOR K:= 1 TO 80 DO MSG[K]:= ' ';
        ENCODE(MSG,1,STAT,PF_FLAG,
            ' SINE CW ROTATION ACT= ',READING:8:4,
            ' EXP 0.055 +/- 0.10');
        CHK_ERR(5000,MSG,PF_FLAG,DUMMY,FALSE,FALSE, 0);
    END
ELSE
    BEGIN
        IF ABS(READING - 6.823) > 0.20 THEN
            PF_FLAG:= 'FAIL'
        ELSE
            PF_FLAG:= ' ';
        FOR K:= 1 TO 80 DO MSG[K]:= ' ';
        ENCODE(MSG,1,STAT,PF_FLAG,
            ' COSINE CW ROTATION ACT= ',READING:8:4,
            ' EXP 6.823 +/- 0.20');
        CHK_ERR(5000,MSG,PF_FLAG,DUMMY,FALSE,FALSE, 0);
    END

```

```

        END;
        MATRLY([K092,K098],OPEN);
        MATRLY([K093,K097],CLOSE);
        IF J = 1 THEN
            MATRLY([K049,K140,K000..K009],OPEN)"Open up sine bus
        ELSE
            MATRLY([K059,K140,K010..K019],OPEN);"Open up cosine
        CLRLINE(11);
    END;
    MATRLY([],CLRMAT);
END;
($PAGE)

($PAGE)
BEGIN " SELFTSTA
    MATRLY([],CLRMAT);
    SUBTESTS_DONE      := FALSE;
    MAX_FAIL           := 30;

    CLRLINE(10); "For subtest title displays
    CASE SUBTEST OF
        1: RES_SIM_CHK;
        OTHERWISE SUBTESTS_DONE:=TRUE
    END;"Case
    IF DATA_RECORD THEN SKIPLINE; "Between subtest data groups
    MATRLY([],CLRMAT);

END; "SELFTSTA

BEGIN {DUMMY PROGRAM}
    {$NO OBJECT}
END. {DUMMY PROGRAM}

```

VITA

Richard H. Cox was born in Bristol, Tennessee on December 28, 1957. He graduated from Sullivan Central High School in June 1976. The following September he entered Tennessee Technological University, Cookeville. Richard was employed as a co-op student by Tennessee Valley Authority and as a teacher's assistant by Tennessee Technological University before receiving a Bachelor of Science degree in Electrical Engineering in March, 1982.

Upon graduation, Richard accepted a permanent position with Texas Instruments in Dallas, Texas. Continuing as a full time employee, Richard began study toward a Master's degree from Southern Methodist University in January 1984. Upon a transfer to Texas Instruments in Johnson City, Tennessee, Richard transferred to the University of Tennessee in September 1986. He received the Master of Science degree in Electrical Engineering in May, 1990. Richard was able to apply his experience as a test engineer to help complete the thesis requirements.