DRC190 Remote Control System

Installation, Operation & Maintenance

Revised 14 November 1985

Hallikainen & Friends, Inc.
141 Suburban Road
San Luis Obispo, CA 93401-7590
(805) 541-0200

Taυle of Contents

Adjustments

Theory of Operation

Component Placement Drawings

Schematic Diagrams

Warranty

Hallikainen & Friends, a California corporation (CORP) hereby warrants, subject to the conditions herein below set forth, that should this product prove defective by reason of improper workmanship or defective materials within one (1) year from date of original purchase, Corp will repair or, at its option, replace the defective unit without charge for either parts or labor.

Contitions of Warranty
1. Notice. Purchaser shall notify Corp at its principal place of business by telephone within three (3) days after malfunction of the product. Time is deemed of essence.
2. Proper Delivery. The unit must be shipped, freight prepaid, or delivered to the manufacturing plant of Corp located at San Luis Obispo, CA 93401, in either its original package or a similar package affording an equal degree of protection.
3. The unit must not have been previously altered, repaired or serviced by anyone other than Corp, except for the replacement of plug in components with electrically identical components, or routine adjustments as outlined in the accompanying manual. Upon repair by customer, the Corp shall replace defective plug in parts returned to Corp, but shall not be liable for any labor expenses incurred in a field repair.
4. The serial number on the unit must not have been altered or removed; the unit must not have been subject to accident, misuse, or operated contrary to the instructions contained in the accompanying manual.
5. This warranty does not cover peripheral devices of other manufacturers supplied as part of a system by Corp (such as CRT terminals, printers, etc.); Purchaser's only remedies for malfunction with respect to such devices are with the equipment's manufacturer.
6. This warranty does not cover transporation expenses to and from service facility.
7. This warranty is in lieu of any other oral, written, or implied warranty, whether made by salesmen, agents, or other representatives of Corp.

Except to the extent prohibited by applicable law, all implied warranties made by Corp in connection with the product, including the warranty of merchantability are limited in duration to a period of one (1) year from the date of original purchase, and no warranties, whether expressed or implied, including said warranty of merchantability shall apply to this product after said period. Should this product prove defective in workmanship or material, the consumer's sole remedy shall be such repairs or replacements as hereinabove expressly provided; and under no circumstances shall Corp be liable for any loss or damage, direct or consequential, arising out of the use or inability to use this product.

FCC Notice

Warning: This equipment generates and uses radio frequency energy and if not installed and used properly, i.e. in strict accordance with the instructions manual, may cause harmful interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment.

Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

Direct Connect Modem FCC Notice

FCC rules and regulations under part 68, requires the following information be provided to the user of FCC-registered terminal equipment such as the Cermetek CH1770 (used on the DRC190 direct connect modem card).

Section 68.100 GENERAL

Terminal equipment may be directly connected to the telephone network in accordance with the rules and regulations...of this part.

Section 68.104 STANDARD PLUGS AND JACKS

(a) General

"Except for telephone company provided ringers, all connections to the telephone network shall be made through standard (USOC) plugs and standard telephone company provided jacks, in such a manner as to allow for easy and immediate disconnection of the terminal equipment. Standard jacks shall be so arranged that if the plug connected thereto is withdrawn, no interference to the operation of equipment at the customer's premises which remains connected to the telephone network shall occur by reason of such withdrawal."

Section 68.106 NOTIFICATION TO TELEPHONE COMPANY

"Customers connecting terminal equipment or protective circuitry to the telephone network shall, before such connection is made, give notice to the telephone company of the particular line(s) to which such connection is to be made, and shall provide the telephone company the FCC Registration Number and Ringer Equivalence of the registered terminal equipment or protective circuitry. The customer shall give notice to the telephone company upon final disconnection of such equipment or circuitry from the particular line(s)."

Section 68.108 INCIDENCE OF HARM

"Should terminal equipment or protective circuitry cause harm to the telephone network, the telephone company shall, where practicable, notify the customer that temporary discontinuance of service may be required; however, where prior notice is not practicable, the telephone company may temporarily disconnect service forthwith, if such action is reasonable in the circumstances. In case of such temporary discontinuance, the telephone company shall (1) promptly notify the customer of such temporary discontinuance, (2) afford the customer the opportunity to correct the situation which gave rise to the temporary discontinuance, and (3) inform the customer of the right to bring a complaint to the Commission pursuant to the procedures set forth in Subpart E of this Part."

Section 68.216 REPAIR OF REGISTERED TERMINAL EQUIPMENT AND REGISTERED PROTECTIVE CIRCUITRY

"Repair of registered terminal equipment and registered protective circuitry shall be accomplished only by the manufacturer or assembler thereof or by their authorized agent; however, routine repairs may be performed by a user, in accordance with the instruction manual if the applicant certifies that such routine repairs will not result in non-compliance with the rules in Subpart D of this Part."

Section 68.218(b) ADDITIONAL INSTRUCTIONS TO USER

1. "...registered terminal equipment or protective circuitry may not be used with coin lines."

2. "...when trouble is experienced, the customer shall disconnect the registered equipment from the telephone line to determine if the registered equipment is manfunctioning, and...if the registered equipment is malfunctioning, the use of such equipment shall be discontinued until the problem has been corrected."

3. "...the user must give notice to the telephone company in accordance with the requirements of Section 68.106..." for connecting the H&F Direct Connect Modem (Cermetek CH1770) to the telephone line.

NOTE:

As part of the H&F agreement with Cermetek, repairs to the H&F direct connect modem board should be referred to Hallikainen & Friends, NOT to Cermetek.

Copyright Notice

Information in this manual is copyrighted and printed with permission of the copyright owner. Copyright owners include: Microsoft Corporation, Cermetek Microelectronics and Matrix Corporation.

Software in EPROM form is protected by copyright (Microsoft) and trade secret (Hallikainen & Friends).

Copying of this manual or the provided software is prohibited except as allowed by copyright law for backup purposes.

System Registration

To comply with software licensing agreements and to insure that the user receives notices regarding software and hardware updates, please complete the form below. Leave this page in the instruction manual, but photocopy it, sign the photocopy, and return the signed copy to Hallikainen & Friends. Thankyou!

Non-Disclosure Agreement

The party below agrees that it is receiving a copy of Hallikainen & Friends DRC190 Firmware for use on a single computer only, as designated below. The party agrees to fill out and mail in this registration form before making use of Hallikainen & Friends DRC190 Firmware. The party agrees that all copies will be strictly safeguarded against disclosure to or use by persons not authorized by Hallikainen & Friends to use the DRC190 Firmware, and that the location of all copies will be reported to Hallikainen & Friends at its request. The party agrees that copying or unauthorized disclosure will cause great damage to Hallikainen & Friends, and this damage is far greater than the value of the copies involved. The party agrees that this agreement shall inure to the benefit of any third party holding any right, title or interest in the Hallikainen & Friends DRC190 Firmware or any software from which it was derived.

DRC190 Unit Serial Number: ________________

User Name: ________________________________

User Company: _____________________________

Company Address: __________________________

Company City, State, Zip: _________________

Company Country: __________________________

Company Telephone Number: _________________

User Signature: ___________________________

Title: ____________________________________

Date: _____________________________________

Please return the signed copy of this form to:

Hallikainen & Friends, Inc.
141 Suburban Road
San Luis Obispo, CA 93401-7590
USA

Introduction

The DRC190 is a data acquisition system optimized for use in the broadcast industry. The software and hardware have been generalized as much as possible, allowing any unit to act as a remote or control unit.

The system can be operated with anywhere between 1 and 100 units in a system. A one unit system might be used to automatically log and control a local transmitter. A two unit system could be used to manually or automatically log and control a single remote transmitter site. A three or four unit system could be used to automatically or manually log and control an AM/FM or AM/FM/TV system with separate transmitter sites. Larger systems could be used to monitor broadcast microwave networks, or other systems with a large number of remote unattended sites.

The basic DRC190 box is a specialized microcomputer. It includes the standard microcomputer components (processor, memory and Input/Output) plus a 1200 bit/second half duplex modem and optional analog to digital converters.

In a simple remote control system, the "studio unit" includes a DRC190 with the processor, memory, modem, and a front panel display and keyboard. The operator keys in the desired site and channel that s/he wishes to check or adjust.

A remote site includes all the functions of a "studio unit" plus up to ten analog to digital converter boards. These boards select the appropriate metering sample and convert it to digital for use by the microprocessor. In addition, the processor can instruct the A/D boards to output control signals to adjust the external equipment.

Since the sample voltages provided to the DRC190 are proportional to the actual parameter to be measured, a scaling factor is established for each channel of metering. This scaling factor is established in the calibration procedure, and is stored in non-volatile memory at the remote site. Non-volatile memory also holds setup information regarding the unit (site number, communication speeds, labels and units for each metering channel, and a Morse Code identifier if ordered). Non-volatile memory (EEPROM) can also hold about 1 Kbyte of a Basic applications program. This program is automatically loaded and run on system reset. This program might be a simple logging program, or a "boot" loader program that loads a larger program from disc.

The sample voltages provided to the DRC190 can have up to +/-100 volts of common mode voltage. The differential voltage must be limited to less than 2 volts. Revision B analog to digital converter boards have a provision to install a voltage divider after the analog multiplexer allowing a higher differential sample voltage. Such a voltage divider can be added to the A/D board if required to avoid building separate voltage dividers for each sample. The temperature coefficeint of the voltage divider resistors will decrease the stability of the indicated sample, so low tempco resistors should be used. The control outputs are open collector and can drive 500 mA with an open circuit voltage of less than 30 volts.

The DRC190 units communicate "half duplex". All units transmit and receive on the same frequencies. Unless a unit has a command or metering information to send, it stays off the communications line and listens. A unit that has information to send waits for its allocated time period in the anti-contention scheme and then brings up its audible transmit carrier. After allowing time for all units to detect the carrier, the carrier is keyed (FSK) with the data to be transmitted. The carrier is then shut down, leaving the communications line free for other units to transmit. Each character of a message includes parity error checking, and the entire message is checked for errors using a checksum.

The DRC190 can use almost any audible communications link. These include standard "3002" data circuits, microwave and broadcast subcarriers, and separate radio links. In radio linked systems, all sites can transmit and receive on the same frequency using the same anti-contention firmware as the audio communications portion of the DRC190 system.

Since a voice-grade communications link is used by the DRC190, an intercom feature was included. When the COM button on the front panel is pressed, the DRC190 sends an FSK code telling other units to enable their front panel speakers. The operator is then allowed to talk into the front panel speaker for up to 30 seconds. That voice information will be heard from the speaker of each other unit in the system.

When the DRC190 is used in radio linked systems that require identification, a Morse code identifier is included. This firmware generates the FCC required 20 WPM 750 Hz Morse code station identification. The number of minutes between station identifications and the actual identifying code are programmable by the user through the DRC190 front panel.

Each DRC190 includes a Basic interpreter. This program, along with the RS-232 port included in each unit, allows the user to write programs in Basic that can display the readings on a CRT terminal, log the readings on a printer, or log and control the operation of the station. If a printer and CRT are used, the CRT must have a peripheral port capable of driving the printer, and the peripheral port enable and disable control codes must be programmed into the DRC190.

The DRC190 also includes optional subcarrier generation and demodulation (20 KHz to 200 KHz) and a serial interface to the Commodore 1541 disc drive for program storage.

The DRC190 also includes ports for controlling IEEE488 automatic test equipment, and status panels. At this writing, the firmware to support these items has not been written. Registered system users will be notified when that firmware is available.

Review of FCC Rules

The FCC is continually making updates to the rules regarding remote control and automatic control of broadcast stations. A good review of the appropriate rules is suggested prior to beginning the installation of a new transmitter control system.

The installer should review a current copy of the following sections of the FCC rules:

73.51	AM Station Power Determination
73.57	AM Remote Reading Antenna or Common Point Ammeters
73.62	AM Directional Antenna System Tolerances
73.69(a)(2)	Antenna Monitor Requirements with Remote Control
73.140	AM Station ATS Authorization
73.142	AM Station ATS Requirements
73.144	AM Station ATS Fail Safe (Transmitter Shutdown)
73.146	AM Station ATS Monitor & Alarm Points (Operator Required!)
73.267	FM Station Power Determination
73.293	Use of FM Subcarriers
73.295(a)	Definition of SCA
73.319	FM Subcarrier Tehnical Standards
73.340	FM Station ATS Authorization
73.342	FM Station ATS Requirements
73.344	FM Station ATS Fail Safe (Transmitter Shutdown)
73.346	FM Station ATS Monitor & Alarm Points (Operator Required!)
73.540	NCE FM Station ATS Authorization
73.542	NCE FM Station ATS Requirements
73.544	NCE FM Station ATS Fail Safe (Transmitter Shutdown)
73.546	NCE FM Station ATS Monitor & Alarm Points (Operator Required)
73.663	TV Station Power Determination
73.665	Use of TV Subcarriers
73.667(a)	Definition of TV SCA
73.932	EBS Monitor and Generator Requirements
73.933	EBS Operation During National Emergency
73.935	EBS Operation During State or Local Emergency
73.936	EBS Operation During State Emergency
73.937	EBS Operation During Local Emergency
73.940(j)	EBS Generator Switch Guard Requirement
73.1215	Specifications for Indicating Instruments
73.1230	Station and Operator License Posting Requirements
73.1400	Remote Control Authorizations
73.1410	Remote Control Operation
73.1560	Operating Power Tolerance
73.1570(b)(2)	Modulation Limit on FM Stations Using Subcarriers
73.1580	Required Transmission System Inspections
73.1690(d)(2)	Commencement of Remote Conrol Operation
73.1690(e)(5)	Installation or Replacement of Subcarrier Generator
73.1800	General Requirements Related to Station Log
73.1820	More Station Log Requirements
73.1840	Retention of Station Logs
73.1860	Transmitter Duty Operators
73.1870(c)(3)	Chief Operator Review of Station Log
73.3544(b)(4)	Change in Control Point
73.4097	EBS Attention Signals on Automated Stations
74.24	Short-term Operation of Auxiliary Stations

74.402(a)(7)	Frequencies Reserved For Operational Communications
74.436(c)(7)	Group P Frequencies Licensed for Remote Control or ATS
74.434	Remote Control of Remote Pickup Stations
74.451(a)	Use of Equipment Type Accepted Under Part 90 Acceptable
74.462(b)	TRL Transmitters Limited to 1.5 KHz Deviation
74.464	TRL Frequency Tolerance
74.465	TRL Frequency Measurements
74.467	Posting of Station License
74.482(b)	Hourly Identification of TRL Transmitters
74.482(d)	International Morse Code Identification of TRL Transmitters
74.531(d)	Use of Subcarriers for Remote Control on Aural STLs
74.533(a)	Remote Control of Aural STLs
74.535(b)	Maximum Deviation For Subcarriers on Aural STLs
74.634	Remote Control of TV STLs
74.734(a)	Operator Required at Control Point for LPTV Local Origination
73.933	Remote Control of ITFS Stations

Below are a set of comments regarding the FCC rules on remote control, automatic logging, and ATS. These are comments only, and should be considered in designing a system. These are not legal opinions, and are offered with no guarantee.

The FCC has recently made substantial revision of the rules regarding the remote control of broadcast transmitters. This is covered in Docket 84-110, RM-3406, and was released 21 November 1984. This revision eliminated most of the "how to" provisions of the old rules. Quoting from the conclusions of the Report & Order, "In general, licensees may operate their station by remote control using any method that assures that: 1) an operator is on duty, 2) the transmitting system operates properly, and 3) the Commission can contact station personnel during hours of operation." In addition, the new rules liberalize the fail-safe requirements. Quoting from the discussion in the Report and Order:

> The Commission shares the concern of the licensees in that short-term losses of control or telemetry should not be cause for immediate termination of operation. Accordingly, the new rules will permit continued operation following a loss of transmitter control, pending repair of the control circuits, as long as the station continues to operate properly. Loss of telemetry, on the other hand, means that the transmitter parameters cannot be monitored remotely. Therefore, the amended rules will require terminaltion of use of remote control within three hours after the detection of the telemetry failure. Although CBS recommended six hours, the majority of the commenters on this subject concurred with the proposed three hour period.

With this material in mind, the following comments on the above listed sections of the FCC rules are offered.

73.51 & 73.57 AM Station Determination of Power & Remote Reading Ammeters

This section requires that AM stations determine power by the "direct" method (unmodulated antenna or common point current squared times the antenna or common point resistance), unless (on a temporary basis) you have reason to suspect a change in resistance or suspect a defective current meter. Under these temporary circumstances, the indirect (final plate voltage times final plate current times final amplifier efficiency) method may be used. Note that the rule specifies unmodulated current. We might assume that if the current indication does not change with modulation, the reading could indeed be taken with modulation. This pretty much eliminates the use of thermocouple ammeters for remote indications. Due to the relative inaccuracy of thermocouple ammeters, and the non-linearity of their indications, they are rarely used for remote metering, although they are still quite commonly used for the "local" ammeter. "Diode" meters, such as the Delta Electronics TCA series offer accuracy, linear scale, remote output, and insensitivity to modulation. It is suggested that such a meter be used to drive a remote control system. Note, however, that the Delta meter responds to the "average" current (as opposed to the RMS response of the thermocouple type), and will vary with modulation if the transmitter exhibits less than perfect carrier amplitude regulation ("carrier shift"). This shift in the carrier amplitude may be due to the transmitter itself, or could possibly be introduced by the antenna system not presenting a constant load to the transmitter over the sideband spectrum. It is unlikely that there will be absolutely no variation in antenna current with modulation, so the question becomes, how much is allowable? Section 73.57(d)(3) requires remote ammeters to agree with the local ammeter within 2%. Section 73.1820 requires that parameters that are affected by modulation be read without modulation. Read literally, this might mean that the antenna current must be read without modulation if the indication deviates .005% with modulation. Of course, no reading frequency is specified any more, but, we might assume (no guarantees here!) that if the deviation of the remote ammeter due to modulation is less than 2%, it is still within the required accuracy of the remote meter. Further, if the antenna current is kept inside a "tighter window" than the FCC specified 90 to 105 percent power such that the possible meter inaccuracy still insures the station is operating within the licensed parameters, the licensee has made a reasonable effort to keep the station within the authorized power limits (no guarantees here either!).

Getting back to section 73.57, there are several ways of measuring the antenna (or common point) current of a station. Although not specifically authorized by the FCC rules, an interesting method has been suggested by Moseley Associates (Goleta, CA). Moseley has suggested actually taking a voltage sample from the antenna and converting it to a DC sample. Hoping that the antenna (or common point) is a constant impedance, the voltage sample will track the current exactly. This appears valid, since we are also assuming that the antenna resistance is constant in our direct power formula. In addition, 73.57(a)(3) authorizes "capacity coupling to radio frequency current sensing device for providing direct current to indicating instrument". We might assume that the "current sensing device" is the antenna, since its impedance can be thought of as a current to voltage converter. We then capacitively couple the voltage off this "current sensing device", convert the sample to DC, and send it to the remote indicator. In any case, it'd appear that another "direct method" of determining power would be the antenna voltage squared and divided by the antenna resistance. This should be just as accurate as the "standard" direct method, if we can accurately measure the antenna voltage.

The Delta TCA series ammeters fall under 73.57(a)(4) as a current transformer. These are a favorite method of getting the local and remote antenna or common point current indications while minimizing the effects of modulation.

Another fairly common method of getting base remote base or common point indications in stations that are directional at least part of the time is to scale the indications of the antenna monitor. When a directional station operates non-directionally, the sampling loop or current transformer at the base of the tower will typically have a current that remains proportional to the base current of the tower. The antenna monitor then converts this to a DC sample that may be sent to a remote indicator. In addition, some DA stations have installed a current sampling transformer immediately prior (transmitter side) to the local common point ammeter. This current transformer then drives an extra input on the antenna monitor. Section 73.57(a)(6) requires independent calibration of such an indication. This may be satisfied with a calibration pot or software calibration in a remote control, or with a separate "pattern select" on the antenna monitor, devoted to base or common point current measurement. Using a sampling transformer on the common point of a directional antenna also allows monitoring of the phase and amplitude ratios of the common point current referenced to each tower in the system. This may aid in troubleshooting an array.

Finally, note that 73.57(d) requires that remote reading ammeters be calibrated against the actual ammeter (not another remote reading meter) as often as necessary to insure the required accuracy of 2%. Note that the accuracy requirements of the actual ammeter (as outlined in 73.1215(a) assuming linear scales for the moment) are 2% of full scale with the full scale reading not more than five times the minimum normal indication. For example, a current of 1 ampere might be read on a 5 ampere meter with an accuracy of 2% of full scale (plus or minus 2% of 5 amperes). The potential error in the indication at the 1 ampere point is plus or minus 0.1 ampere, or 10% of the indication. It may be difficult to get a remote meter to track the inaccuracies of the actual meter within the required 2% over a wide range of currents. Stations that have major changes in operating power will probably have to allocate different remote control channels to the antenna (or common point) current or provide calibration pots to compensate for the non-linearity of either or both of the actual or remote ammeters.

73.62 Directional Antenna System Tolerances

A reminder that the required tolerances of a directional antenna system are plus or minus 5% and plus or minus 3 degrees, unless tighter tolerances are specified on the license (a critical array). The rules specify that the station must maintain the "relative amplitudes" of the base currents and the antenna monitor current indications within 5% of the values specified on the license. The term "relative amplitudes" normally indicates that the ratio of the base currents and the ratio of the antenna monitor current indications (referenced to the reference tower) shall be maintained within 5%. It is fairly common for DRC190 programs operating on directional antennae to calculate the current ratios and deviation from licensed ratios for the antenna monitor current indications. These deviations can be checked against limits, logged, displayed, and minimum, maximum, and average deviations can be accumulated to aid in adjustment of the directional array.

73.69(a)(2) Antenna Monitor Requirements for Remote Control

You must use an antenna monitor approved for remote operation. You cannot add a bunch of relays and pull sample voltages off a "local only" antenna monitor to drive a remote control or extension metering.

73.140 ATS Authorizations for AM Stations

An AM station using a non-directional antenna may operate with ATS after receiving authorization from the FCC. The authorization will be granted upon the station licensee submitting an informal application showing that all ATS requirements have been met. Several comments on ATS follow. In general, H&F does not feel that ATS is worth the trouble. When originally introduced, ATS reduced operator licensing requirements (an operator is still required!), reduced logging requirements, and reduced inspections of the transmitter site. All of these requirements have been reduced to the same level as authorized for ATS for those stations not operating with ATS. Further, ATS is required to shut the station down if it thinks the station is operating with excessive power or modulation. Under manual operation, the actual shut down decision is left to the operator. The operator may note that the "direct power" is too high, but the indirect power appears proper, possibly indicating a defect in telemetery data or antenna current sampling. Since the station is authorized to operate up to three hours with bad telemetry, and by instantly switching to the indirect method of power determination the operator finds that the station is probably not exceeding the authorized power, a shut down could be put off for up to three hours, allowing the chief operator to get to the transmitter site to see what's wrong. No lost air time!

The idea of ATS appears to be to make the transmitter adjust itself and send alarms to the control point when it is unable to accomplish the required adjustment. This does allow a form of remote control with no telemetery, merely control and "status", although the status indications are generally not available directly from the transmitter. Some transmitters do automatically adjust power, but they sample the power at the transmitter output instead of the antenna input. They do not generally include an alarm indicating that the adjustment could not be accomplished. A transmitter meeting the modulation control requirements for ATS is rare.

73.142 AM Station ATS Requirements

The ATS system is required to measure, adjust, alarm, and initiate a transmitter shut down procedure based upon the direct method of determining power, based on the antenna current as read by a remote meter.

The ATS system is required to measure, adjust, alarm, and initiate a transmitter shut down procedure based upon the percentage modulation. Note that most existing modulation monitors do not meet the peak count requirements of ATS. Delta Electronics offers the AMC-1 modulation controller, that includes all the required metering, control and adjustment circuitry.

If the station has parameters that are restricted during certain times of the day (ie, daytime only station, or reduced power at night, or reduced power presunrise authorization, or reduced power post sunset authorization, current rules do not authorize ATS on directional arrays), the ATS must prevent the station from operating with an unauthorized power at a particular time. The ATS must initiate all required mode switching (power changes), terminate the transmission at the specified time, and prevent transmission before the authorized time. Note that 73.140(e) requires manual activation of the system at the beginning of operation (let the machine be sure there is an operator present).

73.144 ATS Fail Safe

The transmitter must be shut down within three minutes of any of the following:

Power over 105% of the licensed power
Modulation excessive
Failure of the mode switching clock
Failure of the "control link" from the monitoring and alarm point
Failure in any required alarm function
Loss of any of the required samples

Operation of the station may be resumed under "manual" direct or remote control if it is determined that the failure was in the ATS and that the station is operating within the terms of the license.

73.146 ATS Monitor and Alarm Points

A station employee holding at least a Restricted Radiotelephone Operator Permit shall be on duty at one of the authorized monitoring and control points at all times the station is in operation. ATS does not reduce or eliminate operator requirements! Further, operators need to be licensed and need to be station employees. The Commission wants the station licensee to have direct "employer/employee" control over each transmitter operator, insuring that the instructions of the licensee are indeed carried out. The use of hotel clerks, answering services or alarm companies to monitor transmitter operation may be questionable. Does the FCC go by the IRS determination of who is an employee? Does the determination that an answering service operator is indeed a station employee require the station to pay that operator according to the minimum wage requirements?

The FCC must authorize all ATS alarm and monitoring (other than an authorized remote control point or the transmitter) points prior to their being used. The FCC wants to be able to reach an operator who has control over the transmitter at any time the transmitter is operating.

73.146 also outlines the various alarms that are required under ATS operation.

One particular requirement that is difficult to meet with an ATS monitoring and alarm point that is not at the studio (program origination point) is covered in 73.146(f). An ATS operator must be able to properly respond to an EBS alert. This requires initiating EBS tones, airing required announcements, and (for participating stations) airing the emergency program.

73.267 FM Station Power Determination

The power of an FM station may be determined using either the direct or indirect method. In the direct method, a transmission line meter indicating the power out of the transmitter is read. In the indirect method, the same procedure outlined for AM indirect power determination is used. Watch the transmission line meter calibration requirements carefully.

73.293 FM Subcarriers

This section was included in this discussion since it is common practice to return telemetry for FM stations on a subcarrier on the broadcast carrier.

FM subcarriers can be used for a variety of purposes, including stereo, quadrophonic, noise reduction circuit activation, program identification, remote cueing messages, station control and telemetry, and Subsidiary Communications services (SCA). Note that station control and telemetry is not considered SCA. SCA is limited to "multipoint" broadcasts, such as backgroud music, reading services, broadcast data services, traffic control signal switching, paging, utility load management, etc. Control and telemetry for the station's own use do not fall into the SCA category.

73.319 FM Multiplex Subcarrier Technical Standards

Note that this section now allows any form of AM or "angle" modulation, including standard AM, DSBSC, SSB, FM, PM, or FSK.

The carrier and all significant sidebands of the subcarrier must be between 20 KHz and 99 KHz during monaural broadcasts, and between 53 KHz and 99 KHz during stereo broadcasts, except that stations within 320 kilometers of Mexico are currently restricted to a maximum frequency (carrier and sidebands) of 75 KHz, until an agreement extending the limit is made.

Depending upon subcarrier frequency, the number of subcarriers, whether the main program is stereo or not, there is a maximum permissable deviation of the main FM carrier by each subcarrier (subcarrier injection), and a maximum sum of all subcarrier injections. This is covered in 73.319(d) and 73.1570(b).

Note that the crosstalk requirements of 73.319(e) apply to telemetry subcarriers as well as SCA subcarriers. Check them!

You can plug in a new subcarrier generator if no electrical or mechanical modification of the transmitter or exciter is required, according to 73.193(g). Upon plugging in such a subcarrier generator, measurments must be made to show that the station still meets all requirements (especially cross-talk into the main and stereo subchannels).

73.340, 73.342, 73.344, 73.346, 73.540, 73.542, 73.544, 73.546 ATS Requirements for FM and NCE FM Stations

These requirements are pretty much the same as those for non-directional AM stations, with the appropriate changes. The power can be determined by the direct or indirect method, the modulation limits are different.

If a modulation controller is desired, the Delta Electronics FMC-1 is available.

73.663 TV Station Power Determination

Since the average and RMS power of the visual transmitter varies with picture content, visual power is determined with a peak reading transmission line meter. This meter must be calibrated as often as necessary, and at least every six months. The power of the aural transmitter may be determined by either the direct or indirect method. If the direct method is used, the transmission line meter must be calibrated so that 100% aural ERP (including aural antenna gain and transmission line losses) corresponds to 22% of the authorized peak visual ERP. This must be calibrated as often as necessary, or at least every six months. There are various requirements for "red markers" indicating limits to the visual power deviation from licensed power. I wonder how many remote controls indicating visual power include "red markers"?

73.665 Use of TV Subcarriers

Similar to FM stations, TV stations are authorized to use subcarriers for various applications including stereo, SCA and telemetry and control signals.

73.667(a) Definition of TV SCA

TV SCA does not include stereo or telemetry and control signals that are exclusively for the station use.

73.669(b) Installation of TV Subcarriers

TV stations may install aural subcarrier equipment without further authority from the FCC. The transmissions must conform to the specifications of 73.682(c), limiting the carrier frequency of all subcarriers to less than 120 KHz, limiting the subcarrier injection level, limiting the total injection of all subcarriers, and limiting the total modulation of the system. It is suggested that these specifications be checked after installing subcarrier equipment.

73.932 EBS Receiver and Generator Requirements

This section is included in this discussion to insure that those installing remote control points at "non-conventional" control points meet the EBS requirements.

Each station must have an operational EBS receiver that will alert the operator of an EBS test or alert any time the station is in operation. The operator must be alerted that an EBS signal is about to arrive (through the EBS attention signal unmuting the speaker), and allow the operator to hear the EBS message to evaluate what action needs to be taken. This action might include entry of a received test in the station log, initiating the broadcast of an emergency program, or shutting the station down. Probably the easiest way to meet this requirement is to have an EBS receiver at each control point.

All stations (except LPTV and NCE FMs running a TPO of 10 watts or less) must have an EBS attention signal generator. The operator must be able to put the EBS attention signal on the air at any time. This could be handled by either having the EBS generator at the program origination point and having all control be from that point, or putting the EBS generator at the transmitter site and operating it by remote control from various authorized control points. Although placing the EBS generator at the transmitter site is non-conventional, it may be required in many cases due to the EBS attention signal modulation requirements of 73.906(c) (each tone must modulate the transmitter a minimum of 40%). This may be difficult to meet if the EBS generator is followed by audio processing equipment. For FM stations running composite STLs, an EBS generator could drive the FM exciter in monaural, since there is no requirement that the signal be in stereo, and even if it were, the stereo signal would probably consist entirely of L+R and pilot. When remoting EBS equipment, the operating of the EBS generator must be sufficiently deliberate to insure that it will not be accidentally activated (meeting the switch guard requirements of 73.940(j)).

If the operator interfaces with the remote control through a Basic program on the DRC190, the operator could be asked to confirm the EBS transmit request. If the operator interfaces directly with the front panel of the remote control, a timing circuit at the transmitter site could require a Lower then Raise sequence before activation of the EBS generator, making a deliberate activation required.

To meet the EBS participation requirements (assuming a participating station), a station must, at a minimum, be able to rebroadcast the common emergency program received from the EBS receiver. This, and other program sources, can be placed on the air by an operator at a program origination point (the studio). An operator at another control point would have difficulty in selecting audio sources from that other point. It may be desirable to have an EBS receiver at the transmitter site that can be swithced on the air instead of the STL by remote control, if there are control points that are not the program origination point. This should meet the minimum EBS program broadcast requirements.

73.1215 Specifications for Indicating Instruments

Accuracy and resolution requirements for the meters (indicating instruments) at the transmitter site are outlined here. The old remote control rules (73.67(a)(6) for example) required remote meters to comply with the requirements for regular meters. The new rules (73.1410(a)) do not specify an accuracy or resolution for remote meters, but instead say it must be "sufficient" to insure compliance with the Rules. It might be a good idea to continue to meet the old accuracy and resolution requirements, the old 2% tracking requirement, and keep parameters inside the legal limits with an allowance for remote and local meter inaccuracies. For example, if the remote meter indicates you are operating at 104.97% of the licensed power, but the remote meter disagrees with the local meter by reading 1% low, and the local meter also reads 1% low, you are operating with greater than the authorized power!

73.1225 FCC Inspection

Note that 73.1225(c)(2)(iii) requires that measurement data taken on the installation of subcarrier equipment demonstrating compliance with the crosstalk and bandwidth limitations be available on request. Take those readings!

73.1230 License Posting

The station license (and other "instruments of authorization") must be posted at the place the station licensee considers the principal control point of the transmitter. At other control points, photocopies must be posted. Operator licenses (or photocopies if the operator works at more than one station and has the original posted at another station) must be posted at the control point where the operator is on duty.

73.1400 Remote Control Authorization

No authorization is required to operate a non-directional AM, an FM or a TV station by remote control. When a remote control point is established at a location other than the main studio and the transmitter (the locations of which are on the station authorization), the FCC (in Washington DC) must be notified of the new location within 3 days of that location first being used. This notification is not required if the FCC can reach "responsible station personnel" at the studio or transmitter during all times that the station is on the air. This generally seems unlikely, as there would then be no need to establish another control point!

A directional AM station must apply for remote control operation using either form 301 (new construction) or 301-A (existing array). An existing array to be operated by remote control must submit a report showing the stability of the array over the past year if the array does not include an "approved" sampling system.

73.1410 Remote Control Operation

These are the new relaxed remote control rules. In general, the new rules require the station to have sufficient telemetry and control to insure that it is operating within the terms of its authorization. Carrying over from the old rules, this generally meant that the operator could read the operating power and directional antenna performance for directional AMs, and video performance on TV stations (from the old 73.67(a)(3) and similar sections for FM and TV). In addition, the old TV rules specifically required an indication that tower lights were operating properly. Old (since deleted) section 73.67(a)(4) required the remote control to allow such adjustments at to allow the operator at the control point to perform all the functions required by the Commissions rules. This was generally interpreted to require enough control to turn the transmitter on and off, change modes (power/pattern) and adjust the output power to compensate for variations in line voltage or other variations. There have traditionally been several parameters that have been checked, but could not be adjusted by remote control (as the power could be adjusted). These included the directional array parameters, the modulation level (since the final audio processing is often done at the transmitter site), the transmitter frequency, the frequency and injection of all subcarriers and pilots, etc. The old television rule (73.676(a)(5)) might be considered a guideline for determining required controls. That is, a sufficient number of control circuits are to be available to perform all tramsmsitter adjustments normally required on a daily basis to insure strict compliance with the technical requirements of the rules. You should, of course, include a method of turning the transmitter on and off, even if this is not done on a daily basis. By reducing the number of controls, you simplify the remote control installation, but possibly have to make more trips to the transmitter site or have to shut down because a parameter is out of authorized limits.

Under the new rules, the decision as to what to meter and control is difficult. The new rules might be interpreted to indicate that if your system is quite stable, readings could be checked "as often as necessary", which might be during a weekly transmitter site inspection. This would require no metering at the control point at all! As long as you never operate with parameters out of tolerance, you'd appear to be legal, but I'd hate to try to get that past an inspector!

Although section 73.1550 covers extension meters, it could be considered to be a guideline as to what to meter. Extension meters are a form of remote metering where the distance between the transmitter and the operator position is less than 100 feet and on the same floor, or one floor above or below the operator position. All metering must be on a full time basis, rather than the standard remote control method of analog multiplexing samples. No control circuits are required. It is assumed the operator will run to the transmitter when something needs to be adjusted. In any case, 73.1550(b) does list parameters that the operator should be able to see. These include meters to determine the DC input power of the last RF stage (plate voltage and current), the antenna current (non-directional AM), the common point current (directional AM), or the transmission line meter (FM). In addition, directional stations must extend the antenna monitor indications. FM stations with a TPO of 10 watts or less need only an indication that the transmitter is on or off. TV stations need the same meters as an FM for the visual and aural transmitters plus other visual monitoring equipment (waveform monitor, vectorscope, etc.). This section might be considered a guideline as to what to meter. It pretty much duplicates the old remote control rules that were deleted.

Eventually, something is going to fail or drift, placing the station operation out of limits. For example, it is unlikely that a capacitor in a directional array, or a tower light, will fail during a weekly or monthly inspection. If it fails at any other time, you will operate outside the terms of the license. Even with full remote metering, there will be a period of time when the station is outside licensed parameters. This would be the time between the failure and the next set of meter readings.

To assure compliance with the rules, and to make the "best effort", it is suggested that readings of all parameters that can result in operation outside the licensed paramters be taken as often as possible to minimize the amount of time between the failure of the system and detection of the failure. Under manual operation, this used to require readings be taken every 30 minutes, then every three hours. The reading frequency has now been relaxed to "as often as necessary". I'd suggest that stations using manual remote control continue to take readings every three hours and at mode changes. The readings should, at a minimum, be compared with limits, and a notation that the readings were found to be within limits logged. Better yet would be a log entry of the readings showing long term variations in system parameters.

With automatic logging using the DRC190, readings can be taken continuously, checked against limits, logged at programmable intervals or on an alarm condition or change of a specified amount. This insures that operation outside licensed parameters will be dectected immediately (depending on reading update time). In addition, statistics of the minimum, maximum and average of each parameter can be stored, giving an indication that the operation was inside the authorized limits during the covered time, and allowing adjustement of the system (especially a directional array) based on the average reading.

As previously, the remote control system must be designed and installed so that it can be activated or controlled only by licensed operators authorized by the licensee.

The remote control and monitoring equipment must be calibrated as often as necessary. No accuracy is specified, although the 2% tracking requirement of the old rules seems reasonable. It's suggested that the system be operated inside a "window" tighter than the legal maximum to allow for inaccuracies of the monitoring equipment and remote metering.

A major change in the new rules covers the "fail safe" requirement. Previously, a malfunction in the control circuits (line faults, etc.) required the system to immediately remove the transmitter from the air. The new rule (73.1410(d)) requires the design to be such that a loss of control does not falsely activate the transmitter or change modes. It is still suggested that a "fail safe" meeting the old requirements be installed. This could be the fail safe circuit of the remote control, or the squelch relay of an STL, providing a continuous on/off control of the transmitter.

A loss of telemetry requires the transmitter be shut down within three hours of detection. The idea here is that hopefully your transmitter will not drift out of tolerance in the three hours allowed to get someone to the transmitter site to take control.

AM stations can amplitude or phase modulate the carrier to return telemetry. The deleted 73.67 authorized amplitude modulation of the AM carrier with a subaudible metering tone of 30 Hz or less with a maximum injection of 6 percent. The new remote control rules do not give such specifications, however, 73.127 allows the use of AM multiplex transmission for broadcast and non-broadcast purposes (between those two, sounds like anything is ok!). The installation of the multiplex equipment must conform to the requirements of 73.1690(e)(6), which requires the audio specifications of 73.40 and 73.44 be met and that the frequency stability requirements of 73.1545(a) be maintained (plus or minus 20 Hz). The only limitation on AM subcarrier immediately apparent in 73.40 would be the limitation on noise (45 dB down between 30 Hz and 20 KHz). If the subcarrier is below 30 Hz, there appears to be no limit on its injection, other than the overall modulation limits. In addition, it would appear that a subcarrier between 30 Hz and 20 KHz could be used if it was below the -45 dB level and the resulting sidebands were at least 25 dB below carrier level if they are more than 15 and less than 30 KHz from the carrier (limitations of 73.44(a)(1). Since the DRC190 does not use subaudible metering and the use of "above audio" subcarriers is untested on AM, that's about all we'll say at this point.

TV or FM stations are permitted under the new rules to use aural subcarriers to return metering. It is important that the performance of the system be tested after the installation of subcarrier equipment to insure that it still meets FCC requirements, especially with regard to crosstalk.

TV stations can additionally use the vertical blanking interval to return metering.

Although not specifically mentioned in the FCC Report and Order regarding remote control, frequencies are available under part 74 of the rules for control and metering signals.

73.1560 Operating Power Limits

The operating power limits for AM, FM and TV stations are listed here. This is one of the readings an operator needs to check for compliance.

73.1570(b)(2) FM Modulation Limits

This is a relatively new section authorizing an increase over 100% modulatin when subcarriers are used on an FM station. This excess modulation is not authorized for stations operating within 320 kilometers of Mexico. Consult this section when using subcarriers.

73.1570(b)(3)(i) TV Modulation Limits

As above, television stations running subcarriers may increase the deviation of the aural transmitter over that normally authorized for monaural sound broadcasts. The maximum injection of the various subcarriers and the maximum overall deviation of the aural carrier is given in 73.682(c).

73.1580 Required Transmitter System Inspections

The entire system must be inspected as often as necessary to insure compliance with the rules and the terms of the license. Since there are some parameters that are often not remoted (carrier frequencies, subcarrier frequencies, subcarrier injections, overall modulation, STL frequency, STL modulation, STL power, TRL frequency, TRL modulation, TRL power, directional array base currents, etc.) a scheduled routine inspection is suggested.

73.1690(d)(2) Notification of FCC on Commencing Remote Operation

The FCC must be notified of a commencement of remote control operation or the addition of control points. Directional AM stations must go through an application procedure prior to starting remote control operation.

73.1690(e)(5) Notification on Installation or Replacement of Subcarrier

The FCC must be notified on the installation or replacement of subcarrier equipment. Note that equipment tests must be run demonstrating the performance of the system still meets FCC requirements after the installation of new subcarrier equipment (73.319(h) or 73.669(b)).

73.1800 General Station Log Requirements

This section outlines changes from the old "operating log" requirements. There are several changes from the traditional logging requirements.

Under the old operating log requirements (and the old program logs, for that matter), an operator was required to sign the log when going on duty, and then sign the log again on going off duty. From the log, it was readily apparent who the licensed operator in charge of the transmitter at a particular time was. The current station log requirements require only that an operator sign the log when making an entry, verifying that the entry was indeed accurate. Apparently it is solely the station licensee's responsibility to insure that a licensed employee is in charge of the transmitter at all times it is on the air. The FCC is not requiring an "audit trail" that shows who the particular operator at any time was. It might not be a bad idea, though.

The logs must be orderly and legible. Abbreviations may be used if they are explained elsewhere on the log. Each page is to be numbered, and time entries must be local (not UCT) and must be identified as advanced or non-advanced.

The method of making a correction to a log remains the same as previously required. The error is striken out and a corrective explanation is made on the log or attached to the log. The explanation must be made by the person who made the log entry, the station chief operator, the station manager, or an officer of the station licensee. Logs are not to be erased or obliterated. Automatically kept logs cannot be altered after the entries have been recorded.

73.1820 Station Log Data

Log entries may be made manually or by an automatic device meeting certain requirements.

Log entries of required parameters must be logged prior to any adjustment of the equipment (perhaps showing out of tolerance operation). If an adjustment is made to restore a parameter, the corrected indications need to be logged along with a notation of the corrective action taken. Indications that are affected by modulation must be taken without modulation. The question is: affected how much? See the discussion of 73.51 above. Finally, the actual time of observation must accompany each log entry.

What to log? All stations are required to log data relating to the improper operation of tower lights, if applicable. Note that section 17.47 requires an observation of the tower lights at least once every 24 hours, or the observation of a properly maintained indicator designed to register the any failure in the lights. As an alternative, an automatic alarm system to detect any failure and indicate such failure to the licensee (or the operator) may be used. Finally, all control devices, indicators and alarm systems must be inspected at least every three months to insure they are operating properly.

A suggested tower light indicator for driving a remote control (such as the DRC190) consists of a current transformer sampling the current driving the lamps on a particular tower. This transformer drives a rectifier and a very low frequency low pass filter. The output of the low pass filter is a DC voltage that is determined by the current of the lamps that are on continuously and the duty cycle and current of the lamps that flash. A failure of any lamp will cause a decrease in the output of the filter. A failure of the flasher will cause a decrease or an increase in the output of the filter, depending upon whether the flasher failed in the on or off position. Such a circuit would detect every possible failure with the exception of the flash frequency going outside the allowed 12 to 40 flashes per minute. It is suggested that only one beacon and set of steady burning lamps drive each such system, as with several driving the same one it may not be possible to detect the failure of a single lamp. This concept has not been tested, but is believed to give a more accurate indication than the typical current transformer driving either an analog or digital remote control. The system may, in fact, compensate for variations in line voltage on its own, since a reduction in line voltage would cause a decrease in lamp power, decreasing the resistance of the lamp, partially compensating for the reduction in current expected due to the decrease in line voltage (ie, incandescent lamps tend to be constant current devices).

The only other item that needs to typically be logged involves EBS operations (receipt and transmission of tests, etc.).

Those AM stations without FCC-approved sampling systems must log the common point current and directional array parameters at least every three hours. The directional array parameters to be logged include phase or phase deviation from licensed value, sample current (loop current) or loop current ratio or deviation of loop current ratio from the licensed ratio. When the DRC190 is programmed to provide automatic logging, it typically logs all these parameters, calculating ratios and deviations from the actual antenna monitor samples. The logging of deviations vastly simplifies the operator responsibility in checking the operation of the station. Daily summaries of minimum, maximum, and averages for all logged parameters give a quick demonstration that the station operated within the terms of the license. The average directional array parameters are an aid in adjusting the directional array so that it "swings" around the licensed parameters.

Finally, TV stations, FM stations, non-directional AM stations and directional AM stations with approved sampling systems are not required to log any parameters. Summarizing, only those whose parameter indications are suspect (due to an out-of-date sampling system) are required to log the indications.

Section 73.1820(b) outlines the requirements for automatic logging of station parameters. Since parameter logging is required only for non-approved directional arrays, it appears that these requirements will not apply to most stations that are using automatic logging and alarm functions to exceed the FCC monitoring requirments, and to aid in the adjustment and maintenance of the station. In any case, it is a good idea to meet these requirements, which are outlined below.

The automatic logging equipment must make accurate entries. It should be calibrated as often as necessary to insure it does so.

The system includes an automatic aural alarm that is actuated if any monitored parameter goes outside the licensed limits. This alarm should be audible to the operator at the operating position.

The alarm circuit must operate continuously, or the logging system must record each parameter at least once each 30 minutes. Since almost all logging and alarm systems use a single analog to digital converter that is scanned among the sample inputs (in fact, all type accepted antenna monitors will give an indication of only one tower's parameters at a time), this is a difficult requirement to meet unless log entries are indeed made every thirty minutes. It is not possible to continuously watch the phase of towers 2 and 3 of a directional array with existing equipment! You may wish to decide that a sample for alarm checking once every minute or so is "continuous", or make the log entry every thirty minutes (you have stock in a paper mill?), or decide that this parameter logging isn't required anyway and is for station use only.

The logging requirements have changed substantially over the past 15 years. It appears that the FCC has specified a minimum log requirement of almost nothing (tower lights and EBS), but routine parameter logging to show that the operation of the station is routinely well within FCC requirements would be helpful. A program running on the DRC190 can help fulfill these supplemental logging requirements.

73.1860 Transmitter Operators

Operators are still required. Even with ATS! This operator must be licensed (any commercial class, unless otherwise endorsed. . . FCC is looking at endorsing General License to prohibit broadcast operation to decrease the number of people taking the test). The operator must be on duty at an authorized remote control point, ATS control point, or the transmitter. The operator must be able to observe the required transmitter and monitor metering to determine deviations from normal indications. Here again, the rules are non-specific as to what parameters need to be observable. Tradition has been that the operator be able to observe the power and any directional array or video parameters. The operator used to have to monitor carrier frequency (used to log that every 30 minutes!) and modulation, but the requirements for monitors have disappeared. Presently, the station is required to stay on frequency and not over modulate. It is up to the station licensee to determine how this is complied with, and how often frequency and modulation are checked. Can power and DA parameters be far behind?

It is interesting that 73.1860(a) requires the transmitter operator to be licensed, but makes no mention of the operator being an employee of the station licensee, while 73.1800(a) requires the station log to be kept by an employee (though no mention of a license is made here) having actual knowledge of the facts required. Generally, the transmitter operator ends up keeping the log, requiring the operator to be licensed and be a station employee. Perhaps other arrangements are possible?

Finally, the transmitter operators must know what they are doing! They should have printed operating instructions and limit charts. The operating instructions should indicate what to do when a reading is out of tolerance (adjust it or shut the station down), how to handle EBS tests and alerts and how to turn the transmitter on and off. The limit chart needs to be detailed enough so that each operator can indeed insure that the operation is within limits. A non-directional AM or an FM determining power by the direct method has it easy: Just list the minimum and maximum antenna current or transmission line power. An FM determining power by indirect method should have a chart giving the minimum and maximum authorized plate current for a variety of expected plate voltages. A directional AM should have limits on the common point current. It should also have a table of minimum and maximum loop currents for each tower based on the various expected values of the reference tower loop current (unless the antenna monitor indicates ratio or deviation directly). Phase limits also need to be posted. When running a Basic program on the DRC190 to do logging and limit checking, the various parameters and calculated parameters can be checked against limits, adjustments made, and alarms sounded.

73.1870(c)(3) <u>Chief Operator to Review Station Logs</u>

The Chief Operator of the station (or his/her designee) must do a weekly review of the station logs to insure that they have been completed properly and to insure that the station has been operating within the terms of the license. On completion of the review, the log is to be signed and dated, any required corrective action taken, and the station licensee advised of any repetitive condition.

73.3544(b)(5) <u>Change in Remote Control Point Location</u>

This section requires an informal application to cover the change in a remote control point when prior authority to operate by remote control is not required. This conflicts with 73.1400(b) and 73.1400(c), which indicate that no notification is required on establishing or changing a remote control point if that control point is at the main studio location (or a "remote" control point at the transmitter). 73.1400(c) indicates that establishing a remote control point at another location (other than main studio or transmitter) requires notification of the FCC within 3 days. Note that directional stations must file a form 301 or 301-A to establish a remote control point. Once the directional station is operating by remote control, additional control points or a change in control point could possibly be implimented using the notification procedure of 73.1400(c).

73.4097 <u>Use of EBS Attention Signal on Automated Stations</u>

This is an FCC policy that is not published in the CFR. As I recall, it disallows the automatic operation of the EBS generator. Some stations were using the program automation system to send an EBS test with the automation system starting the EBS generator. The FCC did not feel this was appropriate. In addition, they clarified the requirement that the EBS attention signal must be from the EBS generator. Tones on a tape cartridge are not acceptable.

74.24 Short Term Operation of Auxiliary Stations

Broadcast stations (licensed under Part 73) are authorized to use Part 74 frequencies on a short term basis without further authority from the FCC. There are various restrictions on the "short term" use of these frequencies. These include:

The other requirements of Part 74 apply.
Operation is on a secondary non-interfering basis.
Operation is limtied to 720 hours annually per frequency.
The antenna must not increase the overall height of an exiting structure by more than 20 feet.
The station is to be identified with the call letters of the broadcast station.
Use of the frequency is to be coordinated with the local frequency coordinating committee.
Various other limitations apply.

This information is included to show a possible way of getting a Part 74 Group P telemetry return link operating quickly.

74.402(a)(7) Frequencies Reserved for Operational Communications

The following frequencies are to be used exclusively for "operational communiations", which includes tones for signalling, remote control or ATS control control and telemetry, or communications concerning the technical or programming operation of a broadcast station and its auxiliaries (such as control and telemetery data of the DRC190, and the intercom voice traffic of the DRC190): 450.01, 450.02, 450.98, 450.99, 455.01, 455.02, 455.98 and 455.99 MHz.

74.434 Remote Control of Auxiliary Stations

Various limitations are put on the remote control operation of these stations. These probably apply to TRL systems on Group P frequencies. The operator needs to be able to turn the transmitter on and off. This would generally require the addition of a latching relay driven by a raise/lower command of the DRC190. The contacts of this relay would be put in series with the keying line of the DRC190.

The modulation needs to be automatically limited to the maximum allowable, or a modulation indicator needs to be at the control point. Most radios used in TRL service include a modulation limiter.

The transmitter needs to be inaccessible to unauthorized persons.

74.462(b) TRL Transmitters Limited to 1.5 KHz Deviation

The service manuals on many radios used for TRL service indicate to adjuste the transmit deviation to 5 KHz. When used on Group P frequencies, this must be reduced to 1.5 KHz. This should be pointed out to any "two way" radio technician who may adjust the equipment.

74.464 TRL Frequency Tolerance

All TRL transmitters are probably base stations, which requires a frequency tolerance of .00025 percent.

74.467 Posting of TRL Licenses

The TRL license is to be posted with the broadcast station license (see 73.1230). If the TRL transmitter is not at that location, there should be a label showing the TRL call sign, the call sign of the broadcast station, the name and address of the licensee, the frequency, and unit designator (if used). This label is actually required at the operating position of the transmitter. If the transmitter is considered to be operated by remote control from the main control point, the license posting at that point may be sufficient, but it never hurts to have identification at the transmitter itself.

74.482(b) Identification of TRL Transmitters

Since TRL transmitters are often doing a continuous transmission, or a continuous series of transmissions, this section requires the transmitter to be identified at approximately one-hour intervals. This can be programmed into the DRC190.

74.482(d) International Morse Code Identification

TRLs can be identified by an automati international Morse code identifier. The DRC190 is equipped with such an identifier as an option. This section requires the ID tone to be 750 Hz plus or minus 10 Hz and the code speed to be 20 to 25 words per minutes. These are set in the DRC190 based on the processor clock, and are not adjustable. The modulation level of the ID is to be 40% plus or minus 10%. This is adjustable on the DRC190 processor board.

74.531(d) Subcarriers on Aural STLs for Control

The use of subcarriers on Aural STLs for control and other operational communications (and other purposes) is authorized by this section.

74.533(a) Remote Control or Unattended Operation of Aural STLs

STL transmitters may be operated by remote control or unattended. If they are operated by remote control, the operator must be able to turn the carrier on and off and must have either an indication from the transmitter that the carrier is on or off, or an indication at the control point that the control circuit should have placed the STL transmitter in a radiating or non-radiating state.

STL transmitters may be operated unattended if an operator verifies at the receiving end of the STL or by listening to the output of the broadcast station using the STL that the STL is operating properly.

74.535(b) STL Subcarrier Injection Levels

This section limits the injection level of subcarriers (in the 947 to 952 MHz band) by making an approximation of the significant sideband bandwidth using the formula 2M+2D. 2M+2D is limited to 500 KHz where M is the highest modulating frequency and D is the deviation of the STL carrier by that subcarrier (injection level).

74.634 Remote Control of TV STLs

Television STLs and inter-city relays are authorized to operate by remote control or unattended with requirements similar to aural STLs. Unattended TV STL operation is covered in 74.635.

74.734(a) Operator Required on LPTV During Local Origination

This section requires an operator at a control point of an LPTV during local origination. The exact control point requirements are not spelled out, but appear, at a minimum, to be that the operator have the ability to turn the transmitter on and off.

74.933 Remote Control of ITFS Stations

An ITFS must have an operator meeting the requirements of the non-existent section 74.966 at the control point of an ITFS station. The operator must have on/off control of the transmitter and must have a carrier operated device which gives a continuous visual indication whenever the antenna is radiating. The on/off control can be handled by the raise/lower controls of a remote control system (such as the DRC190). The "continuous visual indication" could be either an off air receiver or a dedicated metering or status signal returning from the transmitter site. Since the indication is required to be continuous, it appears that a status indication is most desirable, since metering is generally not continuous as other readings are checked.

ITFS remote control still has the old remote control fail safe requirement. That is, the transmitter must shut down immediately on loss of continuous control from the control point.

NOTE:

This discussion of the FCC requirements regarding remote control is just that: a discussion, not a legal opinion. You should always refer directly to the FCC rules and regulations for the exact requirements, and use the advice of an attorney or consultant in resolving the ambiguities of the rules.

The rules are changing. This discussion is based on the October 1, 1984 issue of the CFR and the FCC Report & Order of 21 November 1984. There is talk of further reduction in remote control requirements and the authorization of ATS for directional AM stations and television stations.

Installation

The DRC190 includes copyrighted software in its EPROM. Prior to starting installation of the system, please complete and sign the system registration form as instructed in the FCC and Copyright Notices section of this manual.

When unpacking the DRC190 remote control system, first remove the cover of each unit. Insure that all circuit boards are firmly in place and that all connectors are firmly in place. If the system was shipped with the battery backup option, one of the battery leads was disconnected for shipping. This battery lead should be connected when the system is ready for installation.

WARNING

Once the battery or line cord is connected, the power supply portion of the DRC190 has high voltage present. Be sure to observe safety precautions!

Once you are satisfied that all the pieces are connected together properly, you can apply AC power by plugging the supplied AC cord into the rear panel and a 117 VAC outlet. Hold down the "0" key on the membrane keyboard for a couple of seconds. A "site-channel" message should appear on the front panel LCD. This initial power-up procedure should be repeated for each unit provided as part of the system.

Once each unit has been powered up, try connecting a communications link between the units. Provided with each unit is a 25 pin D connector with a twisted pair of wires connected to it. This plug with cable simulates a telephone line for preliminary tests. These plugs should be connected to J21 on each unit. The twisted pairs from each unit should be connected together, allowing each unit to transmit and receive data to and from the other units.

For testing the system, H&F programs the site number in each unit. The unit with the lowest serial number is programmed as site 0. The next lowest site number is programmed as site 1. This process continues until all units are covered. Note that the site number of each unit can be changed by the user.

With all units connected together, try keying in a site and channel number into each unit. These are keyed in as a four digit number. For example, keying in 0123 results in selection of site 01, channel 23. The DRC checks to see if the requested site is the same as the site number where the entry is being made. If so, the appropriate channel of the A/D converter is selected and the reading displayed. If the requested channel does not exist in the selected site number (for example, channel 23 in a unit with only one A/D board set up to be channels 0 through 9), a reading of zero is returned. If the selected site is not the same as the site the request is being requested from, the DRC sends a request out to the appropriate site on the communications link. Once that site responds, the reading is displayed.

Note that with no samples connected, you'll get random readings displayed. You should be able to step through the channels using the up arrow and down arrow keys. Once a site and channel are selected, a reading should show up within a couple of seconds, assuming the selected site is in the system.

Another preliminary test to complete is the testing of the intercom. On each unit try pressing the COM key. Within a second or so the display on that unit should instruct you to start talking. Talking into the unit with the COM key pressed should cause your voice to be heard at each of the other units. To prevent the failsafe system from dropping out, intercom transmissions are limited to 30 seconds each.

If you have a standard CRT terminal, it can be plugged into J22 of a DRC unit. The DRC is shipped with the terminal baud rate set to 9600 Baud. Other CRT data characteristics include:

1 start bit, always space
7 data bits
1 parity bit, always mark (marking parity)
1 stop bit, always mark

On pressing the reset button on the rear panel of a DRC unit, a sign on message should appear on the CRT. Typing the sample one line command listed below should cause a meter reading to be displayed, substituting the appropriate site number for S.

```
DISPLAY METER$(S,0)
```

Once these functions on the DRC have been tested, it is time to start the actual installation. It is suggested that each DRC190 be rack mounted so that the display is at eye level, or slightly above. The display contrast is optimum when the display is viewed from slightly below horizontal. The following tables indicate the connections to be made to the rear panel connectors.

The metering inputs are isolated from ground. The metering sample can be up to 100 volts above or below ground (100 volts maximum common mode voltage). The sample voltage itself must be less than 2 volts maximum. If the sample voltages to be read are higher than the allowed 2 volts, external voltage dividers can be added to each sample, or on systems with revision B or higher analog to digital converter boards, a voltage divider can be added to the A/D converter board. This voltage divider will be applied to all samples presented to that board. Refer to the A/D board component placement drawing for placement of the series (R20) and shunt (R21) resistors in the voltage divider. To allow for a 20 volt sample, the series resistor might be 9 K while the shunt resistor is 1 K. While the actual value of the resistors is non-critical, the stability of the resistors is quite critical. It is suggested that low tempco resistors be used (such as the 5 ppm/C degree resistors used in the reference voltage divider). Prior to connecting samples, each sample voltage should be measured with the highest expected sample to insure that the sample voltage and the 100 volt common mode voltages are not exceeded.

The control outputs are open collector outputs. When the output is activated, the output is pulled to ground. When the control is released, the maximum voltage applied to the control output should be 30 volts or less. When the output is active, the maximum current should not exceed 500 mA. Again, each control line should be tested prior to connecting it to the DRC.

Once it has been determined that the sample and control lines are within requirements, they can be connected to the DRC. Provided with the DRC are the required connectors, connector pins, connector shells, and a pin insertion/extraction tool. The following table lists the supplied parts and the suggested crimp tool. The use of shielded cables for all connections to the DRC is suggested to insure immunity from high RF fields. The DRC has been tested in 1 MHz fields exceeding 5 volts/meter and high VHF RF fields (the test site includes 4 FM stations and a TV station) without shielded cables with no problem, but shielded cables are good insurance. The shield (or drain wire) of the cable should be connected to pin 1 of the connector.

We have not had any problems with RF at FM stations. On AM stations with extremely high RF fields (ie, open panel phasor a couple of feet away and one tower right outside the building), it may be necessary to add external low pass filters on some lines. Metering and control lines that go to monitoring equipment in the same equipment rack (such as an antenna monitor) have not required additional filtering. Control, metering, and communications lines leaving the equipment rack that the DRC is mounted in may require external filtering. If there is a problem with unstable readings when the station's RF is brought up, or the communications link appears to fail when the RF is brought up, simple LC filters can be put in the offending lines. A series 10 uH choke with a 0.1 uF capacitor to ground (on the DRC side of the choke) has been successful in curing RF problems. The filter assembly can be placed anywhere in the same equipment rack as the DRC. Shielded cables should be used between the DRC and the filter assembly. If resistors are substituted for the chokes, and a shunt resistor is added, an external RF filter can also provide voltage division to bring the sample voltage within range of the A/D.

Once all sample, control, and communications lines have been connected, the system is ready for calibration. Refer to the Setup and Calibration Section.

J21 Connections

J21 connects to the communications system linking the DRC190 units. This link is to be a half duplex (two way, but only one way at a time) voice grade link meeting the requirements of a Bell 3002 circuit with basic conditioning as described in Bell System publication 41004. Telephone lines, subcarrier links and radio links are normally available to meet these requirements. Should a link capable of handling the 1200 Baud data not be available, the DRC data rate can be reduced as required. See the Setup and Calibration Section for information on changing the communications data rate.

Pin	Description
1	Shield
3	Audio input data +
16	Audio input data -
4	Audio output data +
17	Audio output data -
6	Transmit key*
19	Ground (Transmit key return)
7	External Speaker
20	Speaker return
12	Direct Connect Modem Tip
25	Direct Connect Modem Ring
13	Direct Connect Modem Shield

The above list is all that is normally required for connection of the DRC system. If working with a 2 wire telephone line, connect pins 3 and 4 to one side of the line and pins 16 and 17 to the other side of the line. If a 4 wire telephone line or other 4 wire circuit (microwave subcarrier, RF link, etc.) is used, pins 3 and 16 should be connected to the receiver and pins 4 and 17 to the transmitter.

If a subcarrier transceiver inside the DRC190 is used, the lines driven by it should not be connected outside the DRC. For example, if this site is receiving a 110 KHz control subcarrier, the internal subcarrier transceiver puts the received audio on pins 3 and 16. No outside connection should be made to these pins.

The Transmit key* (* indicates the line is active low) line is pulled low whenever this site is transmitting data. This line can be used to key a "TRL" transmitter. Note that the transmit key* output can only sink 40 mA with an open circuit voltage of 15 volts.

The external speaker connections are available on revision B processor boards, and later revisions. An external speaker with a series volume control can be connected to these pins. This speaker is driven by the speaker driver amplifier on the processor board. The keyboard clicks and intercom voice will be heard through this speaker. This can be useful at noisey transmitter sites. The front panel speaker remains the microphone for intercom operation. The impedance presented to these pins should not be below 8 ohms. An 8 ohm speaker with a series pot as a volume control is suggested.

The direct connect modem connections are used only if a direct connect modem was ordered. These pins should be connected to the dial-up telephone line so that the Basic applications program can receive and place data calls.

If the DRC was ordered with the "Dual Audio Option", there is an additional audio input and output on each unit. This allows the DRC to be used at an intermediate microwave site in a large microwave system. A unit with the dual audio option has the following connections on J21.

Pin	Description
1	Shield
2	Audio input number 1 +
15	Audio input number 1 -
3	Audio input number 2 +
16	Audio input number 2 -
4	Audio output number 1 +
17	Audio output number 1 -
5	Audio output number 2 +
18	Audio output number 2 -
6	Transmit key*
19	Ground
7	External Speaker
20	Speaker return
12	Direct Connect Modem Tip
25	Direct Connect Modem Ring
13	Direct Connect Modem Shield

Subcarrier Transceiver Connections

The subcarrier transceiver input and output connections appear above the reset button on the rear panel. The left BNC connector is the subcarrier output, the right BNC connector is the subcarrier input.

If a subcarrier is being used as a control uplink, the subcarrier output at the studio unit should be connected to the multiplex input on the STL transmitter, and the subcarrier input on the transmitter site unit should be connected to the multiplex output of the STL receiver.

If a subcarrier is being used as a metering downlink, the subcarrier output at the transmitter site should be connected to the exciter SCA input (connection for external SCA generator), and the subcarrier input at the studio should be connected to a composite output of the modulation monitor or wideband receiver.

Various portions of the subcarrier system can be replaced with external subcarrier equipment, if desired. For example, some FM exciters have a built in SCA generator. The audio output from the DRC190 could be connected directly to the audio input of this SCA generator. In addition, the modulation monitor/subcarrier demodulator combination at the studio could be replaced with a standard SCA (background music) receiver.

Since STL and exciter specifications vary, some adjustment of the subcarrier transceiver boards will probably be required. The adjustments require the use of an oscilloscope, a digital voltmeter and a frequency counter.

The only control that should require adjustment will be the subcarrier output level (R4). With an oscilloscope connected to the output of the subcarrier generator, adjust R4 for the level required by the STL or exciter multiplex input. STL equipment supplied by Moseley Associates requires a subcarrier level of 1.5 volts P-P. STL equipment supplied by Micro Controls requires a subcarrier level of 1 volt RMS.

Other subcarrier transmitter controls include:

Course Frequency - R6: Adjust this control to the approximate subcarrier frequency desired.

Fine Frequency - R7: Adjust this control to the exact subcarrier frequency desired.

Deviation - R9: Adjust this control to the desired deviation. Deviation is normally set to 1 KHz/Volt. To make this adjustment, ground the audio input and count the output frequency. Apply 1 volt DC to the input and adjust R1 so the frequency is 1 KHz lower than the previously measured value. With the subcarrier deviation control adjusted in this manner, the standard 0 dBm output level of the modem in the DRC190 will result in +/- 1 KHz peak deviation of the subcarrier. Note that once the deviation is set, adjustment of the subcarrier frequency will result in the same percentage deviation. Multiplying the subcarrier frequency by two results in double the deviation. Since the subcarrier demodulator is designed for a constant frequency deviation (rather than a constant percentage deviation), the deviation control will have to be readjusted after a frequency change to maintain the desired 1 KHz/volt deviation.

Symmetry - R1: With an oscilloscope conntected to the subcarrier output, adjust R1 for a symmetrical waveform.

Distortion - R2: Connect an audio distortion analyzer to to the subcarrier output. Adjust R1 and R2 for minimum distortion (minimum subcarrier harmonic content). It should be possible to get distortion below 1% (sum of harmonics 40 dB below subcarrier level).

Subcarrier Receiver Controls include:

Local Oscillator Null - R11: With no input to the subcarrier receiver, adjust R11 for a null in the AC level on pin 2 of the XR2206.

Local Oscillator Frequency - R17: A jumper is supplied on P05. This jumper improves the local oscillator null. When adjusting the local oscillator frequency, remove this jumper. When the local oscillator frequency has been adjusted, return the jumper. With no input to the subcarrier receiver, connect a frequency counter to pin 1 of P05 and adjust R17 for the desired local oscillator frequency. The desired local oscillator frequency is 455 KHz - SCA frequency. Typical local oscillator frequencies are listed below:

Subcarrier Frequency (KHz)	Local Oscillator Frequency (KHz)
26	429
67	388
92	363
110	345

On installing the subcarrier equipment, it would be a good idea to try sending the transmitter directly into the appropriate receiver after the required adjustments have been made (typically only the transmitter output level). Feed a tone into the subcarrier transmitter. If a CRT terminal is available, this can be easily accomplished by typing MODEMTST. A clean sine wave should be apparent at the subcarrier receiver. A distorted sine wave is normally due to excessive deviation of the subcarrier. A noisey signal is normally due to excessively low subcarrier level. The minimum acceptable subcarrier level at the receiver input is approximately 150 mV P-P.

Once the direct connection has been shown to work, connect the subcarrier equipment to the normal RF link and check end to end operation. Again, a distorted signal is probably due to over deviation, although this should not have changed since the direct connection test. A noisy signal may be due to crosstalk from the main channel. To simplify manufacturing and adjustment, no input filtering is included in the subcarrier receiver. Instead, the whole base band is hetrodyned up to 455 KHz (and frequency inverted due to the low side local oscillator) and filtered in a standard ceramic filter. The ceramic filter gives excellent rejection characteristics, but it is possible to get interference from the main channel due to actual main channel harmonics (due to non-linearity of the STL or exciter), or various cross products due to the non-linearity in the mixer. Interference from the main channel can be checked by dropping the main channel and seeing if the subcarrier signal cleans up. Interference from the main channel can sometimes be fixed by adjusting the main channel and subcarrier levels (subcarrier injection).

As an aid in checking the system, the front panel speaker of the DRC190 is enabled when the calibrate mode is first entered and remains active until another key is pressed, or a speaker off command is received (such as at the end of a CW ID or the end of an intercom message). To enter the calibrate mode, key in 1 2 3 4 followed by a decimal point. Leaving the speaker enabled allows one to hear the received data, test tones, etc. as an aid in checking subcarrier operation.

Once the subcarrier equipment is operating properly, the remainder of the

DRC190 system should be tested with the subcarrier equipment.

Radio Transceiver Installation

Hallikainen & Friends is supplying a Standard Communications FX60U transceiver when a separate radio control/metering link is required. This transciever has been modified slightly to permit operation with the DRC190 on part 74 group P frequencies. These modifications include: Change of second IF filter to allow 5 KHz channel spacing, Addition of an attenuator on microphone input to allow the DRC190 to drive the unit with a 0 dBm signal, Adjustment of the FX60U deviation control to yield a peak deviation of 1.5 KHz when the input is driven with 2.2 KHz at 0.776 volts RMS.

Install the FX60U close to the DRC190 (typically immediately above the DRC190 in the equipment rack). Plug the supplied cable into the FX-60U and J21 of the DRC190. Connect the antenna to the FX-60U. Connect the AC power to the FX60U.

Turn on the power switch on the FX-60U. Press CH1, RADIO , and L/R while releasing CH2 and IC. Set the volume control to "9:00" and adjust the squelch control until the speaker just mutes. If desired, an RCA plug can be plugged into the rear panel speaker jack to mute the internal speaker.

If the CW ID has not been set up, see the calibration section of this manual.

If other sites have been installed, you should be able to select that site on the DRC190, see the TX light on the FX-60U flash, see the busy light flash, then have the second line of the DRC190 display update with the telemetry information. If the rear panel external speaker jack has not been plugged (disabling the internal speaker), you should also be able to hear the received data.

The radio link should now be operating.

For reference, the below information on modifications to the FX-60U is provided.

The rear panel barrier strip on the FX-60U has the following pin out:

1 - Intercom low
2 - Intercom high
3 - Intercom low
4 - Intercom high
5 - Receiver audio out low
6 - Receiver audio out high
7 - Transmit key low
8 - Transmit key high (pull to ground to transmit)
9 - Transmit audio in low
10- Transmit audio in high

The following modifications are made to the FX-60U.

Existing wires are disconnected from the barrier strip terminals 5 through 10.

A twisted pair of wires is connected from the rear panel speaker jack to terminals 5 and 6.

Terminal 7 is grounded. Terminal 8 is wired to JP11 on the PLL board. This is the transmit key line.

A 600 ohm resistor is wired across terminals 9 and 10. A resistor is connected to terminal 10. A twisted pair is wired from terminal 9 and the free end of the resistor to JP03 (from resistor) and JP04 (from terminal 9) on the PLL board. The resistor value (50K to 750K) is selected to yield 1.5 KHz deviation when 2.2 KHz at 0 dBm is applied to terminals 9 and 10.

Direct Connect Modem Connectios

If a direct connect modem was ordered with the system, the telephone line tip and ring connections appear on J21 pins 12 and 25, as indicated earlier in this section. It is suggested that shielded cable be used for this telephone line connection, and that the shield be connected to pin 13 of J21.

Prior to connecting the equipment to the telephone line, the local telephone company should be notified. You should give the telephone company which line or lines the modem is to be used on, the FCC Registration Number (B468NR-68618-DM-E) and the Ringer Equivalence (0.4B). For further information regarding direct connection to dial-up telephone lines, see the FCC Notice section of the manual.

The direct connect modem board also makes some more serial and parallel input/output available.

The additional serial input/output is an RS232 port on J23. This port can be used to drive another printer or terminal from Basic using the statements PRINT#3, DISPLAY#3, INPUT#3, INKEY$(3), LINE(3) and MAXLINE(3). The speed of this serial port is set up in the set up mode. It is identified as port 3 and can be programmed to various speeds between 50 and 19.2K bits/second.

The RS232 port at J23 has the following pin out:

1 - Shield
2 - Data from terminal
3 - Data to terminal
4&5 are jumpered giving RTS-CTS handshake
7 - Signal Ground

The parallel input/output of the direct connect modem card is available on J21. J21 has the below pin out when used with the direct connect modem card:

1 - Shield
2 - Input 1
3 - Input 2 (also timer input)
4 - Input 3
5 - Input 4
6 - Input 5
7 - Input 6
8 - Output 1
9 - Output 2
10 - Output 3
11 - Output 4
12 - Output 5
13 - Output 6
14 - Output 7
21 - 33 Signal Ground

These I/O lines are from the 2681 DUART on the modem card. The inputs and outputs are at TTL levels. To read the state of an input line from Basic, use the expression (2'N) AND PEEK(39149). N is the bit number (N=6 for input 6). The expression will have the value 0 if the input was low, and a non-zero value if the input was high.

To program an output pin high, use the statement POKE 39151, (2'N). To program an output low, use the statement POKE 39150, (2'N). For further information, see the 2681 data sheet.

Metering & Control Connections

J1 provides control and metering interface for the first five channels. J2 provides the next five channels, J3 provides the next five channels, and so on.

The below table indicates which channels show up on which connector.

Connector	Channels	A/D Board
J1	0 - 4	0
J2	5 - 9	0
J3	10 - 14	1
J4	15 - 19	1
J5	20 - 24	2
J6	25 - 29	2
J7	30 - 34	3
J8	35 - 39	3
J9	40 - 44	4
J10	45 - 49	4
J11	50 - 54	5
J12	55 - 59	5
J13	60 - 64	6
J14	65 - 69	6
J15	70 - 74	7
J16	75 - 79	7
J17	80 - 84	8
J18	85 - 89	8
J19	90 - 94	9
J20	95 - 99	9

Remember the precautions regarding the sample differential and common mode voltages, and the control voltage and current limits. The sample differential voltage is to be less than 2 volts (unless voltage dividers have been added to the A/D board as indicated in the adjustment section). The sample common mode voltage is to be +/- 100 volts from ground. The control circuit open circuit voltage is to be between 0 volts and +30 volts. The control circuit short circuit current is to be limited to 500 mA.

The Raise* or Lower* lines are enabled (pulled low) when a site in the system has the appropriate site and channel selected and presses Raise or Lower. These control lines can be used to adjust the reading being displayed or to make mode changes (ie, transmitter on/off, power change, pattern change, etc.).

The ChanOut* lines are enabled (pulled low) when a site in the system has the appropriate site and channel selected. These lines indicate to external equipment that an A/D sample is being taken. These lines can be used to drive external sample selecting equipment, such as the tower and parameter select lines on a directional antenna monitor. The calibration mode includes a sample delay provision to allow for the settling time of an antenna monitor.

The failsafe* lines are enabled (pulled low) when all sites that have failsafe enabled are operating in the system. The failsafe requirements of each site can be set in the setup mode. See the section on calibration and setup. The failsafe lines will be released if one of the required other DRC units in the system is not responding or if the power supply in this DRC fails. Note that failsafe 1* and failsafe 2* are separate open collector transistor outputs that are driven with the same signal. These can be used to drive different circuits without steering diodes, if desired. Note, however, that Failsafe 1* and Failsafe 1* appearing on J1 and J2 are the same signal. If different circuits are to be driven, steering diodes will be required and the current limit (500 mA) will have to be watched closely. The failsafe signal is duplicated on each A/D board, so it is available on each metering/control connector.

J1 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 0	21	-Sample 0
3	Raise 0*	22	Ground (Control return)
4	Lower 0*	23	ChanOut 0*
5	+Sample 1	24	-Sample 1
6	Raise 1*	25	Ground (Control return)
7	Lower 1*	26	ChanOut 1*
8	+Sample 2	27	-Sample 2
9	Raise 2*	28	Ground (Control return)
10	Lower 2*	29	ChanOut 2*
11	+Sample 3	30	-Sample 3
12	Raise 3*	31	Ground (Control return)
13	Lower 3*	32	ChanOut3*
14	+Sample 4	33	-Sample 4
15	Raise 4*	34	Ground (Control return)
16	Lower 4*	35	ChanOut 4*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J2 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 5	21	-Sample 5
3	Raise 5*	22	Ground (Control return)
4	Lower 5*	23	ChanOut 5*
5	+Sample 6	24	-Sample 6
6	Raise 6*	25	Ground (Control return)
7	Lower 6*	26	ChanOut 6*
8	+Sample 7	27	-Sample 7
9	Raise 7*	28	Ground (Control return)
10	Lower 7*	29	ChanOut 7*
11	+Sample 8	30	-Sample 8
12	Raise 8*	31	Ground (Control return)
13	Lower 8*	32	ChanOut8*
14	+Sample 9	33	-Sample 9
15	Raise 9*	34	Ground (Control return)
16	Lower 9*	35	ChanOut 9*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J3 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 10	21	-Sample 10
3	Raise 10*	22	Ground (Control return)
4	Lower 10*	23	ChanOut 10*
5	+Sample 11	24	-Sample 11
6	Raise 11*	25	Ground (Control return)
7	Lower 11*	26	ChanOut 11*
8	+Sample 12	27	-Sample 12
9	Raise 12*	28	Ground (Control return)
10	Lower 12*	29	ChanOut 12*
11	+Sample 13	30	-Sample 13
12	Raise 13*	31	Ground (Control return)
13	Lower 13*	32	ChanOut 13*
14	+Sample 14	33	-Sample 14
15	Raise 14*	34	Ground (Control return)
16	Lower 14*	35	ChanOut 14*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J4 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 15	21	-Sample 15
3	Raise 15*	22	Ground (Control return)
4	Lower 15*	23	ChanOut 15*
5	+Sample 16	24	-Sample 16
6	Raise 16*	25	Ground (Control return)
7	Lower 16*	26	ChanOut 16*
8	+Sample 17	27	-Sample 17
9	Raise 17*	28	Ground (Control return)
10	Lower 17*	29	ChanOut 17*
11	+Sample 18	30	-Sample 18
12	Raise 18*	31	Ground (Control return)
13	Lower 18*	32	ChanOut 18*
14	+Sample 19	33	-Sample 19
15	Raise 19*	34	Ground (Control return)
16	Lower 19*	35	ChanOut 19*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J5 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 20	21	-Sample 20
3	Raise 20*	22	Ground (Control return)
4	Lower 20*	23	ChanOut 20*
5	+Sample 21	24	-Sample 21
6	Raise 21*	25	Ground (Control return)
7	Lower 21*	26	ChanOut 21*
8	+Sample 22	27	-Sample 22
9	Raise 22*	28	Ground (Control return)
10	Lower 22*	29	ChanOut 22*
11	+Sample 23	30	-Sample 23
12	Raise 23*	31	Ground (Control return)
13	Lower 23*	32	ChanOut 23*
14	+Sample 24	33	-Sample 24
15	Raise 24*	34	Ground (Control return)
16	Lower 24*	35	ChanOut 24*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J6 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 25	21	-Sample 25
3	Raise 25*	22	Ground (Control return)
4	Lower 25*	23	ChanOut 25*
5	+Sample 26	24	-Sample 26
6	Raise 26*	25	Ground (Control return)
7	Lower 26*	26	ChanOut 26*
8	+Sample 27	27	-Sample 27
9	Raise 27*	28	Ground (Control return)
10	Lower 27*	29	ChanOut 27*
11	+Sample 28	30	-Sample 28
12	Raise 28*	31	Ground (Control return)
13	Lower 28*	32	ChanOut 28*
14	+Sample 29	33	-Sample 29
15	Raise 29*	34	Ground (Control return)
16	Lower 29*	35	ChanOut 29*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J7 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 30	21	-Sample 30
3	Raise 30*	22	Ground (Control return)
4	Lower 30*	23	ChanOut 30*
5	+Sample 31	24	-Sample 31
6	Raise 31*	25	Ground (Control return)
7	Lower 31*	26	ChanOut 31*
8	+Sample 32	27	-Sample 32
9	Raise 32*	28	Ground (Control return)
10	Lower 32*	29	ChanOut 32*
11	+Sample 33	30	-Sample 33
12	Raise 33*	31	Ground (Control return)
13	Lower 33*	32	ChanOut 33*
14	+Sample 34	33	-Sample 34
15	Raise 34*	34	Ground (Control return)
16	Lower 34*	35	ChanOut 34*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J8 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 35	21	-Sample 35
3	Raise 35*	22	Ground (Control return)
4	Lower 35*	23	ChanOut 35*
5	+Sample 36	24	-Sample 36
6	Raise 36*	25	Ground (Control return)
7	Lower 36*	26	ChanOut 36*
8	+Sample 37	27	-Sample 37
9	Raise 37*	28	Ground (Control return)
10	Lower 37*	29	ChanOut 37*
11	+Sample 38	30	-Sample 38
12	Raise 38*	31	Ground (Control return)
13	Lower 38*	32	ChanOut 38*
14	+Sample 39	33	-Sample 39
15	Raise 39*	34	Ground (Control return)
16	Lower 39*	35	ChanOut 39*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J9 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 40	21	-Sample 40
3	Raise 40*	22	Ground (Control return)
4	Lower 40*	23	ChanOut 40*
5	+Sample 41	24	-Sample 41
6	Raise 41*	25	Ground (Control return)
7	Lower 41*	26	ChanOut 41*
8	+Sample 42	27	-Sample 42
9	Raise 42*	28	Ground (Control return)
10	Lower 42*	29	ChanOut 42*
11	+Sample 43	30	-Sample 43
12	Raise 43*	31	Ground (Control return)
13	Lower 43*	32	ChanOut 43*
14	+Sample 44	33	-Sample 44
15	Raise 44*	34	Ground (Control return)
16	Lower 44*	35	ChanOut 44*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J10 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 45	21	-Sample 45
3	Raise 45*	22	Ground (Control return)
4	Lower 45*	23	ChanOut 45*
5	+Sample 46	24	-Sample 46
6	Raise 46*	25	Ground (Control return)
7	Lower 46*	26	ChanOut 46*
8	+Sample 47	27	-Sample 47
9	Raise 47*	28	Ground (Control return)
10	Lower 47*	29	ChanOut 47*
11	+Sample 48	30	-Sample 48
12	Raise 48*	31	Ground (Control return)
13	Lower 48*	32	ChanOut 48*
14	+Sample 49	33	-Sample 49
15	Raise 49*	34	Ground (Control return)
16	Lower 49*	35	ChanOut 49*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J11 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 50	21	-Sample 50
3	Raise 50*	22	Ground (Control return)
4	Lower 50*	23	ChanOut 50*
5	+Sample 51	24	-Sample 51
6	Raise 51*	25	Ground (Control return)
7	Lower 51*	26	ChanOut 51*
8	+Sample 52	27	-Sample 52
9	Raise 52*	28	Ground (Control return)
10	Lower 52*	29	ChanOut 52*
11	+Sample 53	30	-Sample 53
12	Raise 53*	31	Ground (Control return)
13	Lower 53*	32	ChanOut 53*
14	+Sample 54	33	-Sample 54
15	Raise 54*	34	Ground (Control return)
16	Lower 54*	35	ChanOut 54*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J12 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 55	21	-Sample 55
3	Raise 55*	22	Ground (Control return)
4	Lower 55*	23	ChanOut 55*
5	+Sample 56	24	-Sample 56
6	Raise 56*	25	Ground (Control return)
7	Lower 56*	26	ChanOut 56*
8	+Sample 57	27	-Sample 57
9	Raise 57*	28	Ground (Control return)
10	Lower 57*	29	ChanOut 57*
11	+Sample 58	30	-Sample 58
12	Raise 58*	31	Ground (Control return)
13	Lower 58*	32	ChanOut 58*
14	+Sample 59	33	-Sample 59
15	Raise 59*	34	Ground (Control return)
16	Lower 59*	35	ChanOut 59*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J13 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 60	21	-Sample 60
3	Raise 60*	22	Ground (Control return)
4	Lower 60*	23	ChanOut 60*
5	+Sample 61	24	-Sample 61
6	Raise 61*	25	Ground (Control return)
7	Lower 61*	26	ChanOut 61*
8	+Sample 62	27	-Sample 62
9	Raise 62*	28	Ground (Control return)
10	Lower 62*	29	ChanOut 62*
11	+Sample 63	30	-Sample 63
12	Raise 63*	31	Ground (Control return)
13	Lower 63*	32	ChanOut 63*
14	+Sample 64	33	-Sample 64
15	Raise 64*	34	Ground (Control return)
16	Lower 64*	35	ChanOut 64*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J14 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 65	21	-Sample 65
3	Raise 65*	22	Ground (Control return)
4	Lower 65*	23	ChanOut 65*
5	+Sample 66	24	-Sample 66
6	Raise 66*	25	Ground (Control return)
7	Lower 66*	26	ChanOut 66*
8	+Sample 67	27	-Sample 67
9	Raise 67*	28	Ground (Control return)
10	Lower 67*	29	ChanOut 67*
11	+Sample 68	30	-Sample 68
12	Raise 68*	31	Ground (Control return)
13	Lower 68*	32	ChanOut 68*
14	+Sample 69	33	-Sample 69
15	Raise 69*	34	Ground (Control return)
16	Lower 69*	35	ChanOut 69*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J15 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 70	21	-Sample 70
3	Raise 70*	22	Ground (Control return)
4	Lower 70*	23	ChanOut 70*
5	+Sample 71	24	-Sample 71
6	Raise 71*	25	Ground (Control return)
7	Lower 71*	26	ChanOut 71*
8	+Sample 72	27	-Sample 72
9	Raise 72*	28	Ground (Control return)
10	Lower 72*	29	ChanOut 72*
11	+Sample 73	30	-Sample 73
12	Raise 73*	31	Ground (Control return)
13	Lower 73*	32	ChanOut 73*
14	+Sample 74	33	-Sample 74
15	Raise 74*	34	Ground (Control return)
16	Lower 74*	35	ChanOut 74*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J16 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 75	21	-Sample 75
3	Raise 75*	22	Ground (Control return)
4	Lower 75*	23	ChanOut 75*
5	+Sample 76	24	-Sample 76
6	Raise 76*	25	Ground (Control return)
7	Lower 76*	26	ChanOut 76*
8	+Sample 77	27	-Sample 77
9	Raise 77*	28	Ground (Control return)
10	Lower 77*	29	ChanOut 77*
11	+Sample 78	30	-Sample 78
12	Raise 78*	31	Ground (Control return)
13	Lower 78*	32	ChanOut 78*
14	+Sample 79	33	-Sample 79
15	Raise 79*	34	Ground (Control return)
16	Lower 79*	35	ChanOut 79*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J17 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 80	21	-Sample 80
3	Raise 80*	22	Ground (Control return)
4	Lower 80*	23	ChanOut 80*
5	+Sample 81	24	-Sample 81
6	Raise 81*	25	Ground (Control return)
7	Lower 81*	26	ChanOut 81*
8	+Sample 82	27	-Sample 82
9	Raise 82*	28	Ground (Control return)
10	Lower 82*	29	ChanOut 82*
11	+Sample 83	30	-Sample 83
12	Raise 83*	31	Ground (Control return)
13	Lower 83*	32	ChanOut 83*
14	+Sample 84	33	-Sample 84
15	Raise 84*	34	Ground (Control return)
16	Lower 84*	35	ChanOut 84*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J18 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 85	21	-Sample 85
3	Raise 85*	22	Ground (Control return)
4	Lower 85*	23	ChanOut 85*
5	+Sample 86	24	-Sample 86
6	Raise 86*	25	Ground (Control return)
7	Lower 86*	26	ChanOut 86*
8	+Sample 87	27	-Sample 87
9	Raise 87*	28	Ground (Control return)
10	Lower 87*	29	ChanOut 87*
11	+Sample 88	30	-Sample 88
12	Raise 88*	31	Ground (Control return)
13	Lower 88*	32	ChanOut 88*
14	+Sample 89	33	-Sample 89
15	Raise 89	34	Ground (Control return)
16	Lower 89*	35	ChanOut 89*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J19 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 90	21	-Sample 90
3	Raise 90*	22	Ground (Control return)
4	Lower 90*	23	ChanOut 90*
5	+Sample 91	24	-Sample 91
6	Raise 91*	25	Ground (Control return)
7	Lower 91*	26	ChanOut 91*
8	+Sample 92	27	-Sample 92
9	Raise 92*	28	Ground (Control return)
10	Lower 92*	29	ChanOut 92*
11	+Sample 93	30	-Sample 93
12	Raise 93*	31	Ground (Control return)
13	Lower 93*	32	ChanOut 93*
14	+Sample 94	33	-Sample 94
15	Raise 94	34	Ground (Control return)
16	Lower 94*	35	ChanOut 94*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J20 Connections

Pin	Description	Pin	Description
1	Shield	20	Ground (Control return)
2	+Sample 95	21	-Sample 95
3	Raise 95*	22	Ground (Control return)
4	Lower 95*	23	ChanOut 95*
5	+Sample 96	24	-Sample 96
6	Raise 96*	25	Ground (Control return)
7	Lower 96*	26	ChanOut 96*
8	+Sample 97	27	-Sample 97
9	Raise 97*	28	Ground (Control return)
10	Lower 97*	29	ChanOut 97*
11	+Sample 98	30	-Sample 98
12	Raise 98*	31	Ground (Control return)
13	Lower 98*	32	ChanOut 98*
14	+Sample 99	33	-Sample 99
15	Raise 99	34	Ground (Control return)
16	Lower 99*	35	ChanOut 99*
17	Failsafe 1*	36	Ground (Control return)
18	Failsafe 2*	37	Ground (Control return)
19	No connection		

J22 Connections

J22 is an RS-232/C subset connector. This connector gives the user access to the Basic interpreter included in the DRC190. The baud rate on the port is programmed through the calibration and setup procedure. Other data parameters include: 1 start bit, 7 data bits, mark parity, 1 stop bit.

Pin	Description
2	Data from terminal
3	Data to terminal
7	Data ground

Connector Supplies

The below listed connector parts are supplied or are available from H&F.

H&F P/N	Description	Manufacturer Part Number
2100-0016	25 pin male D connector	AMP 205208-1
2100-0017	pins for D connectors	AMP 205202-4
2100-0018	25 pin connector back shell	AMP 205718-1
2100-0091	37 pin male D connector	AMP 205210-1
2100-0092	37 pin connector back shell	AMP 205731-1
2100-0093	pin insertion/extraction tool	AMP 91067-2
2100-0094	crimp tool (not supplied)	AMP 29004-1

CRT & Printer Cables

H&F provides the Liberty Electronics Freedom 100 CRT terminal and the Smith Corona D300 printer with DRC systems, unless other equipment is specified. The CRT terminal is wired to the DRC190, while the printer is connected to the peripheral port on the CRT terminal. The DRC190 prints by enabling the peripheral port, sending the data to be printed, then sending the peripheral port disable character sequence.

The CRT terminal main port is wired to J22 on the DRC190 using a cable described below:

DRC190 J22	Function	Freedom 100 Main Port
1	Shield, Frame Ground	1
2	Terminal data to DRC	2
3	DRC data to terminal	3
4	Request to Send to DRC	4
5	Clear to Send to terminal	5
6	Data Set Ready to terminal	6
7	Signal Ground	7

In addition, the CRT terminal pins 8 (Carrier Detect Input) is jumpered to pin 20 (Data Terminal Ready Output).

If no printer is used with the Freedom 100, print statements are used in the DRC190 program, and the DRC190 has been programmed with the peripheral port on and off codes, a dummy plug must be plugged into the Freedom 100 peripheral port to allow it to complete the required handshaking. Failure to install this plug will cause the Freedom 100 to stop operating while waiting for the non-existant printer to be ready for data. This dummy plug has a jumper between pin 6 and 19, and another jumper between pins 8 and 20.

Finally, if a printer is used, the cable is wired as below:

Freedom 100 Pin	Function	SC Printer Pin
1	Shield - Frame Ground	1
2	Data to Freedom 100 (handshake)	2
3	Data to printer	3
4	Hi=OK, Lo=Busy, to Freedom 100	4
6	Data Set Ready to printer	6
7	Signal Ground	7
8 & 20	Jumper terminal DCD to DTR	
19	Hi=OK, Lo=Busy, to Freedom 100	20

The suggested printer DIP switch settings are listed below, where U indicates "up" and D indicates "down".

Switch	SW-1	SW-2	SW-3
1	D	D	D
2	U	U	U
3	D	D	U
4	U	D	D
5	D	U	
6	U	U	
7	U	U	
8	U	D	

Calibration and Setup

The DRC190 permits metering channels to be calibrated by establishing a scaling factor that the A/D sample is multiplied by. This scaling factor is determined by the DRC firmware dividing the desired indication by the A/D sample. This floating point scaling factor is stored in Electrically Erasable Programmable Read Only Memory (EEPROM). In addition, the displayed reading can be either proportional to the A/D sample or proportional to the square of the A/D sample (square law), allowing direct reading of power output from a reflectometer sample. Finally, a three character label and a two character units designator are stored for each channel.

The Setup mode allows changing the site number of this particular unit, the number of the highest site number in the system; the terminal and communications baud rate, the "site delay", fail safe enable, control enable, and the Morse CW identifier code.

The below procedure describes the calibration and setup of the DRC190.

1. Key in the number sequence "1 2 3 4" and press the decimal point on the front panel keyboard. The "1 2 3 4" sequence makes it difficult to get in the calibrate mode accidentally. The display will indicate you have entered the setup mode and instruct you to use up-arrow to answer yes, and down arrow to answer no. In addition, the front panel speaker is enabled. Any audio received on the communications link will be heard through the speaker, allowing a quick analysis of signal quality. The speaker is disabled on receipt of a speaker mute command (at the end of an intercom or CW ID message) or the pressing of any key.

2. The display asks if you wish to continue calibration. If so, press up arrow. If not, press down arrow and skip to step 10.

3. Enter a 2 digit channel number of the channel you wish to calibrate.

4. The display will show the currently programmed label and units. The first three characters are the label (ie, ICP for common point current), and the last two characters are the units (ie, KV for kilovolts). If the character that the cursor is under is as desired, press the up-arrow for yes. If it is not, press the down-arrow for no. Yes will advance the cursor to the next character position. No will change the character without moving the cursor. Hold the No key down until the desired character appears, the press the Yes key. Each label and unit character is programmed in this manner.

5. Once the label and units have been programmed, you will be asked if the current sample delay is ok. The sample delay is the number of analog to digital conversions that are thrown out before the sample is taken. The CHANOUT* output of the A/D board is enabled at the start of the first conversion. If this output is used to drive the tower select on an antenna monitor, the sample is not immediately available. A/D conversions can be thrown out until the reading should be stable. Each conversion takes about a third of a second. Setting sample delay to 03 allows the antenna monitor a settling time of 1 second. In addition, the Raise and Lower outputs are pulsed for a period of time corresponding to the sample delay. If the sample delay is 03 and a Raise is sent, the appropriate Raise* output of the DRC190 will be held low for 1 second. The sample delay must be between 00 (no delay) and 15 (5 seconds). If the current delay is ok, hit yes. If not, hit no and then enter the 2 digit delay desired.

6. Once the sample delay has been programmed, you will be asked if the linear or square law curve is ok. If not, press down-arrow until the appropriate curve is displayed. Then press up-arrow.

7. You will then be asked if a RAISE or LOWER is required during the sample. This will cause a RAISE or LOWER to be generated during the time we are measuring the sample voltage in the calibration routine. It does not cause a RAISE or LOWER to be generated each time the meter is read.

8. The reading with the current scaling factor will then be displayed. Note that for the best accuracy, the sample voltage should be at its highest expected voltage. A good way of calibrating the system with an antenna monitor is by pressing the 180 degree phase calibrate button on the antenna monitor prior to answering the RAISE/LOWER question above. If this is correct, press up-arrow. If it is not, press down-arrow. The correct reading can then be entered as a five character number, including the decimal point. The sample voltage being calibrated against must be greater than 10 mV (required to get a resolution of 1% on the A/D) and less than 1.900 volts (required to prevent A/D overrange, which occurs at 2.000 volts). In addition, the calibrated indication cannot be 0.

9. The DRC190 will then ask if you wish to continue calibration. This is the same as step 2. If you answer yes, you will be back at step 3. If you answer no, you will be at step 10.

10. The DRC190 will then ask if you want to do setup. If you answer no, you will be dumped back into normal operation. If you answer yes, you will proceed to step 11.

11. The DRC190 will ask if the programmed maximum site number is correct. If not, answer no and enter the highest site number in the system as a 2 digit number.

12. The DRC190 will ask if the programmed site number is correct. If not, answer no and enter the correct site number as a 2 digit number.

13. The DRC190 will ask if the programmed site delay is correct. A typical wire line system, or a system with full time subcarrier channels will use a site delay of 0.05 seconds. A system using the Standard Communications FX60U radio link will use a site delay of 0.25 seconds to allow for keying and squelch delays.

14. The DRC190 will ask if the programmed communications baud rate is correct. If not, answer no and the next choice will be displayed. Answer no to see the next choice or answer yes to accept the currently displayed baud rate. The DRC190 normally operates at 1.2 Kbps (1,200 bits per second), but can be slowed down for marginal communications links.

15. The DRC190 will ask if the programmed terminal baud rate is correct. Answer in a manner similar to question 12. The DRC190 is normally shipped with the terminal baud rate set to 9.6 Kbps.

16. The DRC190 will ask if the programmed port 3 baud rate is correct. Port 3 is the RS232 port on the direct connect modem card. This port is

accessed using PRINT#3, INPUT#3, INKEY$(3), LINE(3) and MAXLINE(3). It is typically used to drive a printer. The speed of this port (if supplied) is changed in the same manner as the other speeds.

17. The DRC190 will then ask if you wish to change the failsafe or control enable programming. If so, answer yes, if not, answer no. The DRC190 will ask if you wish to have failsafe enabled from each of the sites in the system. Answer yes for each site that the failsafe is to be enabled on, and no if the failsafe is not to be enabled. The DRC190 will release its failsafe output if it does not hear from an enabled site at least once every minute. After the highest site number in the system has had failsafe enabled or disabled, the DRC will ask if Raise or Lower should be enabled from each site. Answer yes or no as apporpriate. This programing feature prevents an "AM rock jock" from shutting down a TV station in the same DRC system.

18. The DRC190 will then ask if you wish to change the CW ID. If you don't have the CW ID option (used in TRL radio systems), you can answer no. It does not hurt to answer yes, but without the optional hardware, you'll get no CW ID but the DRC190 may take time on the communications link to try do the the ID anyway. If you have no ID option, the time between IDs should be set to 00 which indicates IDs are not to be done. If you do have the CW ID option and wish to make a change, answer yes.

19. The DRC190 will first ask how many minutes should be between IDs. Answer 00 if no IDs are to be done. The typical ID frequency is 60 minutes. You can enter anything between 01 minutes and 99 minutes.

20. The DRC190 will then ask if the ID character over the cursor is that desired. This is similar to setting the labels and units in the calibrate section. Holding the No key rotates through the various characters available. Pressing the Yes key moves the cursor to the next character position. There is room for up to a 16 character ID. The [character is used to indicate the end of the message. If the ID is 16 characters long, no [is needed. A typical ID would be DE K1234[.

21. The DRC190 will then ask if you wish to set the time, day and date. This allows setting the time, day and date in the Basic program without having to use an ASCII keyboard (some stations have the DRC190 drive a receive only printer only for logging). If you answer yes, you'll be asked to enter the time as HHMMSS (hours, minutes, seconds) in 24 hour format. You'll then be asked to enter the day as a number between 0 and 6 where 0 is Sunday, 1 is Monday, etc. You'll then be asked to enter the date in YYMMDD (year, month, date) format.

If a CRT terminal or printer, but not both, is to be connected to this site, the following instructions for setting up CRTSTR and PRTSTR can be skipped.

If a CRT and printer are both connected, the printer is plugged into the preipheral port of the CRT terminal. When a Basic program is to put information on the CRT, the command DISPLAY (or PRINT#0) is used. When information is to be put on the printer, the command PRINT (or PRINT#1) is used. When a program changes between DISPLAYing data and PRINTING data, the terminal's peripheral port is enabled and disabled. This is done by the DRC190 sending PRTSTR to the CRT to enable the peripheral port, or CRTSTR to the CRT to disable the peripheral port. CRTSTR and PRTSTR are stored in the non-volatile EEPROM. If Hallikainen & Friends supplied the terminal with the system, PRTSTR and CRTSTR have already been set up.

CRTSTR and PRTSTR are set up using POKE commands from Basic to store the desired codes in EEPROM. Each code can be up to 4 characters long and must be terminated with a 00 character after the last character. CRTSTR is located at $A40F and PRTSTR is located at $A40A.

If, for example, we wish to program the DRC190 for use with the Liberty Electronics Freedom 100, the following procedure would be used.

Plug the CRT terminal into J22 on the DRC190. Press the RETURN key several times to insure that the DRC190 is echoing the terminal.

The Freedom 100 requires the following codes to enable the peripheral port: ESCAPE ° Consulting the ASCII-Decimal conversion table (section 12), we find that ESCAPE converts to 27 while ° converts to 96. In addition, $A40A converts to 41994 decimal. Therefore, the following code typed into the terminal will handle the peripheral port enable.

```
POKE 41994,27
POKE 41995,96
POKE 41996,0
```

The Freedom 100 requires the following codes to disable the peripheral port: ESCAPE a . We find that a converts to 97 decimal. In addition, $A40F converts to 41999 decimal. Therefore, the following codes typed into the terminal will handle the peripheral port disable.

```
POKE 41999,27
POKE 42000,97
POKE 42001,0
```

After these have been entered, you should be able to type the following lines on the terminal and have the appropriate text show up on the terminal and the printer.

```
PRINT "This should appear on the printer!"
DISPLAY "This should appear on the terminal!"
```

You may at this point also wish to set the clock and calendar in this DRC unit. Time and Date are entered as 6 digit numbers with the most significant digit first. For time, the format is HHMMSS. For date, the format is YYMMDD. For example, if the time is 2:35:07 PM, and the date is November 11, 1984, I'd type in the following:

```
TIME=143507
DATE=841111
```

In addition, the day of week is set as a single digit number between 0 and 6. Sunday is 0, Saturday is 6. For today, I'd type in:

```
DAY=0
```

We can check all of this by typing:

```
DISPLAY TIME$, DAY$, DATE$
```

The DRC190 should respond with:

```
2:35:07 PM     Sunday     November 11, 1984
```

Of course, a few seconds has probably elapsed since the clock was set, so the time (or, if it's real late at night, the day and date) will have advanced some.

Use With External Computer

The DRC190 RS232 port may be connected to an external computer instead of a terminal/printer combination. When this is done, the external computer is typically not expecting the DRC190 input to be echoed, line feeds to be added, or an OK message to be output after each request. For this reason, a byte of EEPROM is set aside as BASECHO. If BASECHO is non-zero, the DRC190 RS232 port handles a terminal/printer combination. If BASECHO is zero, the DRC190 does not echo the command from the external computer, does not add line feeds to carriage returns, and does not give the OK message. See section 4 for more information on using the DRC190 with an external computer.

To use the DRC190 with an external computer, set BASECHO to zero by typing the following command:

```
POKE 42021,0
```

To return the DRC190 to normal, set BASECHO non-zero by typing the following command:

```
POKE42021,255
```

Sample Basic Programs

Each unit of the DRC190 system includes a Basic Interpreter and an RS-232 port to allow connection of a CRT or Printer. If the CRT has a peripheral port that can be enabled and disabled under software control, a printer can be connected to this port. When the peripheral port is used in this manner, the peripheral port on and off codes must be programmed into the DRC190 EEPROM as outlined in the calibration section of the manual.

Each DRC190 unit has 8 Kbytes of RAM (2 Kbytes on revision A processor boards) at a minimum. This is enough memory to write some simple logging or display programs. Additional memory is available as an option to allow for larger programs.

If we wish to continuously update a CRT display with the readings at several sites, we can use the below listed program. This program assumes we are interested in displaying ten channels of metering (channels 0..9) from sites 1, 2, and 3 of our system.

```
10DISPLAY"Channel","Site 1","Site 2","Site 3"
20FORC=0TO9
30DISPLAYC,
40FORS=1TO3
50DISPLAYMETER$(S,C),
60NEXT
70DISPLAY
80NEXT
90DISPLAYCHR$(30)
100GOTO10
```

Note that no spaces were put between words. Although this makes the program a little difficult to read, it saves memory.

The program works like this:

Line 10 puts up headings labelling the columns for channel number and the site numbers.

Lines 20 and 40 cause the remainder of the program to access Site 1 Channel 0 followed by Site 2 Channel 0. . . then Site 1 Channel 1. This access method results in a nicer screen display. Lines 10, 20 and 40 can be changed to display more or fewer sites and channels.

Line 30 displays the channel number we are currently accessing on the left side of the screen.

Line 50 actually takes the meter reading at Site S, Channel C and displays it. The use of METER$ instead of METER results in a string containing the label and units programmed in during calibration to be printed, rather than a numeric value alone. The comma following the METER$ causes the next information to be printed to go in the next tab field on the screen.

Line 60 causes line 40 to set S (the site to be accessed) to the next number. Execution then continues immediately after line 40.

Line 70 forces the next data to be displayed to the next line, overriding the comma following the METER$ once all the sites have been displayed.

Line 80 causes line 20 to set up the next channel to be read. Execution then continues immediately after line 20.

Line 90 sends the "home cursor" code to the terminal. The code used here (30) applies to the Liberty Electronics Freedom 100. It will probably have to be changed for different terminals.

Finally, line 100 causes the entire program to be run again.

To get out of the program at any time, press the control key and the C key at the same time (CTRL-C).

Note that if the DRC attempts to access a site that is not in the system, it will continue to try for about 60 seconds, then return a value of 1 and set the ERR flag. After an error has been detected, METER is set to 1 and METER$ is "ERR=1!!". For this reason, line 40 should be modified to only access sites that are in the system. Line 10 should be modified to only display headings for sites that will actually be shown.

A simple program to print a log appears below. Text headings, dates, and other niceties have been left off this to try to fit it in the minimum memory of a system. The program can, of course, be expanded to include headings, limit checking, alarms, etc., if sufficient memory is available.

```
10IFTIME<RTTHENRT=-9E9
20IFTIME<RT+3000THEN10
30RT=TIME
40PRINTTIME$,
50FORC=0TO9
60PRINTMETER$(1,C),
70NEXT
80PRINT
90GOTO10
```

On startup, RT and all other variables are zero. Since the time is a positive integer, it is never less than 0 and line 10 is not executed. If the time is less than RT+3000 (reading time + 30 minutes), then line 20 loops the program back to line 10. If 30 minutes have elapsed, line 20 allows the program to drop through to line 30 where RT is updated to the current time. Line 40 prints the current time in string form (12 hour am/pm) and the comma leaves the printhead at the next tab stop. Lines 50 through 70 form a FOR-NEXT loop with C (the metering channel) varying between 0 and 9. Line 60 prints the appropriate channel of site 1, tabbing over after printing each. Line 80 forces a carriage return and line feed after the last channel has been printed. Line 90 loops the program back to the beginning, where it waits for another 30 minutes to elapse, or for time to be less than RT.

Line 10 waits for time to be less than RT. If this is the case, RT is set to -9E9 (-9 x 10 to the 9). Since -9E9 is substantially less than any valid time, it will foce line 20 to allow execution of the program to drop through. Since RT is updated to time each time the readings are printed, time will not be less than RT until the beginning of a new day, when time goes to 0 and RT remains at 233000 or so. Line 10 forces the program to print readings at the beginning of a new day.

Note that time is stored in HHMMSS format. If time is 120500 (12:05:00 PM), TIME+3000 is 123500 (12:35:00 PM). This is indeed one half hour from when the readings were last printed. However, what happens if we now add another 30 minutes? 123500+3000=126500 (12:65:00 PM). TIME, however, goes from 125959 to 130000, which is greater than 126500, causing a log print at the top of the hour. This is generally no problem. If you wish, however, you can write a routine in Basic to handle this base 60/base 10 problem.

If you are using the DRC190 without a disc drive, you may wish to save the logging program in EEPROM. There is about 1 Kbyte of free EEPROM for basic storage. To save your program, type

```
SAVE EEPROM
```

To load a program from EEPROM, type

LOAD EEPROM

The program in EEPROM is automatically loaded and run on system reset.
The SAVE and LOAD EEPROM statements given above use the "boot" area of the EEPROM on the processor board. If you have a RAM board that is partially loaded with EEPROM, the instructions can be modified to use this EEPROM. For example,

SAVE EEPROM 24576 will save the program in EEPROM residing at 24576 (6000 Hex).

LOAD EEPROM 24576 will load that program into RAM.

If using a program in EEPROM on the RAM board, the boot program could be:

```
10 LOAD EEPROM 24576
```

This would cause the DRC190 to load and execute the boot program on power up. The boot program would load and execute the actual program (residing in EEPROM at 24576).

If you are using a DRC190 without a disc drive, you may wish to download Basic programs to the DRC190 from another computer. Since the DRC190 Basic interpreter compacts each line of code after it is entered, programs cannot be downloaded at high speed. The download program (located in the host computer) should check for the echo of the first character of each line sent to the DRC190. This character is typically the line number. Once this character has been echoed, the DRC190 has finished "crunching" the last line entered and is ready for the current line. A suggested program (written in Pascal) is listed below:

```
PROGRAM ToPCC;

CONST
  StatPort = 16#20;
  DataPort = 16#21;

TYPE
  SType = STRING[132];

VAR
  Compiled,
  InFile : TEXT;
  NLine,
  FileName,
  Line : SType;
  Question : SType;
  Default,
  Process,
  InQuotes : BOOLEAN;

  PROCEDURE PrQuest;

    BEGIN
      IF LENGTH(Question) > 1
```

```
   THEN
    DELETE(Question, 2, LENGTH(Question) - 1);

 END;

PROCEDURE WCompLin(VAR Line : SType);

 ¼ Write compiled line;  This routine will write a line to the PCC if that
   is what the user selected.  It send the first character, wait for the PCC
   to echo it, and send the rest of the line.  Otherwise, it will just do
   a WRITELN to Compiled.  ¶

 VAR
  TestCh : CHAR;
  I      : INTEGER;

 FUNCTION InChar(Dummy : INTEGER) : CHAR;

  ¼ Input a character from the PUNCH: port.  ¶

  BEGIN
   REPEAT
    INPORT(StatPort, Dummy)
   UNTIL (Dummy MOD 128) > 63;
   INPORT(DataPort, Dummy);
   InChar := CHR(Dummy MOD 128)
  END;

 BEGIN ¼WCompLin¶
     WRITE(Line[1]);
     WRITE(Compiled, Line[1]);
     REPEAT
     UNTIL InChar(0)=Line[1];
     FOR I := 2 TO LENGTH(LINE) DO
      BEGIN
       WRITE(Line[I]);
       WRITE(Compiled, Line[I])
      END;
     WRITELN;
     WRITE(Compiled, CHR(13));
 END;

PROCEDURE PrLine;

 VAR
  I : INTEGER;

 BEGIN
  NLine := '';
  IF (Line < '999') AND (POS(' ',Line) = 4)
   THEN
    NLine := Line
   ELSE
    BEGIN
     InQuotes := FALSE;
```

```
            Line := CONCAT(Line, '  ');
            FOR I := 1 TO LENGTH(Line) - 2 DO
              BEGIN
                IF Line[I] = '"'
                  THEN
                    InQuotes := NOT InQuotes;
                IF (COPY(Line, I, 3) = 'REM') AND NOT InQuotes
                  THEN
                    I := LENGTH(Line);
                IF (Line[I] <> ' ') OR InQuotes
                  THEN
                    NLine := CONCAT(NLine, COPY(Line, I, 1))
              END;
            IF NLine[LENGTH(NLine)] = ':'
              THEN
                DELETE(NLine, LENGTH(NLine), 1)
          END;
      Line := NLine
    END;

  PROCEDURE WTitle;

    BEGIN
      WRITE(CHR(27), 'E');
      WRITELN('               ********************************************');
      WRITELN('               *                                          *');
      WRITELN('               *         PCC TRANSFER PROGRAM             *');
      WRITELN('               *                                          *');
      WRITELN('               ********************************************');
      WRITELN;
      WRITELN;
      WRITELN;
      WRITELN;
      WRITELN;
      WRITELN;
      WRITELN;
      WRITELN('       A William F. Foote/Hallikainen & Friends in house production.');
      WRITELN;
      WRITELN

    END;

BEGIN
  WTitle;
  WRITE('What file do you want to transfer to the PCC?  ');
  READLN(FileName);
  Default := FALSE;
  IF FileName[LENGTH(FileName)] = ',' then begin
    delete(FileName, length(FileName), 1);
    Default := true
  end;
  if pos('.', FileName) = 0
    THEN
      FileName := CONCAT(FileName, '.OBJ');
  REWRITE(Compiled, 'PUNCH:');
  WRITELN;
  WRITE('Do you want me to remove the blanks from ', FileName, '?  ');
  IF Default
    THEN
      BEGIN
        Question := 'No';
        WRITELN(Question)
      END
    ELSE
      READLN(Question);
  PrQuest;
  Process := (Question = 'Y') OR (Question = 'y');
  WRITELN;
  RESET(InFile, FileName);
  WHILE NOT EOF(InFile) DO
    BEGIN
      READLN(InFile, Line);
      IF Process
        THEN
          PrLine;
      IF Line <> ''
        THEN
          WCompLin(Line)
    END;
');

  CLOSE(InFile);
  CLOSE(Compiled)
END.
```

Finally, it may be desired to use an external computer to access DRC190 data through the RS-232 port. This can be accomplished by using a "null modem" cable and having the external computer appear to the DRC190 as a terminal. This "terminal" is then making immediate data requests of the DRC190. To make the DRC190 actually appear to be a terminal to the external computer, the DRC190 RS232 echo should be disabled. This is done using a POKE statement as outlined in the Calibration section of this manual. This POKE clears a flag called BASECHO in EEPROM. Since it is in EEPROM, the setting of this flag will be retained through power failures.

When BASECHO is non-zero, the DRC190 echoes each character input through the RS232 port. In addition, it adds line feeds to carriage returns and says "OK" after executing an immediate statement. When BASECHO is zero, character echo is suppressed, line feeds are not added to carriage returns, and the "OK" message is suppressed.

With a non-zero BASECHO, the following would appear on the RS232 port output in response to a meter request:

```
?METER$(1,3)
[CR][LF]
ABC=1.2345DE
[CR][LF]
[CR][LF]
OK
[CR][LF]
```

where [CR] is carriage return and [LF] is line feed, ABC is programmed label, DE is programmed units, and 1.2345 is the reading. Note that ? is the same as typing PRINT.

If BASECHO is zero, a similar request would be handled as below:

```
ABC=1.2345DE[CR]
```

Note that this is the way an operator would request to a prompted input request from a typical high level language program. The operator puts in the requested data followed by a carriage return. The DRC190 with BASECHO=0 does the same.

A program that demonstrates acquiring data from the DRC190 is shown below.

```
5  DIM M(4) :REM Dimension an array to hold readings
10 S=1 :REM Set site to 1
20 FOR C=0 TO 4 :REM Scan channels 0 thru 4
30 GOSUB 200 :REM Go get the reading
40 M(C)=M :REM Store the reading in an array
50 NEXT :REM Get the next channel
60 FOR N=1 TO 24: PRINT: NEXT :REM Scroll stuff off screen
70 FOR C=0 TO 4: PRINT C,M(C) :REM Display channel and reading
80 NEXT : Display next channel and reading
90 GOTO 10 :REM Go do it again

198 REM Subroutine to get reading.  Call with Site & Channel.
199 REM return with Meter and ERror code.
200 PRINT CHR$(4);"PR#4" :REM Select SSC in slot 4 for output
210 PRINT CHR$(1);"E D" :REM Prevent outgoing data from echoing
220 PRINT "?METER("; S; ","; C; ");CHR$(44);ERR"
222 REM Above line sends request to DRC. DRC gets
223 REM ?METER(1,3);CHR$(44);ERR  assuming S=1, C=3
230 PRINT CHR$(4);"PR#0": REM Send further output to screen
231 REM preventing input prompt ("?") from going to DRC
240 PRINT CHR$(4);"IN#4": REM Get input from SSC in slot 4
250 INPUT M, ER :REM Get meter reading and error code.
260 PRINT CHR$(4);"IN#0": REM Take further input from keyboard
270 RETURN
```

This program assumes an Apple Super Serial Card in Slot 4 of an Apple 2. The card should be set up as below:

```
Switch 1
     1 - Off  Baud Rate = 4800 Bits per second
     2 - Off
     3 - On
     4 - On
     5 - On   Communications Mode
     6 - On
     7 - On   Standard RS-232

Switch 2
     1 - On   Data Format:  7 Data, 1 Stop
     2 - Off
     3 - Off  Even Parity
     4 - Off
     5 - Off  Do not generate LF after CR
```

6 - Off Interrupts Off
7 - Off Standard RS-232

The MODEM-TERMINAL jumper should be pointing UP towards MODEM.

Note that the SSC is set for 4800 bits per second. The DRC190 should, of course, be set for the same speed using the Calibration and Set Up mode. The choice of RS-232 speed is a trade off. Too high a speed may introduce glitches in long cables between the computer and the DRC190. Too low a speed will result in slow update times for the computer.

Note that program line 250 uses an INPUT statement to collect data from the DRC190. The INPUT statement sends a ? prompt to the output device. We keep this from getting to the DRC190 by returning output to the screen in line 230. The ? does less harm on the screen than it does to the DRC190. In addition, the INPUT statement echoes received characters to the screen. This results in a bit more garbage on the screen. It is possible to use GET statements to get the DRC190 data character by character without echoing to the screen. We found that the time taken by Basic to add the latest character to the string, check for a carriage return and to GOTO back to the GET statement severely limited the speed of communications from the DRC190 to the Apple. It may be worthwhile playing with assemmbly routines, other Basic methods, or a compiled langauage to get a clean screen and high speed. Finally, the INPUT statement will hang if it does not get the expected input. A GET statement would also hang when the DRC190 stops sending data. Perhaps something similar to INKEY$ with a loop counter could be used to make the program more fail-safe.

Direct Connect Modem Interface Software

The manual on the Cermetek CH1770 is reprinted in the rear of this manual. Refer to this documentation for further information on using the modem. This section will briefly cover the typical Basic statements used to interface to the modem.

The modem appears as device number 2 to Basic. It can be accessed using the below listed words:

PRINT#2,	Sends data to the modem
INPUT#2,	Receives data from the modem
INKEY$(2)	Gets a single character from the modem
LINE(2)	Returns the line number of the modem device
MAXLINE(2)	Sets the maximum line number used by modem
MDMSPD	Sets the modem speed (3=300 BPS, 12=1200 BPS)

All commands sent to the modem must be preceeded by a control-N (CHR$(14)). A few of the typical commands are listed below:

PRINT#2, CHR$(14);"D 'TB(805)541-0200'"	Originate a call to H&F
PRINT#2, CHR$(14);"A"	Force modem to answer
MDMSPD=3	Change to 300 BPS
MDMSPD=12	Change to 1200 BPS

The modem is initiallized to give unsolicited status messages. This is done so that speed change messages are properly received. .pn 1

Input/Runtime

Input/Runtime is a set of programs under development for the DRC190. Input will generate a set of disc files describing the operation of each station under each possible state (ie, Main TX Day, Main TX Night, Aux TX Day, Aux TX Night, Down). Runtime then uses these files to perform the requested logging, alarming and control of the station.

At this writing, the disc drive data storage and retrieval firmware has not been written. This prevents Input/Runtime from being tested. Be sure to complete the system registration form at the beginning of the book to receive notices on software updates that will enable the operation of the disc drive and Input/Runtime.

Programming in H&F BASIC

The DRC uses a slightly extended BASIC interpretter written by Microsoft. It is popularly called the 6800 8K BASIC. This section provides an introduction to H&F BASIC. It is not intended to be a detailed course in BASIC programming. It will, however, serve as an excellent introduction for those unfamiliar with the language.

The text here will introduce the primary concepts and uses of H&F BASIC to get you started writing programs.

We recommend that you try each example in this section as it is presented. This will enhance your "feel" for H&F BASIC and how it is used.

After powering up the DRC, it should print a message followed by OK. If not, press and release the RESET button on the rear panel.

> NOTE: All commands to H&F BASIC should end with a carriage return. The carriage return tells H&F BASIC that you have finished typing the command. If you make a typing error, type a back-space to eliminate the last character. Repeated use of back-space will eliminate previous characters. A control-U (U typed with control key pressed) will eliminate the entire line that you are typing.

Now, try typing the following:

```
PRINT 10-4   (end with carriage return)
```

The DRC will immediately print:

```
6

OK
```

The print statement you typed in was executed as soon as you hit the carriage return key. The DRC evaluated the formula after the "PRINT" and then typed out its value, in this case 6.

Now try typing in this:

```
PRINT 1/2,3*10          ("*" means multiply, "/" divide)
```

The DRC will print:

```
.5   30
```

As you can see, the DRC can do division and multiplication as well as subtraction. Note how a " , " (comma) was used in the print command to print two values instead of just one. The comma divides the 132 character line into 10 columns, each 13 characters wide. The result is a " , " causes the DRC to skip to the next character field on the printer, where the value 30 was printed.

Commands such as the "PRINT" statements you have just typed in are called Direct Commands. There is another type of command called an Indirect Command. Every Indirect Command begins with a Line Number. A Line Number is any integer from 0 to 64000.

Try typing in the following lines:

```
10 PRINT 2+3
20 PRINT 2-3
```

A sequence of Indirect Commands is called a "Program". Instead of executing indirect statements immediately, H&F BASIC saves Indirect Commands in the DRC memory (RAM). When you type RUN , H&F BASIC will execute the lowest numbered indirect statement that has been typed in, the the next highest, etc. for as many as were typed in.

Suppose we type RUN now:

```
RUN
```

H&F BASIC will type:

```
 5
-1

OK
```

In the example above, we typed in line 10 first and line 20 second. However, it makes no difference in what order you type in indirect statements. H&F BASIC always puts them into correct numerical order according to the line number.

If we want a listing of the complete program currently in memory, we type in LIST . Type this in:

```
LIST
```

H&F BASIC will reply with:

```
10 PRINT 2+3
20 PRINT 2-3
OK
```

Sometimes it is desirable to delete a line number of a program altogether. This is accomplished by typing the Line Number of the line we wish to delete, followed only by a carriage return.

Type in the following:

```
10
LIST
```

H&F BASIC will reply with:

```
20 PRINT 2-3
OK
```

We have now deleted line 10 from the program. There is no way to get it back. To insert a new line 10, just type in 10 followed by the statement we want H&F BASIC to execute.

Type in the following:

```
10 PRINT 2*3
```

```
LIST
```

H&F BASIC will reply with:

```
10 PRINT 2*3
20 PRINT 2-3
OK
```

There is an easier way to replace line 10 than deleting it and then inserting a new line. You can do this by just typing the new line 10 and hitting the carriage return. H&F BASIC throws away the old line 10 and replaces it with the new one.

Type in the following:

```
10 PRINT 3-3
LIST
```

H&F BASIC will reply with:

```
10 PRINT 3-3
20 PRINT 2-3
OK
```

It is not recommended that lines be numbered consecutively. It may become necessary to insert a new line betwen two existing lines. An increment of 10 between lines is generally sufficient.

If you want to erase the complete program currently stored in memory, type " NEW ". If you are finished running one program and are about to type in a new one, be sure to type " NEW " first. This should be done to prevent mixture of the old and new programs.

Type in the following:

```
NEW
```

H&F BASIC will reply with

```
OK
```

Now type in:

```
LIST
```

H&F BASIC will respond with:

```
OK
```

Often it is desirable to include text along with answers that are printed out, in order to explain the meaning of the numbers.

Type in the following:

```
PRINT "ONE THIRD IS EQUAL TO",1/3
```

H&F BASIC will reply with:

```
ONE THIRD IS EQUAL TO         .333333

OK
```

As explained earlier, including a " , " in a print statement causes it to space over to the next print field before the value following the comma is printed.

If we use a " ; " instead of a comma, the value next will be printed immediately following the previous value.

NOTE: Numbers are always printed with at least one trailing space. Any text to be printed is always to be enclosed in double quotes (").

Try the following examples:

```
A -  PRINT "ONE THIRD IS EQUAL TO";1/3
     ONE THIRD IS EQUAL TO .333333

     OK

B -  PRINT 1,2,3
      1            2            3

     OK

C -  PRINT 1;2;3
      1  2  3

     OK

D -  PRINT -1;2;-3
     -1  2 -3

     OK
```

Number Format

We will digress for a moment to explain the format of numbers in H&F BASIC. Numbers are stored internally to over six digits of accuracy. When a number is printed, only six digits are shown. Every number may also have an exponent (a power of ten scaling factor).

The largest number that may be represented in H&F BASIC is 1.70141E38, while the smallest positive number is 2.93874E-39.

When a number is printed, the following rules are used to determine the exact format:

1. If the number is negative, a minus sign (-) is printed. If the number is positive, a space is printed.

2. If the absolute value of the number is an integer in the range of 0 to 999999, it is printed as an integer.

3. If the absolute value of the number is greater than or equal to

.1 and less than or equal to 999999, it is printed in fixed point notation, with no exponent.

4. If the number does not fall under categories 2 or 3, scientific notation is used.

Scientific notation is formatted as follows: SX.XXXXXESTT . Each X is an integer between 0 and 9. The leading "S" is the sign of the number, a space for a positive one, and a " - " for a negative one. One non-zero digit is printed before the decimal point. This is followed by the decimal point and then the other five digits of the mantissa. An "E" is then printed (for exponent), followed by the sign of the exponent; then the two digits (TT) of the exponent itself. Leading zeroes are never printed; i.e. the digit before the decimal point is never zero. Also, trailing zeroes are never printed. If there is only one digit to print after all trailing zeroes are suppressed, no decimal point is printed. The exponent sign will be "+" for positive and "-" for negative. Two digits of the exponent are always printed; that is zeroes are not suppressed in the exponent field. The value of any number expressed thus is the number to the left of the "E" times 10 raised to the power of the number to the right of the "E".

No matter what format is used, a space is always printed following a number. H&F BASIC checks to see if the entire number will fit on the current line. If not, a carriage return/line feed is executed before printing the number.

The following are examples of various numbers and the output format H&F BASIC will place them into:

Number	Output Format
+1	1
-1	-1
6523	6523
-23.460	-23.46
1E20	1E+20
-12.3456E-7	-1.23456E-06
1.234567E-10	1.234567E-10
1000000	1E+06
999999	999999
.1	.1
.01	1E-02
.000123	1.23E-04

A number input from the terminal or a numeric constant used in a BASIC program may have as many digits as desired, up to the maximum length of a program line (111 characters). However, only the first 7 digits are significant, and the seventh digit is rounded up.

```
PRINT 1.23456789012345678
 1.234567

OK
```

PRINT USING

Often the default number formats listed above are not desirable. To limit the number of digits that are printed, or to align decimal points in a table of numbers, the PRINT USING command is available.

The format for a PRINT USING statement is

```
PRINT USING US$ N; M, P
```

where US$ is a string describing the print format (or picture for COBOL fans), and N, M and P are numeric variables that are to be printed using this format.

US$ takes the form of "#####.#####". Each "#" represents a digit before or after the decimal point. In this example, there would be five digits before the decimal point and five after. The sum of the number of digits before and after must not exceed ten, or a syntax error will result. US$ can be a predefined string variable, or can be a literal string included in the statement.

PRINT USING will fit the numeric variables in the print statement to the number format, if possible. One extra space will be allocated to the number prior to the leading digit for the sign. Leading zeroes will be converted to spaces. Trailing zeroes will be printed to fill out the format.

String variables or numerics that cannot be fit into the format will be printed without reformatting. Strings will be unchanged and numbers will be printed using the above listed number formats, if they cannot fit the PRINT USING format.

To demonstrate the use of PRINT USING, try the below listed programs. The first creates a table without using PRINT USING.

```
10 FOR N = 1 TO 10
20 PRINT 100*(RND(1)-RND(1))
30 NEXT
```

Line 20 prints random numbers between -100 and +100. A typical run might appear

```
RUN
-46.7245
 38.2826
-5.01143
-1.75031
-24.694
 73.1252
 58.9242
 4.09352
-19.0732
-67.6954
```

Notice the lack of decimal point alignment and the lack of "right fill". Try the below program.

```
10 FOR N=1 TO 10
20 PRINT USING "###.####" 100*(RND(1)-RND(1))
30 NEXT

RUN
 -11.6151
 -24.8873
  13.4291
```

```
-43.3089
-30.5268
 -3.4926
-34.0303
-50.7231
 81.6775
 14.9600
```

Finally, multiple USINGs can be used in the same line. The above program might be modified as below:

```
10 FOR N=1 TO 10
20 PRINT N; USING "###.#####", N; USING "###.##", N
30 NEXT

RUN
 1           1.00000   1.00
 2           2.00000   2.00
 3           3.00000   3.00
 4           4.00000   4.00
 5           5.00000   5.00
 6           6.00000   6.00
 7           7.00000   7.00
 8           8.00000   8.00
 9           9.00000   9.00
 10         10.00000  10.00
```

This is a simple implimentation of PRINT USING. It allows the simple formatting of logs and display screens. It does not allow for floating dollar signs and other functions that are available with more advanced PRINT USING, FORMAT or PICTURE statements in other languages.

PRINT, DISPLAY & PRINT#

There are several text I/O devices available on the DRC190. These are listed below:

0 - Standard Console (terminal plugged into J22)
1 - Standard Printer (plugged into terminal peripheral port)
2 - Direct Connect Modem (optional)
3 - RS232 port on direct connect modem (optional J23)

Use of the word DISPLAY sends output to the console. Use of the word PRINT sends output to the printer plugged into the console peripheral port. Use of the words PRINT # (# is a key word, so spaces are not required) allows direction of output to depend upon a specified number or a numeric variable. The number or numeric variable must be followed by a comma or semicolon to separate it from the remainder of the statement. Sample statements are given below:

```
PRINT#0,"HELLO"              Puts "HELLO" on console
PRINT#1,"HELLO"              Puts "HELLO" on console printer
PRINT#2,"HELLO"              Puts "HELLO" out on modem
PRINT#3,"HELLO"              Puts "HELLO" out on modem RS232 port
```

N=3:PRINT#N,"HELLO"	Sends "HELLO" to device 3
N=2:PRINT#N	No comma required. Sends CRLF to modem

INPUT

The following is an example of a program that reads a value from the terminal and uses that value to calculate and print a result:

```
10 INPUT R
20 PRINT 3.14159*R*R
RUN
10
 314.159

OK
```

Here's what's happening. When BASIC encounters the input statement, it waits for you to type in a number. When you do (in the example above, 10 was typed), execution continues with the next statement in the program after the variable R has been set (in this case to 10). In the above example, line 20 would now be executed. When the formula after the PRINT statement is evaluated, the value 10 is substituted for the variable R each time R appears in the formula. Therefore, the formula becomes 3.14159*10*10, or 314.159 .

If you haven't already guessed, what the program above acutally does is to calculate the area of a circle with radius R.

If we wanted to calculate the area of various circles, we could keep rerunning the program over each time for each successive circle. But, there's an easier way to do it simply by adding another line to the program as follows:

```
30 GOTO 10
RUN
10
 314.159
3
 28.2743
4.7
 69.3977
CTRL-C

BREAK IN LINE 10
```

Note that when a program is "control-C'd" or stops on an error, the DRC190 will beep the terminal once per second until a key is pressed. This insures that a program crash does not go unnoticed.

INPUT#

Just as with PRINT, input can be specified to come from a specific device. The format for this is INPUT#N, where N is the device number. The number must be followed by a comma or semicolon to separate it from the rest of the statement. Device 0 (the console) is the only device allowed to interrupt a program using control-C. Other devices will have a control-C changed to a space.

```
10 INPUT#2, A$
20 DISPLAY A$
```

The above program gets A$ from the direct connect modem and displays it on the console terminal. Note that line 10 will wait for a carriage return from the specified device. If this never arrives, the program will continue to wait. For this reason, it is suggested that broadcast control programs use INKEY$ instead of INPUT. This allows the device to be checked for a keystroke without hanging at that point in the program.

GOTO

By putting a " GOTO " statement on the end of our program, we have caused it to go back to line 10 after it prints each answer for the successive circles. This could have gone on indefinitely, but we decided to stop after calculating the area of three circles. This was accomplished by typing CTRL-C (control-C) by holding the key marked CONTROL while typing C. This will always stop program execution.

Numeric Variables

The letter "R" in the program we just used was termed a "variable". A variable name can be any alphabetic character and may be followed by any alphanumeric character. Any alphanumeric character after the first two are ignored. An alphanumeric character is any letter (A-Z) or any number (0-9).

Below are some examples of legal and illegal variable names:

LEGAL	ILLEGAL	
A	%	First character must be alphabetic
Z1	Z1A	Variable name too long
TP	TO	Names cannot be reserved words
COUNT	RGOTO	Names cannot contain reserved words

The words used as BASIC statements are reserved for this specific purpose. You cannot use these words as variable names or inside of any variable name. For instance, "FEND" would be illegal because "END" is a reserved word.

The following is a list of the reserved words in H&F BASIC:

ABS, ASC, AND, ATN, CHR$, CLEAR, CLRSTK, CONT, COS, DATA, DATE, DATE$, DAY, DAY$, DEF, DIM, DIR, DISPLAY, EEPROM, END, ERR, EXP, FN, FOR, FRE, GOSUB, GOTO, IF, INKEY$, INPUT, INT, LEFT$, LET, LEN, LET, LINE, LIST, LOAD, LOG, LOWER, LOWER$, MAXLINE, MDMSPD, MEMTEST, METER, METER$, MODEMTST, MID$, NEW, NEXT, NOT, ON, OR, PEEK, POKE, POS, PRESET, PRINT, RAISE, RAISE$, READ, REM, RESET, RESTORE, RETURN, RIGHT$, RND, RUN, SAVE, SGN, SET, SIN, SPC(, SQR, STATUS, STATCLR, STEP, STOP, STR$, SWAP, SYSTAT, TAB(, TAN, THEN, TO, TIME, TIME$, TROFF, TRON, USING, USR, VAL, WAIT, #, +, -, *, /, ², >, <, =

Besides having values assigned to variables with an input statement, you can also set the value of a variable with a LET or assignment statement.

Try the following examples:

```
A=5
```

```
OK

PRINT A,A*2
5    10

OK

LET Z=7

OK

PRINTZ, Z-A

7    2

OK
```

As can be seen from the examples, the "LET" is optional in an assignment statement.

H&F BASIC remembers the values that have been assigned to variables using this type of statement. This "remembering" process uses space in the DRC memory to store the data.

The values of the variables are thrown away and the space in memory used to store them is released when one of four things occur:

1. A new line is typed into the program or an old line is deleted.

2. A CLEAR command is typed in.

3. A RUN command is typed.

4. NEW is typed.

Another important fact is that if a variable is encountered in a formula before it is assigned a value, it is automatically assigned the value zero. Zero is then substituted as the value of the variable in the particular formula. Try the example below:

```
PRINT Q, Q+2, Q*2
 0   2    0

OK
```

Another statement is the REM statement. REM is short for remark. This statement is used to insert comments or notes into a program. When H&F BASIC encounters a REM statement the rest of the line is ignored.

This serves mainly as an aid for the programmer himself, and serves no useful function as far as the operation of the program in solving a particular problem.

IF-THEN

Suppose we want to write a program to check if a number is zero or not. With the statements we've gone over so far, this could not be done. What is needed is a statement which can be used to conditionally branch to another statement. The "IF-THEN" statement does just that.

Try typing in the following program: (remember, type NEW first).

```
10 INPUT B
20 IF B=0 THEN 50
30 PRINT "NON-ZERO"
40 GOTO 10
50 PRINT "ZERO"
60 GOTO 10
```

When this program is typed into the DRC and run, it will ask for a value of B. Type any value you wish. The DRC will come to the "IF" statement. Between the "IF" and the "THEN" portion of the statement there are two expressions separated by a relation.

A relation is one of the following six symbols:

RELATION	MEANING
=	Equal to
>	Greater than
<	Less than
<>	Not equal to
<=	Less than or equal to
=>	Greater than or equal to

The IF statement is either true or false, depending upon whether the two expressions satisfy the relation or not. For example, in the program we just did, if 0 was typed in for B, the IF statement would be true because 0=0. In this case, since the number after the THEN is 50, execution of the program would continue at line 50. Therefore, "ZERO" would be printed and then the program would jump back to line 10 (because of the GOTO statement in line 60).

Suppose a 1 was typed in for B. Since 1=0 is false, the IF statement would be false and the program would continue execution with the next line. Therefore, "NON-ZERO" would be printed and the GOTO in line 40 would send the program back to line 10.

H&F BASIC uses the number -1 to represent TRUE and 0 to represent FALSE. Actually, any non-zero number will be interpretted as TRUE. In the above program, the expression B=0 is replaced by a -1 if B does indeed equal zero. Otherwise it is replaced by a zero. The -1 again represents TRUE and the 0 represents FALSE.

This idea can be used in other than IF-THEN statements. For example, the statement PRINT 0=1 will print 0 since 0=1 is FALSE. The statement PRINT 1E3=1E3 will print -1, since 1E3=1E3 is TRUE.

Now try the following program for comparing two numbers:

```
10 INPUT A, B
20 IF A<=B THEN 50
30 PRINT "A IS BIGGER"
40 GOTO 10
50 IF A<B THEN 80
60 PRINT "THEY ARE THE SAME"
70 GOTO 10
80 PRINT "B IS BIGGER"
90 GOTO 10
```

When the program is run, line 10 will input two numbers from the terminal. At line 20, if A is greater than B, A<=B will be false. This will cause the next statement to be executed, printing " A IS BIGGER" and then line 40 sends the computer back to line 10 to begin again.

At line 20, if A has the same value as B, A<=B is true so we go to line 50. At line 50, since A has the same value as B, A<B is false; therefore, we go to the following statement and print "THEY ARE THE SAME". Then line 70 sends us back to the beginning again.

At line 20, if A is smaller than B, A<=B is true so we go to line 50. At line 50, A<B will be true so we then go to line 80. "B IS BIGGER" is then printed and again we go back to the beginning.

Try running the program several times. It may make it easier to understand if you try writing your own programs at this time using the IF-THEN statement. Actually trying programs of your own is the quickest and easiest way to understand how H&F BASIC works. Remember, to stop these programs, just type CTRL-C (Control-C).

One advantage of computers is their ability to perform repetitive tasks. Let's take a closer look and see how this works.

Suppose we want a table of square roots from 1 to 10. The H&F BASIC function for square root is "SQR"; the form being SQR(X), X being the number you wish the square root calculated from (the function "argument"). We could write the program as follows:

```
10 PRINT 1, SQR(1)
20 PRINT 2, SQR(2)
30 PRINT 3, SQR(3)
40 PRINT 4, SQR(4)
50 PRINT 5, SQR(5)
60 PRINT 6, SQR(6)
70 PRINT 7, SQR(7)
80 PRINT 8, SQR(8)
90 PRINT 9, SQR(9)
100 PRINT 10, SQR(10)
```

This program will do the job; however, it is terribly inefficient. We can improve the program tremendously by using the IF statement just introduced as follows:

```
10 N=1
20 PRINT N, SQR(N)
30 N=N+1
40 IF N<=10 THEN 20
```

When this program is run, its output will look exactly like that of the 10 statement program above. Let's look at how it works.

At line 10 we have a LET statement which sets the value of the variable N at 1. At line 20 we print N and the square root of N using its current value. It this becomes 20 PRINT 1, SQR(1), and this calculation is printed out.

At line 30 we use what will appear to be a rather unusual LET statement. Mathematically, the statement N=N+1 is nonsense. However, the important thing to remember is that in a LET statement, the symbol " = " does not signify equality. In this case " = " means "to be replaced with". All the statement does is to take the current value of N and add 1 to it. Thus, after the first time through line 30, N becomes 2.

At line 40, since N now equals 2, N<=10 is true so the THEN portion branches us back to line 20, with N now at a value of 2.

The overall result is that lines 20 through 40 are repeated, each time adding 1 to the value of N. When N finally equals 10 at line 20, the next line will increment it to 11. This results in a false statement at line 40, and since there are no further statements to the program, it stops.

FOR-NEXT

This technique is referred to as "looping" or "iteration". Since it is used quite extensively in programming, there are special H&F BASIC statements for using it. We can show these with the following program.

```
10 FOR N=1 TO 10
20 PRINT N, SQR(N)
30 NEXT N
```

The output of the program listed above will be exactly the same as the previous two programs.

At line 10, N is set equal to 1. Line 20 causes the value of N and the square root of N to be printed. At line 30, we see a new type of statement. The "NEXT N" statement causes one to be added to N, and then if N<=10 we go back to the statement following the "FOR" statement. Note that in H&F BASIC, the N in "NEXT N" is optional. By simply using NEXT for line 30, the DRC looks for the previous FOR statement and indexes the variable it specifies. The overall operation then is the same as with the previous program.

Suppose we want to print a table of square roots from 10 to 20, only counting by two's. The following program would perform this task:

```
10 N=10
20 PRINT N, SQR(N)
30 N=N+2
40 IF N<=20 THEN 20
```

Note the similar structure between this program and the previous one for printing square roots for numbers 1 to 10. This program can also be written using the FOR-NEXT loop just introduced.

```
10 FOR N=10 TO 20 STEP 2
20 PRINT N, SQR(N)
30 NEXT
```

Notice that the only major difference between this program and the previous one using the FOR-NEXT loop is the addition of "STEP 2".

This tells H&F BASIC to add 2 to N each time, instead of 1 as in the previous program. If no "STEP" is given in a "FOR" statement, H&F BASIC assumes that one is to be added each time. The "STEP" can be followed by any expression.

Suppose we want to count backwards from 10 to 1. A program for doing this would be as follows:

```
10 I=10
20 PRINT I
30 I=I-1
40 IF I>=1 THEN 20
```

Notice that we are now checking to see that I is greater than or equal to the final value. The reason is that we are now counting by a negative number. In the previous examples, it was the opposite, so we were checking for a variable less than or equal to the final value.

The "STEP" statement previously shown can also be used with negative numbers to accomplish the same purpose. This can be done using the same format as in the other program, as follows:

```
10 FOR I=10 TO 1 STEP -1
20 PRINT I
30 NEXT
```

"FOR" loops can also be "nested". An example of this procedure follows:

```
10 FOR I=1 TO 5
20 FOR J=1 TO 3
30 PRINT I, J
40 NEXT: REM THIS NEXT LOOPS BACK TO J
50 NEXT: REM THIS NEXT LOOPS BACK TO I
```

Notice that the "NEXT J" comes before the "NEXT I". This is because the J-loop is inside of the I-loop. By the way, the colon (:) allows us to put more than one statement on a line. In this case, the second statement on the line was a REMark, which the DRC ignores. REMarks aid the programmer in remembering how the program is supposed to work. The following program is incorrect; run it and see what happens.

```
10 FOR I=1 TO 5
20 FOR J=1 TO 3
30 PRINT I, J
40 NEXT I
50 NEXT J
```

It does not work because when the "NEXT I" is encountered, all knowledge of the J-loop is lost. This happens because the J-loop is "inside" of the I-loop.

Matrices

It is often convenient to be able to select any element in a table of numbers. H&F BASIC allows this to be done through the use of matrices.

A matrix is a table of numbers. The name of this table, called the matrix name, is any legal variable name, "A" for example. The matrix name "A" is distinct and separate from the simple variable "A", and you can use both in the same program.

To select an element of the table, we subscript "A": that is to select the I'th element, we enclose I in parenthesis "(I)" and then follow "A" by this subscript. Therefore, "A(I)" is the I'th element in the matrix "A".

In this section of the manual we will be concerned with one-dimensional matrices only.

"A(I)" is only one element of matrix A, and H&F BASIC must be told how much space to allocate for the entire matrix.

This is done with a "DIM" statement, using the format "DIM A(15)". In this case, we have reserved spaces for the matrix index "I" to go from 0 to 15. Matrix subscripts always start at 0; therefore, in the above example, we have allowed 16 numbers in matrix A.

If "A(I)" is used in a program before it has been dimensioned, H&F BASIC reserves space for 11 elements (0 through 10).

As an example of how matrices are used, try the following program to sort a list of 8 numbers with you picking the numbers to be sorted.

```
10 DIM A(8)
20 FOR I=1 TO 8
30 INPUT A(I)
50 NEXT
70 F=0
80 FOR I=1 TO 7
90 IF A(I)<=A(I+1) THEN 140
100 T=A(I)
110 A(I)=A(I+1)
120 A(I+1)=T
130 F=1
140 NEXT
150 IF F=1 THEN 70
160 FOR I=1 TO 8
170 PRINT A(I),
180 NEXT
```

When line 10 is executed, H&F BASIC sets aside space for 9 numerical values, A(0) through A(8). Lines 20 through 50 get the unsorted list from the user. The sorting itself is done by going through the list of numbers and upon finding any two that are not in order, we switch them. "F" is used to indicate if any switches were done. If any were done, line 150 tells H&F BASIC to go back and check some more.

If we did not switch any numbers, or after they are all in order, lines 160 through 180 will print the sorted list. Note that a subscript can be any expression.

GOSUB & RETURN

Another useful pair of statements are "GOSUB" and "RETURN". If you have a program that performs the same action in several different places, you can duplicate the same statements for the action in each place within the program.

The "GOSUB-RETURN" statements can be used to avoid this duplication. When a "GOSUB" is encountered, H&F BASIC branches to the line whose number follows the "GOSUB". However, H&F BASIC remembers where it was in the program before it branched. When the "RETURN" statement is encountered, H&F BASIC goes back to the first statement following the last "GOSUB" that was executed. Observe the following program.

```
10 PRINT "WHAT IS THE NUMBER";
30 GOSUB 100
40 T=N
50 PRINT "WHAT IS THE SECOND NUMBER";
70 GOSUB 100
80 PRINT "THE SUM OF THE TWO NUMBERS IS", T+N
90 STOP
```

```
100 INPUT N
110 IF N = INT(N) THEN 140
120 PRINT "SORRY, NUMBER MUST BE AN INTEGER. TRY AGAIN"
130 GOTO 100
140 RETURN
```

What this program does is ask for two numbers which must be integers, and then prints the sum of the two. The subroutine in this program is lines 100 to 130. The subroutine asks for a number, and if it is not an integer, asks for a number again. It will continue to ask until and integer value is typed in.

The main program prints "WHAT IS THE NUMBER". and then calls the subroutine to get the value of the number into N. When the subroutine returns (to line 40), the value input is saved in the variable T. This is done so that when the subroutine is called a second time, the value of the first number will not be lost.

"WHAT IS THE SECOND NUMBER" is then printed, and the second value is entered when the subroutine is called again.

When the subroutine returns the second time, "THE SUM OF THE TWO NUMBERS IS" is printed, followed by the value of their sum. T contains the value of the first number that was entered and N contains the value of the second number.

The next statement is a "STOP" statement. This causes the program to stop execution at line 90. If the "STOP" statement was not included in the program, we would "fall into" the subroutine at line 100. This is undesirable because we would be asked to input another number. If we did, the subroutine would try to return; and since there was no "GOSUB" which called the subroutine, an RG error would occur. Each "GOSUB" executed in a program should have a matching "RETURN" executed later, and the opposite applies, i.e. a "RETURN" should be encountered only if it is part of a subroutine which has been called by a "GOSUB".

Either "STOP" or "END" can be used to separate a program from its subroutines. "STOP" will print a message saying at what line "STOP" was encountered.

In broadcast or other process control applications, we don't really want the program to stop. Instead, we want it to forever repeat its control program. In the above program, this could be accomplished by substituting GOTO 10 for line 130. The program would continue to run (asking for numbers to be added) until a Control-C was typed in. So, we can use "STOP", "END", or "GOTO" to separate the main program from subroutines.

DATA & RESTORE

Suppose you had to enter numbers to your program that didn't change each time the program was run, but you would like it to be easy to change them if necessary. H&F BASIC contains special statements for this purpose, called the "READ" and "DATA" statements.

Consider the following program:

```
10 PRINT "GUESS A NUMBER";
20 INPUT G
30 READ D
40 IF D=-999999 THEN 90
50 IF D<>G THEN 30
60 PRINT "YOU ARE CORRECT"
70 END
```

```
90 PRINT "BAD GUESS, TRY AGAIN."
95 RESTORE
100 GOTO 10
110 DATA 1,393,-39,28,391,-8,0,3.14,90
120 DATA 89,5,10,15,-34,-999999
```

This is what happens when this program is run. When the "READ" statement is encountered, the effect is the same as an INPUT statement. But, instead of getting a number from the terminal, a number is read from the "DATA" statements.

The first time a number is needed for a READ, the first number in the first DATA statement is returned. The second time one is needed, the second number in the first DATA statement is returned. When the entire contents of the first DATA statement have been read in this manner, the second DATA statement will be used. DATA is always read sequentially in this manner, and there may be any number of DATA statements in your program.

The purpose of this program is to play a little game in which you try to guess one of the numbers contained in the DATA statements. For each guess that is typed in, we read through all of the numbers in the DATA statements until we find one that matches the guess.

If more values are read than there are numbers in the DATA statements, an out of data (OD) error occurs. That is why in line 40 we check to see if -999999 was read. This is not one of the numbers to be matched, but is used as a flag to indicate that all of the data (possible correct guesses) has been read. Therefore, if -999999 was read, we know that the guess given was incorrect.

Before going back to line 10 for another guess, we need to make the READ's begin with the first piece of data again. This is the function of the "RESTORE". After the RESTORE is encountered, the next piece of data read will be the first piece in the first DATA statement again.

DATA statements may be placed anywhere within the program. Only READ statements make use of the DATA statements in a program, and any other time they are encountered during program execution they will be ignored.

Strings

A list of characters is referred to as a "String". H&F, DRC, and THIS IS A TEST are all strings. Like numeric variables, string variables can be assigned specific values. String variables are distinguished from numeric variables by a "$" after the variable name.

For example, try the following:

```
A$="Hallikainen & Friends "

OK
PRINT A$
Hallikainen & Friends

OK
```

In this example, we set the string variable A$ to the string value "Hallikainen & Friends". Note that we also enclosed the character string to be assigned to A$ in quotes.

LEN

Now that we have set A$ to a string value, we can find out what the length of this value is (the number of characters it contains). We do this as follows:

```
PRINT LEN(A$), LEN("HELLO")
 21  5

OK
```

The "LEN" function returns a integer equal to the number of characters in a string.

The number of characters in a string expression may range from 0 to 255. A string which contains 0 characters is called the "NULL" string. Before a string variable is set to a value in the program, it is initialized to the null string. Printing a null string on the terminal will cause no characters to be printed, and the print head (or cursor) will not be advanced to the next column. Try the following:

```
PRINT LEN(Q$);Q$;3
 0  3

OK
```

Another way to create the null string is: Q$="" .

Setting a string variable to the null string can be used to free up the string space used by non-null string variables.

LEFT$

Often it is desirable to access parts of a string and manipulate them. Now that we have set A$ to "Hallikainen & Friends", we might want to print out only the first six characters of A$. We would do so like this:

```
PRINT LEFT$(A$,6)
Hallik

OK
```

"LEFT$" (pronounced "LEFT-STRING") is a string function which returns a string composed of the leftmost N characters of its string argument. Here's another example:

```
FOR N=1 TO LEN(A$):PRINT LEFT$(A$,N):NEXT
H
Ha
Hal
Hall
Halli
Hallik
Hallika
Hallikai
```

```
Hallikain
Hallikaine
Hallikainen
Hallikainen
Hallikainen &
Hallikainen &
Hallikainen & F
Hallikainen & Fr
Hallikainen & Fri
Hallikainen & Frie
Hallikainen & Frien
Hallikainen & Friend
Hallikainen & Friends

OK
```

Since A$ has 21 characters this loop will be executed with N=1, 2, 3, . . . printed. The second time through the first two characters will be printed.

RIGHT$

There is another string function called "RIGHT$" which returns the right N characters from a string expression. Try substituting RIGHT$ for LEFT$ in the previous example and see what happens.

MID$

There is also a string function which allows us to take characters from the middle of a string. Try the following:

```
FOR N=1 TO LEN(A$):PRINT MID$(A$,N):NEXT
Hallikainen & Friends
allikainen & Friends
llikainen & Friends
likainen & Friends
ikainen & Friends
kainen & Friends
ainen & Friends
inen & Friends
nen & Friends
en & Friends
n & Friends
 & Friends
& Friends
 Friends
Friends
riends
iends
ends
nds
ds
s
```

```
OK
```

MID$ returns a string starting at the Nth posistion of A$ to the end (last character) of A$. The first position of the string is position 1 and the last possible postion of a string is position 255.

Very often it is desirable to extract only the Nth character from a string. This can be done by calling MID$ with three arguments. The third argument specifies the number of characters to return.

For example:

```
FOR N=1 TO LEN A$:PRINT MID$(A$,N,1),MID$(A$,N,2):NEXT
H    Ha
a    al
l    ll
l    li
i    ik
k    ka
a    ai
i    in
n    ne
e    en
n    n
      &
&    &
      F
F    Fr
r    ri
i    ie
e    en
n    nd
d    ds
s    s

OK
```

Concatenation

Strings may also be concatenated (put or joined together) through the use of the "+" operator. Try the following:

```
B$="WONDERFUL"+" "+A$

OK
PRINT B$
WONDERFUL Hallikainen & Friends

OK
```

Concatenation is especially useful if you wish to take a string apart and then put it back together with slight modifications. For instance:

```
C$=LEFT$(B$,4)+"-"+MID$(B$,6,6)+"-"+RIGHT$(B$,4)
```

```
OK
PRINT C$
WOND-RFUL H-ends

OK
```

VAL & STR$

Sometimes it is desirable to convert a number to its string representation and vice-versa. "VAL" and "STR$" perform these functions. Try the following:

```
STRING$="567.8"

OK
PRINT VAL(STRING$)
 567.8

OK
STRING$=STR$(3.1415)

OK
PRINT STRING$,LEFT$(STRING$,5)
3.1415    3.14

OK
```

"STR$" can be used to peform formatted I/O on numbers. You can convert a number to a string and then use LEFT$, RIGHT$, MID$, and concatenation to reformat the number as desired. Also see "USING".

"STR$" can also be used to conveniently find out how many print columns a number will take. For example:

```
PRINT LEN(STR$(3.157))
6

OK
```

If you have an application where a user is typing in a question such as "WHAT IS THE VOLUME OF A CYLINDER OF RADIUS 5.36 FEET, HEIGHT 5.1 FEET?" you can use "VAL" to extract the numeric values 5.36 and 5.1 from the question.

The following program sorts a list of string data and prints out a sorted list. This program is very similar to one given earlier for sorting a numeric list.

```
100 DIM A$(15):REM ALLOCATE SPACE FOR STRING MATRIX
110 FOR I=1 TO 15:READ A$(I):NEXT:REM READ IN STRINGS
120  F=0:I=1:REM EXCHANGE FLAG = 0 & SUBSRIPT FLAG = 1
130 IF A$(I)<=A$(I+1) THEN 180:REM IN ORDER, NO CHANGE
140  T$=A$(I+1):REM SAVE A$(I+1)
150  A$(I+1)=A$(I): REM EXCHANGE
160 A$(I)=T$
170 F=1:REM FLAG THAT WE EXCHANGED ELEMENTS
180 I=I+1:IF I<15 GOTO 130
```

```
185 REM ONCE WE HAVE MADE A PASS THRU ALL ELEMENTS,
186 REM CHECK TO SEE IF WE EXCHANGED ANY.  IF NOT,
187 REM WE ARE DONE SORTING
190 IF F THEN 120:REM EQUIVALENT TO IF F<>0 THEN 120
200 FOR I=1 TO 15:PRINT A$(I):NEXT:REM PRINT LIST
210 REM STRING DATA FOLLOWS
220 DATA APPLE, DOG, CAT, BITS, BYTES, RANDOM
230 DATA MONDAY, "***ANSWER***", " FOO"
240 DATA COMPUTER,  FOO, ELP, MILWAUKEE, SEATTLE
250 DATA ALBUQUERQUE
```

METER

All of the previous material has dealt with Basic as might be used in a business or scientific application. To log the operation of your broadcast station, you'd have to type the readings in using INPUT statements. This is not very efficient.

METER is a function that can be called with one or two arguments. It is generally called with two arguments in the form

```
METER(S,C)
```

where S is the site number and C is the channel number that we wish to get a reading from. For example,

```
PRINT METER(0,0)
```

will print the meter reading at site 0, channel 0. Site and channel must be between 0 and 99 or a function call error will occur.

There are several single argument methods of calling METER. These are useful in that a single array (matrix) element can be used to specify the metering data to be acquired. One number can specify both site and channel. In addition, a few negative numbers can be used to specify other common logging data such as time, day, and date.

METER(-1) returns the time as a 6 digit integer in HHMMSS format (see the section on time). METER(-2) returns the day of week as a number between 0 and 6 where 0 is Sunday. METER(-3) returns the date as a 6 digit integer in YYMMDD format.

If a single positive integer is used as the argument to METER, the argument assumes the following meaning:

```
(512*SITE) + (256*RAISE) + (128*LOWER) + CHANNEL
```

For example, METER(513) is equivalent to METER(1,1).

RAISE or LOWER take the value of 0 normally. They take the value of 1 if a RAISE or LOWER during the sample is required (sometimes required to drive an antenna monitor).

Meter$

METER$ (meter-string) returns the meter reading in string form with the label and units programmed during system calibration. This allows simple programs to put the station parameters with labels and units on a CRT. A sample program using METER$ is shown in the Sample Basic Programs section of this manual.

METER$ can be called with the same arguments as METER (described above). If, for example, we call METER$(-3), we'll get a string with the date in English (September 11, 1985).

RAISE

RAISE can be used as a statement to generate a RAISE control pulse at a specified site and channel. The arguments passed to RAISE are of the same form as those used in METER. This allows the two argument form, or the single argument form.

```
RAISE(1,0)
```

This statement might be used to turn on the transmitter filaments.

RAISE can also be used as a function. In this case, it will return the meter reading that occurs if the selected site and channel are read while a RAISE pulse is applied. A sample program statement might be:

```
PRINT RAISE(1,0)
```

The RAISE function can also be used in string form, returning the METER$ that results when a RAISE is sent to the designated site and channel. A sample program statement might appear:

```
PRINT RAISE$(1,0)
```

LOWER

LOWER can be used as a statement or function in the same manner as RAISE.

TIME

TIME can be used as a statement or a function. It almost appears to be a system variable, but the setting of time must be done through what looks like a LET statement. TIME cannot be set with an INPUT statement or a DATA statement. A typical routine to set time would be:

```
10 DISPLAY "Is it now "; TIME$
20 INPUT YN$
30 IF YN$="Y" OR YN$="y" THEN RETURN
40 DISPLAY "Type the current time in HHMMSS format.  "
50 INPUT A
60 TIME=A
70 GOTO 10
```

Note that the time was set using a LET statement in line 60. INPUT TIME will not work!

TIME is to be a 6 digit integer representation of the time in HHMMSS format. 6:01:02 pm would be 180102.

TIME can be printed or used in calculations. As such, it appears as a system variable, but is actually implimented as a function call with no arguments required.

TIME is most often used to determine if something needs to be done based on a time schedule. If you need to switch pattern at sunset and SS(9) (sunset array entry number 9, for the ninth month) holds the sunset time for this month, the following statement might be used:

```
IF (TIME=>SS(9)) AND (METER(1,9)>1000) THEN LOWER(1,9)
```

This program line checks to see if it's past sunset and if we are still in day pattern (indicated by METER(1,9) being greater than 1000). If so, a LOWER statement is sent to the pattern select channel. If this line is included in a program loop, the pattern will only be changed once, since the line checks to see if it's already been done. Similar statements could be used to check the pattern between midnight and sunrise, and between sunrise and sunset.

When the time is printed or displayed, TIME$ is normally used. The below program line demonstrates the difference between TIME and TIME$.

```
DISPLAY TIME, TIME$
 140524    2:05:24 PM
```

Note that TIME$ cannot be set in a LET statement. TIME$ is determined based on TIME, which can be set using a LET statement.

DAY, DAY$, DATE, DATE$

These statements and functions act the same way as TIME and TIME$.

DAY is an integer between 0 and 6, inclusive, that indicates the current day of the week. It can be set in a LET statement or evaluated in a numeric formula. It is often used when program decisions are based on the day of the week. A typical application of DAY would be in a routine that tries to determine if we should currently be on advanced (daylight savings) or non-advanced (standard) time.

DAY$ returns the day of the week in English. The below program line demonstrates this:

```
DISPLAY DAY, DAY$
 2         Tuesday
```

DATE is a 6 digit integer representing the date in YYMMDD format. It can be set using a LET statement and read using DATE or DATE$. A sample program line is shown belos:

```
DISPLAY DATE, DATE$
 850911    September 11, 1985
```

The DATE can be taken apart to get the day of the month, the month, and the year using the below statements.

```
10 YR=INT(DATE/1E4)
20 MO=INT((DATE-(YR*1E4))/100)
30 DT=DATE-(YR*1E4)-(MO*100)
40 DISPLAY "Year = "; YR
50 DISPLAY "Month= "; MO
```

```
60 DISPLAY "Day  = "; DT
```

This is handy to index into an array holding the sunrise and sunset times for the year.

Program Storage

Programs reside in system RAM once they are typed in or are executing. These programs will be lost during a power failure, or during an extended power failure if battery backup is included. Short programs (about 1 Kbyte) can be stored in space left in the EEPROM (Electrically Erasable Programmable Read Only Memory). Once a program has been typed into the DRC190, executing the following statement will save the program in EEPROM, if there is sufficient space available.

```
SAVE EEPROM
```

The program in EEPROM can be loaded using the following statement:

```
LOAD EEPROM
```

In addition, a program in EEPROM will be loaded and run on a system reset. This program might be a simple logging program, or could be a program to load another program from disc. A sample of this is shown below:

```
10 DISPLAY "Loading Logging Program"
20 LOAD "LOGGER PROGRAM",8
```

In addition, if there is additional EEPROM in the system (typically loaded into high memory on the RAM board), larger programs can be saved in EEPROM by specifying the address. A program residing in RAM can be saved in EEPROM on the RAM board using the statement:

```
SAVE EEPROM 24576
```

The program can be loaded in a similar manner:

```
LOAD EEPROM 24576
```

If a large program is saved in EEPROM on the RAM board, the boot program in the processor board EEPROM can load the other program. The boot program might be:

```
10 LOAD EEPROM 24576
```

This program would be loaded and executed on power up. The execution of this program loads and executes the other program from EEPROM on the RAM board.

Disc Program Storage

Programs may be stored on a Commodore 1541 disc drive plugged in to the optional disc interface on the DRC190. Program saving and loading procedures are similar to that used on Commodore computers, since the Commodore operating system (residing in the disc drive) is used.

The following statement loads a program from drive 8 (usually the first drive in a system):

```
LOAD"PROGRAM NAME",8
```

The following statement saves a program residing in the DRC190 to disc drive number 8.

```
SAVE"PROGRAM NAME",8
```

Note that the above statement will not replace an existing file with the same name. To replace a file with the same name, use the below statement:

```
SAVE"@0:PROGRAM NAME",8
```

Finally, you may get a directory of what is on a disc by typing the following statement:

```
DIR N
```

where N is the drive number. If N is omitted, drive 8 is assumed.

There are various other functions and statements available in H&F Basic. These are covered in the following section.

PROGRAMMING REFERENCE MATERIAL

Descriptions and examples of key word usage are listed below. These are listed in alphabetical order.

Words are broken into statements and functions.

A statement instructs the processor to take some action. Examples are NEW, LIST, RAISE, LOWER, CLEAR, etc. Most statements can be used within a program (a sequence of statements with line numbers), or as a command. A command is usually given after H&F BASIC has typed OK. This is called the "Command Level". Commands may be used as program statements. Certain commands, such as LIST and NEW will terminate program execution when they finish.

Functions act upon zero or more arguments and return a result to be used in an expression. For example, PRINT N is a statement that prints the numeric expression (a somewhat simple one) N. PRINT SQR(N) is a statement that prints the numeric expression SQR(N). SQR is a function that finds the square root of its argument (N) and returns it to the expression that called it.

NOTE: In the following descriptions, an argument of V or W denotes a numeric variable, X denotes a numeric expression, X$ denotes a string expression and an I or J denotes an expression that is truncated to an integer before the statement is executed. Truncation means that any fractional part of the number is lost, e.g. 3.9 becomes 3, 4.01 becomes 4.

An expression is a series of variables, operators, function calls and constants which after the operations and function calls are performed using the precedence rules, evaluates to a numeric or string value.

A constant is either a number (3.14) or a string literal ("FOO").

A string may be from 0 to 255 characters in length. All string variables end in a dollar sign ($); for example, A$, B9$, K$, HELLO$.

String matrices may be dimensioned exactly like numeric matrices. For instance, DIM A$(10,10) creates a string of 121 elements, eleven rows by eleven columns (rows 0 to 10 and columns 0 to 10). Each string matrix element is a complete string, which can be up to 255 characters in length.

Statements and functions available on the DRC190 are listed below.

NAME	EXAMPLE	PURPOSE/USE
ABS(X)	120 PRINT ABS(X)	Gives the absolute value of the expression X. ABS returns X if X>=0, -X otherwise.
AND	100 IF (A<B) AND (B<C) THEN 500	AND can be used to indicate a logical or bitwise AND function. In the 100 example, it is indicating a logical AND. The two relational expressions return a -1 if they are true, and a 0 if they are false. The -1 is represented as a 16 bit integer (1111111111111111), which when ANDed with another -1, yields another -1, which indicates TRUE. TRUE AND TRUE is TRUE.
	200 PRINT 16 AND 31	In line 200, the two integers are changed to 16 bit binary form, ANDed, and the result returned. In this case, 16 AND 31 is 0000 0000 0001 0000 AND 0000 0000 0001 1111, giving a result of 0000 0000 0001 0000 or 16.
ASC(X$)	300 PRINT ASC(X$)	Returns the ASCII numeric value of the first

		character of the string expression X$. An FC error will occur if X$ is the null string.
ASC(X$,N)		Returns the ASCII numeric value of the Nth character of string X$.
ATN(X)	210 PRINT ATN(X)	Gives the arctangent of the argument X. The result is returned in radians and ranges from -PI/2 to PI/2.
CHR$(I)	275 PRINT CHR$(I);	Returns a one character string whose single character is the ASCII equivalent of the value of the argument (I) which must be >=0 and <=255.
CLEAR	CLEAR 500	Clears all variables, resets FOR-NEXT, GOSUB, and DATA states. Sets aside 500 characters of string storage.
	CLEAR 500,1E3	Clears all variables, resets FOR-NEXT, GOSUB, and DATA states. Sets aside 500 characters of string storage. Tells BASIC to use only 1000 (1E3) bytes of RAM, if available. This command allows the user to set aside high RAM for use by other than the BASIC program. This command may be executed from within a program, typically in an initialization routine. Remember, however, that CLEAR clears the GOSUB stack, so a RETURN cannot be used at the end of a routine that includes a CLEAR.
CLRSTK	20 CLRSTK	Clears the subroutine and FOR-NEXT stack. Used to abort a subroutine. Typically followed by a GOTO statement returning program control to the main level of the program.
CONT	CONT	Continues execution of program after it was interrupted by CTRL-C. Variables can be examined and changed in command mode, but no program changes can be made or command level errors encountered, or continue cannot be done (CN error).
COS(X)	200 PRINT COS(X)	Gives the cosine of the expression X. X is interpreted as being in radians.
DATA	10 DATA 1,3,-1E3,.04	Specifies data, read from left to right. Information appears in data statements in the same order it will be read in the program.
	20 DATA " FOO",ZOO	Strings may be read from DATA statements. If you want the string to contain leading spaces (blanks), colons (:) or commas (,), you must enclose the string in double quotes ("). It is impossible to have double quotes within string data or a string literal. (""HELP"") is not legal. See READ.
DATE	500 PRINT DATE	DATE is an integer variable holding the date in YYMMDD format. January 23, 1982 is stored as 820123. By comparing DATE with a previously stored value, date dependent functions can be performed. For example, INT(DATE/100)-100*INT(DATE/10000) represents the month. This can be used to select pattern change times. DATE is updated by the internal clock.
	600 DATE=820123	Set DATE using a LET statement. DATE cannot be

		set using an INPUT or READ statement. If this is desired, INPUT or READ to a temporary variable and then set DATE using a LET statement. Setting DATE also sets DATE$.
DATE$	800 PRINT DATE$	DATE$ is a string variable that can be read, but not written to (DATE$=Tuesday is NOT legal). As such (and it is in fact coded this way in the firmware), it may be thought of as a zero argument function. The example prints date (set up above) in English. Date$ is set up using a LET DATE= statement. DATE$ cannot be set other than through DATE.
DAY	100 IF DAY=0 THEN 300 200 DAY=3	DAY is an integer variable holding the day of week. 0 represents Sunday, 6 represents Saturday. DAY is normally used in IF statements. DAY can be initialized with a LET statement. It cannot be initialized with an INPUT or DATA statement. If this is desired, use INPUT or DATA to set a temporary variable, then use a LET statement to set DAY to the temporary variable. DAY is updated by the internal clock.
DAY$	400 PRINT DAY$	Prints day (set up above) in English. Since DAY$ follows DAY, it cannot be set using a string assignment statement. Set the numeric DAY instead.
DEF	100 DEF FNA(V)=V/B+C	The user can define functions like the built in functions (SQR, SGN, ABS, etc.) through the use of the DEF statement. The name of the function is "FN" followed by any legal variable name, for example: FNX, FNJ7, FNK0, FNR2. User defined functions are restricted to one line. A function may be defined to be any expression, but may have only one argument. In the example, B and C are variables that are used in the program. Executing the DEF statement defines the function. User defined functions can be redefined by executing another DEF statement for the same function. User defined string functions are not allowed. "V" is called the dummy variable.
	110 Z=FNA(3)	Execution of this statement following the above would cause Z to be set to 3/B+C, but the value of V would be unchanged.
DIM	113 DIM A(3),B(10)	Allocates space for matrices. All matrix elements are set to zero by the DIM statement.
	114 DIM R3(5,5),D$(2,2,2)	Matrices can have more than one dimension. Up to 255 dimensions are allowed, but due to the restriction of 113 characters per program line (as opposed to 132 characters allowed per print line) the practical maximum is about 34 dimensions.
	115 DIM Q1(N),Z(2*I)	Matrices can be dimensioned dynamically during program execution. If a matrix is not explicitly dimensioned with a DIM statement, it is assumed to be a single dimensioned matrix whose single subscript may range from 0 to 10 (11

		elements).
	117 A(8)=4	If this statement was encountered before a DIM statement for A was found in the program, it would be as if a DIM A(10) had been executed previous to the execution of line 117. All subscripts start at zero (0), which means that DIM X(100) really allocates 101 matrix elements.
DIM	25 DIM A$(10,10)	Allocates space for a pointer and length for each element of a string matrix. No string space is allocated.
DIR	DIR 8	Displays a directory of the specified disc drive (8 in the example) on the console. If the drive number is not specified, drive 8 is assumed.
DISPLAY	430 DISPLAY A, A$	DISPLAY operates the in the same manner as PRINT, except that DISPLAY drives the CRT terminal while PRINT drives the printer. On the DRC DISPLAY and PRINT both drive the same RS232 port. When a change between the CRT or printer is encountered, the CRTSTR or PRTSTR string of characters is sent to disable or enable the printer port on the CRT ternimal. Note that the same character counter is used by both DISPLAY and PRINT. It is suggested that DISPLAY and PRINT statements ending with semicolons (;) or commas (,) not be intermixed, since the shared character counter will cause strange print formatting.
DISPLAY USING		Formats text according to a string expression. See PRINT USING.
END	999 END	Terminates program execution without printing a BREAK message. (see STOP). CONT after an END statement causes execution to resume at the statement after the END statement. END can be used anywhere in the program, and is optional.
ERR	550 PRINT ERR	ERR returns a numeric error code. If ERR=1, then metering data was not present during the last METER, RAISE, LOWER, or STATUS function. Reading ERR clears it (sets it to 0). ERR cannot be used on the left side of a LET statement. As such, it may be thought of as a zero argument function.
EXP(X)	150 PRINT EXP(X)	Gives the constant "E" (2.71828) raised to the power of X. (E^X). The maximum argument that can be passed to EXP without overflow occuring is 87.3365.
FOR	300 FOR V=1 TO 9.3 STEP .6	(see NEXT statement) V is set equal to the expression following the equal sign, in this case 1. This value is called the initial value. Then the statements between FOR and NEXT are executed. The final value is the value of the expression following the TO. The step is the value of the expression following STEP. When the NEXT statement is encountered the step is added to the variable.

310 FOR V=1 TO 9.3 If no STEP was specified, it is assumed to be one. If the step is positive and the new value of the variable is <= the final value (9.3), or the step value is negative and the new value of the variable is => the final value, then the first statement following the FOR statement is executed. Otherwise, the statement following the NEXT statement is executed. All FOR loops execute the statements between the FOR and the NEXT at least once, even in cases like FOR V=1 TO 0.

315 FOR V=10*N TO 3.4/Q STEP SQR(R) Note that expressions (formulas) may be used for the initial, final and step values in a FOR-NEXT loop. The values of the expressions are computed only once, before the body of the FOR-NEXT loop is executed.

320 FOR V=9 TO 1 STEP -1 When the statement after the NEXT is executed, the loop variable is never equal to the final value, but is equal to whatever value caused the FOR-NEXT loop to terminate. The statements between the FOR and its corresponding NEXT in both examples above (310 & 320) would be executed 9 times.

330 FOR W=1 TO 10: FOR W=1 TO 7: NEXT W: NEXT W

ERROR!!! Do not use nested FOR-NEXT loops with the same index variable. FOR-NEXT loop nesting is limited only by the available memory.

FRE(X) 270 PRINT FRE(0) Gives the number of memory bytes currently unused by H&F BASIC if the argument is a numeric, such as example line 270.

275 PRINT FRE(A$) Gives the number of bytes of memory allocated to string storage and currently unused. Can be changed using the CLEAR command.

GOTO 50 GOTO 100 Branches to the statement specified.

GOSUB 10 GOSUB 910 Branches to the specified statement (910) until a RETURN is encountered; when a branch is then made to the statement after the GOSUB. GOSUB nesting is limited only by the available memory.

IF...GOTO 32 IF X<=Y+23.4 GOTO 92 Equivalent to IF-THEN, except that IF-GOTO must be followed by a line number, while IF-THEN can be followed by either a line number or another statement.

IF...THEN IF X<10 THEN 5 Branches to specified statment if the relation is true.

20 IF X<0 THEN PRINT "X LESS THAN 0" Executes all of the statements on the remainder of the line after the THEN if the relation is true.

25 IF X=5 THEN 50:Z=A WARNING!!! The "Z=A" will never be executed because if the relation is true, H&F BASIC will branch to line 50. If the relation is false, H&F BASIC will proceed to the line after line 25.

26 IF X<0 THEN PRINT "ERROR, X IS NEGATIVE":GOTO 350

In this example, if X is less than 0, the PRINT statement will be executed and then the GOTO statement will branch to line 350. If the X was

		0 or positive, H&F BASIC will proceed to execute the lines after line 26.
INKEY$	10 A$=INKEY$(N)	Inkey$ gets a single byte string from the specified I/O device (0 - Console, 1 - Printer, 2 - Direct Connect Modem, 3 - RS232 port 3). If there has been no keystroke at the specified device, inkey$ returns with the null string ("").
INPUT	3 INPUT V,W,W2	Requests data from the terminal. Each value typed in must be separated by a comma (,). The last value should be followed by a carriage return. If more data was requested in an INPUT statement than was typed in, a "??" is printed and the rest of the data should be typed in. If more data was typed in than was requested, the extra data will be ignored. Strings must be input in the same form as they are specified in DATA statements (leading spaces require quotes surrounding string).
	5 INPUT "VALUE";V	Optionally types a prompt string ("VALUE") before requesting data from the terminal. The prompt string is typed on the console (device 0). Typing a carriage return in response to an INPUT statement leaves the variable unchanged.
	40 INPUT X$	Reads a string from the CRT. String does not have to be quoted, but if not, leading blanks will be ignored and the string will be terminated on a "," or ":" character.
INPUT#	150 INPUT#2,A$ 160 N=2 170 INPUT#N,"HI";A$	INPUT# allows input from a selected input output device. The device is specified as a numeric expression following the # sign and is followed by a comma or semicolon. A prompt string can be included after the "#N,", and it will be sent to the selected device. Device numbers are: 0 - Console CRT, 1 - Printer plugged into CRT peripheral port, 2 - Direct Connect Modem, 3 - RS232 port 3, which appears on J23 on systems equipped with the direct connect modem.
INT(X)	140 PRINT INT(X)	Returns the largest integer less than or equal to its argument. For example: INT(.23)=0, INT(7)=7, INT(-.1)=-1, INT(-2)=-2, INT(1.1)=1. The following would round X to D decimal places: INT(X*10^D+.5)/10^D
LEFT$(X$,I)	310 PRINT LEFT$(X$,I)	Gives the leftmost I characters of the string expression X$. If I<=0 or >255 an FC error occurs.
LEN (X$)	220 PRINT LEN (X$)	Gives the length of the string expression X$ in characters (bytes). Non-printing characters and blanks are counted as part of the length.
LET	300 LET W=X	Assigns a value to a variable.
	310 V=5.1	LET is optional.
	27 LET A$="FOO"+V$	Assigns the value of a string expression to a string variable. LET is optional.
LINE	600 IF LINE(1)>45 THEN 10	LINE is an integer variable holding the current

		line number of the specified device (device 1 or the printer in this case). Separate line counters are maintained for each device. The line number is updated each time a line feed is sent to the device. LINE is incremented on each line feed until LINE(N) exceeds MAXLINE(N), whereupon LINE(N) is returned to zero.
	602 LINE(1)=0	LINE(N) may be set to any number in an assignment (LET) statement. Typically, it is set to 0 after the operator has set the printer paper to the top of a page.
LIST	LIST	Lists entire current program.
	LIST 100	Lists program starting at line 100.
	?:LIST	? is equivalent to the word PRINT, so it sets the I/O device number to 1. LIST does not change the I/O device number, so the listing is sent to the printer.
LOAD	LOAD FN$,8 10 LOAD"HI",8	Loads the specified file name (FN$) from the specified disc drive (8) into the DRC. Any previous program is deleted from the DRC. All variables are cleared. If LOAD is used within a running program, the running program is deleted, the new program is loaded and run. This allows for simple chaining of programs.
LOAD EEPROM		Loads a program from non-volatile memory (EEPROM or Electrically Erasable Programmable Read Only Memory). On system reset, the EEPROM program is loaded and run. This may be used to auto-start a simple logging program, or may be used to auto-boot a more complex program from disc.
LOAD EEPROM N		Loads a program from non-volatile memory (EEPROM) at the specified address (N). This allows loading larger programs than can be contained in the "boot" section of the processor board EEPROM. The RAM board in the system must be partially loaded with EEPROM. When loaded as H&F suggests, N=24576.
LOG(X)	160 PRINT LOG(X)	Gives the natural (Base E) logarithm of its argument X. To obtain the base Y log of X, use the formula LOG(X)/LOG(Y). For example, the base 10 (common) log of 7 = LOG(7)/LOG(10).
LOWER	500 LOWER (3,2)	The LOWER statement selects the desired site and channel. It then sends a LOWER pulse to the appropriate remote control. In the example, the LOWER pulse was sent to site 3, channel 2.
	250 PRINT LOWER (S,C)	Selects site S, channel C, sends a series of LOWER pulses. Returns the resulting meter reading. The length of time that the LOWER output of the DRC190 is held while waiting for the reading to stabalize is determined by the "sample delay" programmed during calibration of the particular channel.
LOWER$	255 PRINT LOWER$ (S,C)	Does the same as LOWER except that a string is returned. The string consists of 3 label characters, an equal sign, the reading, and 2

label characters. ICP= 4.52AP

MAXLINE 603 IF MAXLINE(1)-LINE(1)<5 THEN GOSUB 1000

MAXLINE(N) is a system defined variable that represents the maximum line number of device N. This can be used with LINE for end of page detection. This is more valuable than an EOP (End Of Page) function available on some systems, since it allows you to determine how much space is left on a page instead of merely knowing that you just ran off the end. It allows log entries to include multiple lines without the entry going over a page break. This is similar to the .CP (conditional page break) command in Wordstar.

10 MAXLINE(1)=65 MAXLINE(N) can also be set with an assignment statement (LET). Since line numbers start at zero, MAXLINE should be the number of lines per page less one. MAXLINE cannot be set using an input or read.

MDMSPD 10 MDMSPD=3
20 PRINT MDMSPD

MDMSPD allows setting of the direct connect modem board speed. This modem will operate at either 300 or 1200 bits per second. MDMSPD indicates the speed in hundreds of bits per second. MDMSPD can be read or written to (can be on either side of the equals sign in an assignment statement), but cannot be written to or set using an INPUT or READ statement.

MDMSTAT$ 10 IF MDMSTAT$="A" THEN 100

MDMSTAT$ contains the last received modem status character. These characters come from the direct connect modem. The direct connect modem sends a control-N followed by a status characater and a line feed and a carriage return. The DRC190 firmware captures all control-N - status character sequences, preventing the control-N or the status character from showing up in an INPUT#2 or INKEY$(2) result. The line feed, carriage return sequence following the control-N status character sequence is not trapped out, however. MDMSTAT$ holds the single character status message from the direct connect modem, as described here, and in the modem manual (reprinted in the rear of this manual). MDMSTAT$ characters are: R - Indicates the line the modem is connected to is ringing. A - Indicates the modem has answered an incoming call, or an out-going call has been answered by a remote modem. N - Indicates that there was no answer on an out-going call (and the modem has disconnected itself). D - Indicates the modem has disconnected. Note that the modem module sends a D only when it initiates the disconnect (due to a loss of carrier from the other modem). The DRC190 firmware sets MDMSTAT$ to "D" on receiving a disconnect status message from the modem module

		(the remote modem has disconnected), or on an END command (control-N E) being sent to the modem module by the DRC190 application program (ie, PRINT#2,CHR$(14);"E"). Finally, note that MDMSTAT$ changes only on receiving a status message from the modem, or on receiving an END command from the applications program. If MDMSTAT$ is N due to no answer on a previous call, and another call is placed, MDMSTAT$ will remain N until that call is answered. It is suggested that prior to placing a call, the modem be sent an END command, forcing MDMSTAT$ to D. Then, when the call is placed, the applications program can watch MDMSTAT$ for A or N indicating that the call was answered or not answered.
MID$(X$,I)	330 PRINT MID$(X$,I)	MID$ called with two arguments returns characters from the string expression X$ starting at character position I. If I>LEN(X$), then MID$ returns a null (zero length) string. If I<=0 or >255, an FC error will occur.
MID$(X$,I,J)	340 PRINT MID$(X$,I,J)	MID$ called with three arguments returns a string expression composed of the characters of the string expression X$ starting at the Ith character for J characters. If I>LEN(X$), MID$ returns a null string. If J specifies more characters than are left in the string, all characters from the Ith are returned.
METER	260 PRINT METER (S,C)	Selects site S, channel C and returns the resulting meter reading. This can also be called with a single argument. That argument is (512*SiteNumber)+(1*Raise)+(2*Lower)+ChanNum where Raise is 1 if a RAISE pulse is required during the reading and Lower is 1 if a LOWER pulse is required during the reading.
METER$	265 PRINT METER$ (S,C)	Same as meter, but returns a string consisting of label, reading and units.
MODEMTST	MODEMTST	Puts the DRC190 in the modem test routine. This routine allows checking of transmit levels and frequencies, and checks modem demodulator tuning. See the processor board adjustments section for further information.
NEW	NEW	Deletes current program and all variables.
NEXT	340 NEXT V	Marks the end of a FOR-NEXT loop.
	345 NEXT	If no variable is given, it matches the most recent FOR.
	350 NEXT V,W	A single NEXT may be used to match multiple FOR statements. Equivalent to NEXT V:NEXT W.
NOT	100 IF NOT A=B THEN 200	A logical or bitwise NOT. NOT inverts the logic of the expression following it. In the example, if A=B was TRUE, the NOT A=B is FALSE. Note that TRUE is represented by a -1 in 16 bit integer form while FALSE is represented by a zero. NOT does a bitwise inversion, changing all the 1s to 0s and 0s to 1s. This corresponds to the described logical NOT.

ON...GOSUB

110 ON I GOSUB 50,60 Identical to ON-GOTO except that a subroutine call (GOSUB) is executed instead of a GOTO. RETURN from the GOSUB branches to the statement after the ON-GOSUB.

ON...GOTO 100 ON I GOTO 10,20,30 Branches to the line indicated by the I'th number after the GOTO. That is:

```
    IF I=1 THEN GOTO 10
    IF I=2 THEN GOTO 20
    IF I=3 THEN GOTO 30
```

If I=0 or I attempts to select a nonexistent line (>=4 in this case), the statement after the ON statement is executed. However, if I is >255 or <0, an FC error will result. As many line numbers as will fit on a line can follow an ON...GOTO.

105 ON SGN(X)+2 GOTO 40,50,60 This statement will branch to line 40 if the expression X is less than zero, to line 50 if it equals zero, and to line 60 if it is greater than zero.

OR 100 IF A=B OR A=C THEN 200

Logical or bitwise OR. If the expression on either side of the OR is true, then the entire expression is true. In a bitwise OR, if a particular bit position in either of the arguments is 1, then that bit position will be a 1 in the result. For example, 1 OR 3 = 3.

PEEK(X) 356 PRINT PEEK(I) The PEEK function returns the contents of memory address I. The value will be >=0 and <=255. If I is >65535 or <0, an FC error will occur.

POKE POKE I,J The POKE statement stores the byte specified by its second argument (J) into the location of the memory map specified by its first argument (I). The byte to be stored must be =>0 and <=255, or an FC error will occur. The address (I) must be =>0 and <=65535 or an FC error will occur. Careless use of the POKE will probably cause you to "POKE" H&F BASIC to death; that is the machine will hang, and you will have to reset it, losing your program. The main use of POKE on the DRC190 is to allow you to change certain machine characteristics that are stored in EEPROM, or to give direct access to input output devices (such as the parallel port on the direct connect modem card).

POS(I) 260 PRINT POS(I) Gives the current print head or cursor position on the last used I/O device. The leftmost position is position 0.

PRINT

```
350 PRINT X,Y;Z
370 PRINT
380 PRINT X,Y;
390 PRINT"VALUE=";A
400 PRINT A2,B,
```

Prints the value on the printer (plugged into peripheral port on the CRT terninal). If the list of values to be printed does not end with a comma or a semicolon, then a carriage return/line feed is executed after all the values have been printed. Strings enclosed in quotes (") may also be printed. If a semicolon

		separates two expressions in the list, their values are printed next to each other. If a comma appears after an expression in the list, then the print head is advanced to the next TAB position, which may be on the next line. A comma moves the print head to the next "comma-field", each of which is 13 characters wide. If there is no expression to be printed, as in line 370 of the examples, then a carriage return/line feed is sent to the printer.
	60 PRINT X$	Prints the string expression on the printer.
	70 PRINT "FOO"+A$	
	410 PRINT MID$(A$,2)	String expressions may be printed.
PRINT USING	10 PRINT USING "##.##" A 20 A$="###.##" 30 PRINT USING A$ 3.4	PRINT USING allows numbers that are not in exponential form to be formatted. They are formatted according to the string or string expression following the word USING. The number of #s prior to the decimal point corresponds to the number of digits to be printed prior to the decimal point. Leading zeroes will be suppressed by the printing of spaces instead of zeroes. Recall that an additional space is saved for the sign of the number. The number of #s following the decimal point corresponds to the number of digits that will be printed after the decimal point. Trailing zeroes are not replaced with spaces. Multiple USINGs may be used in one PRINT statement to change the print format without having to do another PRINT. PRINT USING makes it easy to align decimal points in a table.
RAISE	100 RAISE (S,C)	The RAISE statement selects site S, channel C, and then sends a RAISE command to the appropriate remote control.
	279 PRINT RAISE(S,C)	Selects site S, channel C and sends a series of RAISE pulses. Returns the resulting reading.
RAISE$	280 PRINT RAISE$(S,C)	Does same as RAISE, except that a string consisting of the label, reading, and units is returned.
READ	490 READ V,W	Reads data into specified variables from a DATA statement. The first piece of data will be the first piece of data listed in the first DATA statement of the program. The second piece of data read will be the second piece listed in the first DATA statement, and so on. When all of the data have been read from the first DATA statement, the next piece of data to be read will listed in the second data statement of the program. Attempting to read more data than there is in all the DATA statements in a program will cause and OD (Out of Data) error.
	50 READ X$	Reads a string from DATA statements within the program. Strings do not have to be quoted, but if they are not, they are terminated by a "," or ":" character or end of line and leading spaces are ignored. See DATA for the format of string

		data.
REM	500 REM NOW SET V=0	Allows the programmer to put comments in her/his program. REM statements are not executed, but can be branched to. A REM statement is terminated by the end of a line, but not a ":".
	505 REM SET V=0: V=0	In this case, V=0 will never be executed by H&F BASIC.
	506 V=0: REM SET V=0	In this case, V=0 will be executed.
RESTORE	510 RESTORE	Allows the re-reading of DATA statements. After a RESTORE, the next piece of data read will be the first piece listed in the first DATA statement of the program. The second piece of data read will be the second piece listed in the first DATA statement, and so on, as in a normal READ operation.
RETURN	50 RETURN	Causes a subroutine to return to the statement after the most recently executed GOSUB.
RIGHT$(X$,I)	320 PRINT RIGHT$(X$,I)	Gives the rightmost I characters of the string expression X$. When I<=0 or >255 an FC error will occur. If I>=LEN then RIGHT$ returnss all of X$.
RND(X)	170 PRINT RND(X)	Generates a random number between 0 and 1. The argument X controls the generation of random numbers as follows: X<0 starts a new sequence of random numbers using X. Calling RND with the same X starts the same random number sequence. X=0 gives the last random number generated. Repeated calls to RND(0) will always return the same random number. X>0 generates a new random number between 0 and 1. Note that (B-A)*RND(1)+A will generate a random number between A and B.
RUN	RUN	Starts execution of the program currently in memory at the lowest numbered statement. Run deletes all variables and restores DATA.
	RUN 100	Starts execution of program in memory at line 100. Run deletes all variables and restores DATA.
SAVE	SAVE "FN$",8	Saves program currently loaded in DRC RAM to the specified drive number (8) under the specified file name (FN$).
	SAVE "@0:HI",8	If an existing file on the disc with he same file name is to be replaced, FN$ must start with "@0:".
SAVE EEPROM		Saves the program currently in DRC RAM to non-volatile memory (EEPROM), in the boot program section of the processor board EEPROM, if the program will fit. About 1 Kbyte is available. This program can be reloaded by using the LOAD EEPROM command, or by doing a system reset. On system reset, the program is reloaded and run.
SAVE EEPROM 24576		If an address is specified after the SAVE EEPROM command, the program is saved in EEPROM starting at that address. This is allowed if a portion of the RAM board has been loaded with EEPROM. See the theory of operation section on the RAM board.

SGN(X)	230 PRINT SGN(X)	Gives 1 if X>0, 0 if X=0, and -1 if x<0.
SIN(X)	190 PRINT SIN(X)	Gives the sine of the expression X. X is interpretted as being in radians. Note: COS(X)=SIN(X+3.14159/2) and that 1 Radian is 180/PI degrees, or 1 Radian = 57.2958 degrees; so that the sine of X degrees is SIN(X/57.2958).
SPC(I)	250 PRINT SPC(I)	Prints I space (or blank) characters on the printer. May be used only in a PRINT or DISPLAY statement. X must be >=0 and <=255 or an FC error will occur.
SQR(X)	180 PRINT SQR(X)	Gives the square root of the argument X. An FC error will occur if X<0.
STATUS	190 IF STATUS (S,C) THEN 300 200 ?STATUS(S,C)	STATUS returns a TRUE or FALSE according to whether the status indicator at site S, channel C is ON or OFF. TRUE is represented by a -1. False is represented by a 0. Note that on this writing, the firmware to support the word STATUS has not been written. Use of the word will return a syntax error.
STOP	9000 STOP	Causes a program to stop execution and to enter the command mode. Prints BREAK IN LINE 9000 (for this example). The CRT will beep once per second until a keystroke to indicate the program execution has stopped. CONT after a STOP branches to the statement following the STOP.
STR$	290 PRINT STR$(X)	Gives a string which is the character representation of the numeric expression X. For instance, STR$(3.1)=" 3.1".
SWAP	SWAP A,B SWAP A$,B$ SWAP A(N),A(N+1) SWAP A$(N),A$(N+1)	Swaps the contents of two variables. If A=4 and B=5 prior to executing the SWAP, A=5 and B=4 after executing the swap. Swaps can be between any two numeric variables or between any two string variables. Either or both variables can be elements of an array or matrix. An attempt to swap a string variable with a numeric variable results in a TM (Type Mismatch) error.
TAB(I)	240 PRINT TAB(I)	Spaces to the specified print position on the terminal. May be used only in PRINT or DISPLAY statements. Zero is the leftmost column of the printer, while 131 is the rightmost. The rightmost column on the CRT varies with the CRT used. Care should be used to insure that you do not TAB beyond the capability of the CRT. If the carriage (or cursor) is beyond position I, no change is made. Note that there is only one position counter shared between the printer and CRT.
TAN(X)	200 PRINT TAN(X)	Gives the tangent of the expression X where X is in radians.
TIME	800 IF TIME>=RT+20000 THEN GOSUB 1000 1000 TIME=RT	TIME is an integer variable representing the time of day in 24 hour format. 235900 represents 11:59:00 pm. 0 represents midnight. Use of TIME in comparisons allows program operation to vary with time of day. Example line 800 causes the

		DRC190 to go to subroutine 1000 if it has been 2 hours or more since RT was last updated. This would typically cause a set of readings to be printed every two hours. Time is set using a LET statement. TIME is updated by the internal clock. TIME cannot be set using an input or read statement.
TIME$	PRINT TIME$	Prints time (set up above) in HH:MM:SS 12 hour format.
TROFF	100 TROFF TROFF	Turns TRace OFF. May be used within a program, or may be used in immediate or command mode.
TRON	100 TRON 0 TRON 0	Turns on program trace. With program trace on, a CRLF followed by the line number of the line about to be executed is sent to the specified I/O device, and followed by a colon and a space. TRON can be used within a program or in the immediate or command mode.
VAL(X$)	280 PRINT VAL(X$)	Returns the string expression X$ converted to a number. For instance, VAL("3.1")=3.1 . If the first non-space character of the string is not plus (+) or minus (-) sign, a digit or a decimal point (.) then zero will be returned.

OPERATORS

SYMBOL	SAMPLE STATEMENT	PURPOSE/USE
=	A=100	Assigns a value to a variable.
	LET Z=2.5	The LET is optional.
-	B=-A	Negation. Note that 0-A is subtraction, while -A is negation.
²	130 PRINT X²3	Exponentiation. Sample is X raised to the third power (X*X*X). 0²0=1. 0 to any other power = 0. A²B with A negative and B not an integer gives an FC error.
*	140 X=R*(B*D)	Multiplication.
/	150 PRINT X/1.3	Division.
+	160 Z=R+T+Q	Addition.
-	170 J=100-I	Subtraction.

RULES FOR EVALUATING EXPRESSIONS:

1. Operations of higher precedence are performed before operations of lower precedence. This means the multiplication and divisions are performed before additions and subtractions. As an example, 2+10/5 equals 4, not 2.4 . When operations of equal precedence are found in a formula, the left hand one is executed first: 6-3+5=8, not -2.

2. The order in which operations are performed can always be specified explicitly through the use of parentheses. For instance, to add 5 to 3 and then divide by 4, we would use (5+3)/4, which equals 2. If instead we had used 5+3/4, we would get 5.75 as a result (5 plus 3/4).

The precedence of operators used in evaluating expressions is as follows, in order beginning with the highest precedence. Operators listed on the same line have the same precedence.

1. Formulas in parenthesis are always evaluated first.
2. ² Exponentiation.
3. Negation. -X where X may be a formula.
4. * / Multiplication and division.
5. + - Addition and subtraction.
6. Relational Operators (Equal precedence for all 6):

=	Equal
<>	Not equal
<	Less than
>	Greater than
<=	Less than or equal
>=	Greater than or equal

7. NOT Logical and bitwise "NOT". Like negation, NOT takes only the formula to its right as an argument.
8. AND Logical and bitwise "AND"
9. OR Logical and bitwise "OR"

Relational Operator expressions will always have a value of TRUE (-1) or a value of FALSE (0). Therefore, (5=4)=0, (5=5)=-1, (4>5)=0, (4<5)=-1, etc.

The THEN clause of an IF statement is executed whenever the formula after the IF is not equal to 0. That is to say, IF X THEN. . . is equivalent to IF X<>0 THEN. . .

SYMBOL	SAMPLE STATEMENT	PURPOSE/USE
=	10 IF A=15 THEN 40	Expression equals expression.
<>	70 IF A<>0 THEN 5	Expression doesn't equal expression
>	30 IF B>100 THEN 8	Expression greater than expression
<	160 IF B<2 THEN 10	Expression less than expression.
<=,=<	180 IF A<=1 THEN 2	Expression less than or equal to expression.
>=,=>	190 IF Q>=R THEN 7	Expression greater than or equal to expression.
AND	2 IF A<5 AND B>2 THEN 7	If expression 1 (A<5) AND expression 2 (B>2) are BOTH true, then branch to line 7.
OR	IF A<1 OR B<2 THEN 2	If EITHER expression 1 (A<1) or expression 2 (B<2) is true, then branch to line 2.
NOT	IF NOT Q3 THEN 4	If expression "NOT Q3" is true (because Q3 is false), then branch to line 4. NOT(-1)=0, NOT(TRUE)=FALSE.

STRING OPERATORS

NAME	EXAMPLE	PURPOSE/USE
= < > <= >=		String comparison operators. Comparison is made on the basis of ASCII codes, a character at a time until a difference is found. If during the comparison of two strings, the end of one is reached, the shorter string is considered smaller. Note that "A " is greater than "A" since trailing spaces are significant.
+	30 LET Z$=R$+Q$	String concatentation. The resulting string must be less than 256 characters in length or an LS error will occur.

SPECIAL CHARACTERS

CHARACTER USE

Control-U Erases current line being typed and types a carriage return/line feed.

BACKSPACE Erases last character typed. If no more characters are left on the line, types a carriage return/line feed.

CARRIAGE RETURN A carriage return must end every line typed in. Returns printhead or CRT cursor to first position on next line.

CONTROL C Interrupts execution of a program or list command. The DRC returns to command level. Prints "BREAK IN LINE XXXX" , where XXXX is line number of next statement to be executed. On a program break due to a control-C or an error, or a STOP statement, the CRT beeps once per second until a keystroke to indicate that the program has stopped.

: (colon) A colon is used to separate statements on a line. Colons can be used in direct and indirect statements. The only limit on the number of statements per line is the program line length (113 characters). It is not possible to GOTO or GOSUB to the middle of a line, although it is possible to RETURN to the middle of a line.

? Question marks are equivalent to PRINT. For instance, ?2+2 is equivalent to PRINT 2+2. Question marks can also be used in indirect statements. 10 ? X , when listed will list at 10 PRINT X .

ERROR MESSAGES

After an error occurs, H&F BASIC generally returns to the command level and types OK. If the error occurred during the execution of a program (instead of during an immediate command), the error will be displayed and the console bell (or beep) will signal once a second until any key is pressed on the console. This prevents an error in the program from shutting the system down with no warning. The system operator is notifed of the error and is signalled until the error is acknowledged. After an error, H&F BASIC to command level, variable values and the program text remain intact, but the program cannot be continued, and all GOSUB and FOR context is lost. Variables can be inspected using PRINT or DISPLAY to see why the program crashed.

When an error occurs in a direct statement, no line number is printed. The error message format is:

Direct Statement	XX ERROR
Indirect Statement	XX ERROR IN YYYY

In each case, XX is the error code and YYYY is the line number where the error occured for the indirect statement.

The error codes and their meanings follow:

BS Bad Subscript. An attempt was made to reference a matrix element which is outside the dimensions of the matrix. This error can occur if the wrong number of dimensions are used in a matrix reference. For example, LET A(1,1,1)=Z when A has been dimensioned DIM A(2,2).

CN Continue Error. Attempt to continue a program when none exists, an error occured, or after a new line was typed into the program.

DD Double Dimension. After a matrix was dimensioned, another dimension statement for the same matrix was encountered. This error often occurs if a matrix has been given the default dimension of 10 because a statement like A(I)=3 is encountered and then later in the program a DIM A(100) is found.

FC Function Call error. The parameter passed to a math or string function was out of range. FC errors can occur due to:

a- a negative matrix subscript (LET A(-1)=0)
b- an unreasonably large matrix subscript (>32767)
c- LOG negative or zero argument
d- SQR negative argument
e- A^B with A negative and B not an integer
f- calls to MID$, LEFT$, RIGHT$, PEEK, POKE, TAB, SPC, or ON GOTO with an improper argument.
g- A METER, RAISE, LOWER or STATUS call with improper channel or site.

ID Illegal Direct. You cannot use INPUT or DEFFN statement as a direct command.

LS Long String. Attempt was made by use of concatenation operator to create a string more than 255 characters long.

NF Next without For. The variable in a NEXT statement corresponds to no previously executed FOR statement.

OD Out of Data. A READ statement was executed but all of the DATA statements have already been read. The program tried to read too much data or insufficient data was included in the program.

OM Out of Memory. Program too large, too many variables, too many FOR loops, too many GOSUBS, too complicated an expression or any combination of the above.

OS Out of string space. Too many characters have been stored as strings. Can be fixed by reusing string variables, setting variables no longer in use to the null string (""), or use the CLEAR statement to allocate more string space.

OV Overflow. The result of a calculation was too large to be represented in H&F BASIC's number format. If an underflow occurs, zero is given as the result and execution continues without any error message being printed.

RG RETURN without GOSUB. A RETURN statement was encountered without a previous GOSUB statement being executed.

SB Serial Bus error. An attempt was made to access a device (such as a disc drive) on the serial bus, and that device is not responding, or that device is in an error condition (such as a request to load a non-existant file).

SN Syntax error. Missing parenthesis in an expression, illegal character in a line, incorrect punctuation, etc.

ST String Temporaries. A string expression was too complex. Break it into two or more shorter ones.

TM Type Mismatch. The left hand side of an assignment statement was a numeric variable and the right side was a string, or vice versa; or, a function which expected a string argument was given a numeric one or vice versa.

UF Undefined Function. Reference was made to a user defined function which had never been defined.

US Undefined Statement. An attempt was made to GOTO, GOSUB, or THEN to a statement which does not exist.

/0 Division by Zero.

SPEED HINTS

The execution of a program can be speeded up by using the space hints in the next section plus the following:

1. Define frequently used variables early in the program. Variables are stored in the order they are encountered. Putting frequently used ones at the start of the list minimizes search time.
2. Deleting the index variable in NEXT statements slightly improves speed.

SPACE HINTS

In order to make your program smaller and save space, the following hints may be useful.

1. Use multiple statements per line. There is a small amount of overhead (5 bytes) associated with each line in the program. Two of the five bytes contain the line number in binary. This means that no matter how many digits in your line number (minimum line number is 0, maximum is 64000), it takes the same number of bytes. Putting as many statements as possible on a line will cut down on the number of bytes used by your program.
2. Delete all unnecessary spaces from your program.
3. Delete all REM statements
4. Use variables instead of constants.
5. A program need not end with END, so an END statement can be deleted.
6. Reuse the same variables.
7. Use GOSUBs to execute sections of program statements that perform identical actions.
8. Use the zero elements of matrices.

Storage Allcoation Information

Simple (non-matrix) variables use 6 bytes; 2 for the name, 4 for the value. Simple non-matrix string variables also use 6 bytes; 2 for the variable name, 2 for the length, and 2 for the pointer.

Matrix variables use a minimum of 12 bytes. Two bytes are used for the name, two for the size of the matrix, two for the number of dimensions, and two for each dimension aong with 4 bytes for each of the matrix elements

String variables also use one byte of string space for each character in the string. This is true whether the string is a simple string variable or an element of a string matrix.

When a new function is defined by a DEF statement, 6 bytes are used to store the definition.

Reserved words like FOR, GOTO, NOT and the names of any intrinsic functions such as COS, INT, and STR$ take only one byte of program storage. All other characters in programs use one byte of program storage each.

When a program is being executed, spece is dynamically allocated on the stack as follows:

22 bytes for each active FOR NEXT loop.
6 bytes for each active GOSUB
4 bytes for each parenthesis in an expression
12 bytes for each temporary result in an expression

DERIVED FUNCTIONS

The following functions can be calculated using existing H&F BASIC functions.

SECANT(X)	1/COS(X)
COSECANT(X)	1/SIN(X)
COTANGENT(X)	1/TAN(X)
ARCSINE(X)	ATN(X/SQR(-X*X+1))
ARCCOS(X)	-ATN(X/SQR(-X*X+1))+1.5708
ARCSEC(X)	ATN(SQR(X*X-1))+(SGN(X)-1)*1.5708
ARCCSC(X)	ATN(1/SQR(X*X-1))+(SGN(X)-1)*1.5708
ARCCOT(X)	-ATN(X)+1.5708
SINH(X)	(EXP(X)-EXP(-X))/2
COSH(X)	(EXP(X)+EXP(-X))/2
TANH(X)	-EXP(-X)/(EXP(X)+EXP(-X))*2+1
SECH(X)	2/(EXP(X)+EXP(-X))
CSCH(X)	2/(EXP(X)-EXP(-X))
COTH(X)	EXP(-X)/(EXP(X)-EXP(-X))*2+1
ARGSINH(X)	LOG(X+SQR(X*X+1))
ARGCOSH(X)	LOG(X+SQR(X*X-1))
ARGTANH(X)	LOG((1+X)/(1-X))/2
ARGSECH(X)	LOG((SQR(-X*X+1)+1)/X)
ARGCSCH(X)	LOG((SGN(X)*SQR(X*X+1)+1/X)
ARGCOTH(X)	LOG((X+1)/X-1))/2

ASCII CHARACTER CODES

0	NUL	32	SPACE	64	@	96	Grave Accent
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(	72	H	104	h
9	HT	41	)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	S	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	ETB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESCAPE	59	;	91	[	123	Left Brace
28	FS	60	<	92	Back Slash	124	Vertical Bar
29	GS	61	=	93	]	125	Right Brace
30	RS	62	>	94	Caret	126	Tilde
31	US	63	?	95	Under Score	127	DEL

NOTE that the first column of codes are referred to as "control-codes." On most keyboards, these can be generated by holding the control key while striking the corresponding key two columns to the right. For example, control-@ yields a NUL. Control-G yields a BEL, etc.

The meaning of most control codes varies between terminals. The "standard" meanings are listed below.

NUL	Null, or all zeroes
SOH	Start of heading
STX	Start of text
ETX	End of text
EOT	End of transmission
ENQ	Enquiry
ACK	Acknowledge
BEL	Bell or alarm
BS	Backspace
HT	Horizontal tabulation
LF	Line feed
VT	Vertical tabulation
FF	Form feed
DC1	Device control 1
DC2	Device control 2
DC3	Device control 3
DC4	Device control 4
NAK	Negative acknowledge
SYN	Synchronous idle
ETB	End of transmission block
CAN	Cancel
EM	End of medium
SUB	Substitute
ESC	Escape
FS	File separator
GS	Group separator

CR	Carriage return
SO	Shift out
SI	Shift in
DLE	Data link escape
RS	Record separator
US	Unit separator
SP	Space
DEL	Delete

Monitor & EEPROM Initialization

The DRC190 firmware includes a "monitor" program to aid in troubleshooting.

This program can be reached in a couple of different ways. If the Basic program is operating properly, typing MONITOR at the command level will drop you into the monitor. If the Basic program is not operating properly (which can happen if the EEPROM gets crashed), the monitor program can be reached by generating a Non-Maskable Interrupt (NMI) on the processor. This is accomplished by grounding pin 6 on the processor itself momentarily, or grounding momentarily the wire wrap in installed on the back-plane between the A/D boards (if any) and the remainder of the boards in the system. When the monitor is called from Basic, the terminal port is left at the existing speed. If the monitor is called with an NMI, the serial port is set to 9600 bits/second.

When the monitor is initiated, the processor status should appear on the screen.

The processor status display has the following meaning:

S=0078	Contents of the stack pointer
C=22	Contents of the condition code register
B=FF	Contents of accumulator B
A=00	Contents of accumulator A
X=FF00	Contents of the index register
P=00C0	Contents of the program counter

Note that the monitor initialization sets the stack pointer to $00FF to insure the monitor stack is in RAM. The stack indication shown is actually 7 bytes below the stack pointer address prior to the NMI (which pushed everything on the stack).

Monitor commands include:

P	Prints the above processor status
B	Show the contents of accumulator B *
A	Show the contents of accumulator A *
X	Show the contents of index register X *
Mnnnn	Show contents of memory location nnnn where nnnn is in hex *
I	Initialize EEPROM

B, A, X and M allow the current contents of the register or location to be changed by keying in the 2 digit or 4 digit hexadecimal replacement. If no change is required, type carriage return.

Typing a Line Feed after M has displayed a memory location will show the contents of the next memory location.

The EEPROM holds calibration data used by the A/D converter subroutines. A new EEPROM will typically have all one's programmed in every address. This will cause a errors in the A/D routines, preventing the DRC190 from operating properly.

To initialize the EEPROM, type I while in the monitor. When the terminal does a carriage return and line feed, the EEPROM has been initialized.

EEPROM initialization sets all scaling factors to 1 (causing the displayed reading to be the A/D conversion in hundreds of microvolts). All labels and units are set to question marks. The site number is set to zero, the maximum site number to 1. The site delay is set to 100 mS. The CW ID frequency is set to 0. The CW ID message is initialized to "H&F DRC190[". The modem speed is set to 1200 bits per second. The terminal speed is set to 9.6 K bits per second.

Adjustments PC1441 A/D Board

The closest thing to an adjustment on the A/D board are the address select jumpers. These jumpers (along with the memory map PROM on the processor board) determine where the A/D board resides in the system memory map. This determines which channels this A/D board covers.

The A/D boards reside at $98X0 where X is the board number. Board 0 covers channels 0-9; board 1 covers channels 10-19. . . Board 9 covers channels 90-99. The board number is determined by the bottom 4 programming jumpers on P01 on the A/D board. The jumper is in place to program a 0 and is removed to program a 1. Wtih this in mind, the below table can be used to determine the proper jumper positioning for the desired board number.

Channels	Board	Top-Jumpers-Bottom
00..09	0	1 0 0 0 0 0 0 0
10..19	1	1 0 0 0 0 0 0 1
20..29	2	1 0 0 0 0 0 1 0
30..39	3	1 0 0 0 0 0 1 1
40..49	4	1 0 0 0 0 1 0 0
50..59	5	1 0 0 0 0 1 0 1
60..69	6	1 0 0 0 0 1 1 0
70..79	7	1 0 0 0 0 1 1 1
80..89	8	1 0 0 0 1 0 0 0
90..99	9	1 0 0 0 1 0 0 1

If desired, a voltage divider may be added in front of the A/D converter. This allows for sample voltages higher than the normal limit of +/- 2 volts. Normally, a jumper is shipped in the socket for R20 (the series resistor of the voltage divider) and R21 (the shunt resistor) is left open. The user may plug resistors into these locations to allow for higher sample voltages. For example, if R20 is 60 K and R21 is 10K, a maximum sample voltage of 14 volts would be allowed. The A/D would then have a resolution of 1.4 mV. The resistors used in this voltage divider should be very stable to minimize the introduction of errors. It is suggested that resistors with a temperature coefficeint of 5 ppm/degree C be used.

Adjustment of PC1442 Processor Board

The processor board has several adjustments. Most are connected with the modem.

The one adjustment that is not connected with the modem is the LCD viewing angle. Multiplexed LCDs have a limited viewing angle. The viewing angle adjustment is used to insure that the optimum viewing angle of the LCD is the angle that will actually be used. Outside the viewing angle range the LCD loses contrast, making it difficult to read.

It is suggested that the DRC190 be installed with the display at "eye-level". When the DRC190 is on the test bench and the cover is removed, adjust R37 for optimum contrast at the desired viewing angle.

Modem Adjustments

For the modem adjustments, it is suggested that the DRC190 be set up for two-wire telephone line operation. This can be done by jumpering pins 3 to 4 and pins 16 to 17 on J21 on the rear panel. Connect an audio voltmeter and a frequncy counter to pins 3 and 16 of J21. Connect a CRT terminal to J22 on the rear panel.

On the CRT terminal, type MODEMTST followed by a carriage return. Instructions for the modem adjustment will appear on the CRT. For preview, these instructions are:

Adjust R10 for desired TX level. Normally, this is 0 dBm as measured with the audio voltmeter.

Mark frequency 2090 - 2310 Hz. There is no adjustment for Mark frequency. We are just checking to insure it is within tolerance. If it is not, R15, C04 or U10 should be checked.

Space frequency 1140 - 1260 Hz. There is no adjustment for Space frequency. We are just checking to insure it is within tolerance. If it is not, R14, C04 or U10 should be checked.

Modem tuning. The DRC190 modem idles in the mark condition and sends space pulses that are one bit long at 1200 bits/second. These pulses are sent by setting the speed of the modem portion of the DUART to 4800 bits/second and sending 3 space bits following the start bit. The remaining bits of the character are marks. The receive portion of the DUART checks to see that the three space bits are received properly. If too few or too many space bits are received, you are instructed to adjust R21 (the demodulator tuning) to compensate. R21 should be adjusted to the middle of the range where modem tuning is indicated as being proper. At the edge of the range, the modem tuning is off by 1 bit at 4800 bits per second, or 0.25 bits at 1200 bits per second. This corresponds to a bias distortion of 25%, well within the error free range of the received portion of the DUART.

Adjust R41 for CW ID level. This instruction applies only if the DRC190 has the CW ID option. The DRC190 generates the CW ID tone using a timer in U16. The timer divides the system clock down to the FCC required CW ID frequency of 750 Hz (+/- 10 Hz). The tone is filtered from a square wave to a triangle wave. The FCC requires the CW ID level to be at 40% modulation (+/- 10%). Most transmitters used in TRL service (including those supplied by H&F) have a 6dB/octave preemphasis. Based on this, if the transmitter is adjusted to yield 100% modulation (1.5 KHz deviation) when driven with 2.2 KHz at 0 dBm (0.776 volts RMS), R41 should be adjusted for 0.910 volts RMS at 750 Hz.

As the DRC sets up for each adjustment, it puts the instructions for the adjustment on the CRT. When the adjustment is complete, pressing any key on the CRT keyboard will advance you to the next instruction.

This completes the adjustment of the PC1442 processor board.

Adjustment PC1443 Power Supply Interface

The PC1443 board serves three purposes: Disc drive interface, Status panel interface, and Power supply interface.

If the disc drive and status interface is provided, programming jumpers on P01 and the memory mapping PROM on the processor board determine the address of the interface. This interface is normally placed at $98F0, requiring jumpers as shown below. A 1 indicates the lack of a jumper, while a 0 indicates the presence of a jumper.

Address Line	Jumper Position	Jumper*
11	Top	1
10		0
09		0
08		0
07		1
06		1
05		1
04	Bottom	1

If the DRC was provided with an Uninterruptable Power Supply (UPS), then R02 is used to adjust the battery charger voltage. Disconnect one of the battery leads and connect a DVM to the two battery leads (reading the charger voltage). R02 should be adjusted to give 14.00 volts.

Connect the battery leads.

Subcarrier Transceiver Board 1444 Adjustment

There are several adjustments on the subcarrier transceiver board. Some of these are outlined in the installation section. They will all be covered here. The adjustments are broken into two groups: those on the subcarrier generator, and those on the subcarrier demodulator. We'll start with the subcarrier generator.

Subcarrier Transmit Frequency

Connect a frequency counter to the subcarrier output on the rear panel of the DRC190 or on P01-1 on the subcarrier board. Adjust R06 to get the frequency within 1 KHz of the desired frequency. Adjust R07 to get the exact frequency.

Subcarrier Deviation

Apply exactly 1 volt DC to the subcarrier generator input (P03-1). Adjust R09 so that the subcarrier frequency is 1 KHz offset from the frequency with no applied voltage. This corresponds to the recommended deviation of 1 KHz/volt.

Subcarrier Distortion

If the controls are far out of adjustment, connect an oscilloscope to the subcarrier output. Adjust R01 for a symmetrical waveform. Then connect a distortion analyzer or a spectrum analyzer to the subcarrier output. Alternately adjust R01 and R02 for minimum harmonic distortion.

Receive Local Oscillator Null

With no signal connected to the subcarrier input, adjust R11 for a null in the signal on U02 pin 2, as observed with an oscilloscope.

Receive Local Oscillator Frequency

To improve the local oscillator null, there is normally a jumper installed on P05. Remove this jumper when measuring the local oscillator frequency. With no signal connected to the subcarrier input (thereby disabling the AFC), adjust R17 for the desired local oscillator frequency, as measured on P05-1. The desired local oscillator frequency is 455 KHz - SCA, where SCA is the desired SCA receive frequency. For example, to receive 110 KHz, the local oscillator frequency should be 345 KHz. Return the jumper to P05.

Audio Select Jumpers

P03 is used to select whether DRC190 audio should be sent to the subcarrier generator, and whether received subcarrier audio should be sent to the DRC190. If the DRC190 is to drive the subcarrier generator, put the top two programming jumpers in place. Otherwise, put them on one pin each only.

If the DRC190 is to receive demodulated subcarrier signals, put the bottom two programming jumpers in place.

Subcarrier Frequency	Local Oscillator Frequency
26 KHz	429 KHz
67 KHz	388 KHz
92 KHz	363 KHz
110 KHz	345 KHz

Adjustment PC1445 Direct Connect Modem

The direct connect modem board has no adjustments other than its address in the memory map. The board normally resides at $98E0. This requires the jumpers to be installed as below. A 1 indicates the absence of a jumper, while a 0 indicates the jumper is present.

Top	1
	1
	1
Bottom	0

DRC190 Firmware Theory of Operation

This section will give you a general idea of the major portions of the DRC190 firmware. The firmware can be broken into two portions: The Basic interpreter and Everything Else.

The Basic Interpreter is licensed from Microsoft Corporation. As Microsoft has written many of the Basics in common usage, it should be familair to most.

The interpreter can either execute programs from memory (RAM) or can execute single line commands that are entered through the RS-232 port.

Several commands and functions have been added to Basic. These include METER, METER$, RAISE, RAISE$, LOWER, LOWER$, TIME, TIME$, DATE, DATE$, DAY, DAY$, LINE, plus several others. Most of these functions and commands request information from the "everything else" portion of the firmware. If this information is immediately available (such as TIME), it is immediately returned. If the requested data is not immediately available (such as METER), the Basic program awaits return of the data from the other portion of the firmware.

Basic is running in the "background", that is, there are several other processes interrupting it.

The interrupting processes include routines to handle:

Transmit through FSK modem.
Receive from FSK modem.
Check front panel keyboard.
Update front panel display.
Select an A/D channel.
Interpret the A/D sample and return it to requesting routine.
Update clock/calendar.

The front panel keyboard and LCD are considered a device that can transmit or receive data from other devices. Other devices include the A/D converter and Basic. As required, each of these devices generates messages in a message buffer. Part of the message includes to and from address for the message. Once the message is built, a routine called XFERMESS is called, transferring the message to the addressed device. If the addressed device is not at this site, the entire message is sent to the modem. When the modem receives a message, it is dumped into XFERMESS if the site number of this site matches the to site of the message.

Most of the devices that receive data through XFERMESS are not fast enough to take the data at high speed. Therefore, there are "circular" buffers at the input of several routines. There are circular buffers for the modem, the LCD, and the A/D converter.

Each of these interrupt routines is checked every 10 mS. The DUART is programmed to generate an interrupt every 10 mS.

The modem routines are quite interesting. It is necessary to insure that only one DRC190 transmits data at a time. This is accomplished by having all units time from the last valid character received. After 1 site delay (typically 50 mS, but can be changed by user), site 0 is allowed to transmit data, if it has any. If site 0 has no data, it does not bring up its carrier. After 2 site delays from carrier drop, site 1 is allowed to respond, if it has any data. This sequence is continued until either a site responds (where the sequence starts over again when it transmits its last character), or we reach the highest site number in the system. After the highest site number in the system is allowed to respond, we start timing for site 0 again. This scheme allows every site to respond while insuring that no contention takes place.

You can see the data transmitted or received by the modem by changing a byte of EEPROM called DEBUG. If DEBUG is 1, received data (in hexadecimal) will be sent to the terminal plugged into the RS232 port. The data has this format:

To Site Number in HEX with $80 added to it. MSB is To Site identifier
To Channel number in Hex
To Device Number in Hex
From Site number in Hex
From Channel number in Hex
From Device Number in Hex
Byte Count for following ASCII message, if any. No message makes this 00
0 or more bytes of ascii message
7 bit checksum (MSB cleared)

If the received message was not to this site, only the To Site number byte will be printed. If the message was to this site, the entire message as shown above will be printed. In addition, if the checksum indicates that the message is valid, an exclamation point will be printed at the end of the message.

If DEBUG is set to 2, the data transmitted by the modem will be displayed. This data takes the same form as above, except no exclamation point is printed.

To display received modem data, type the following on the terminal:

```
POKE 41965,1
```

To display transmitted modem data, type the following on the terminal:

```
POKE 41965,2
```

To return the DRC190 to normal, type the following on the terminal:

```
POKE 41965,0
```

Reading Schematics

The DRC190 was designed using a design automation system from Dasoft Corporation (Berkeley, CA). This system handles schematic capture, net list generation, schematic plotting, circuit board routing, and circuit board artwork plotting. The schematic plotting uses the newer international symbols for components, which many people may not be familiar with. This section describes the interpretation of the schematics.

The schematic for a board is normally broken into several pages to keep the plots down to a reasonable size. We've tried to put break the schematics into logical blocks for each page. The theory of operation section of the manual describes the operation of the board page by page of the schematic.

A component is shown on the schematic as a rectangle, perhaps with a notch in it. If there is no notch, the device has only inputs and outputs. The inputs are on the left, and the outputs are on the right. Many components have bi-directional lines, so we've tried to assign them in some logical manner (ie, processor lines on the left, peripheral lines on the right). If there is a notch in the component block, a three letter descriptor of the device (ie, CMP for comparator) is listed inside the block at the notch. Lines on the left above the notch are "control inputs", while lines on the right above the notch are "control outputs". Control inputs might be interrupt inputs on the processor, or interrupt outputs on a peripheral device.

For each input and output is a descriptor of the function of that input or output inside the component rectangle. These might be D0, D1, D2, etc. for data lines. Immediately outside the rectangle is the corresponding pin number on the device. Immediately outside the pin number is the "net name".

Net names are assigned to all signals on the board. Whenever it is desired to have a particular signal show up on a specified pin of a component, the corresponding net name is associated with that pin. For example, the processor and all peripheral chips and all memory chips on the processor board need the eight data lines (D0 through D7). You'll see that these net names have been listed in the appropriate places on the schematics. The schematic router and the printed ciruit board router connect together all pins that have the same net name. The connections on the schematic are not critical, as someone evaluating the schematic does not generally follow the lines around the drawing, but instead looks for the same net name showing up on the different devices. For that reason, the schematic router sometimes does not route a line, due to lack of space. The printed circuit router, of course, must always route a line. We've tried to choose standard net names, so that if you are having trouble with a particular chip, you can see that data shows up on these pins, address on these pins, IRQ on this pin, etc., without having to follow lines all over a large drawing. In addition, nets that leave a particular page of a schematic are listed on the sides of the schematic (if room is available), followed by a list of the page numbers that the net also appears on.

Finally, in accordance with the STD bus standard, active low net names end with an asterisk (*). The plotter prints the asterisk as a small block following the net name. For example, WE* is the active low Write Enable, OE* is the active low Output Enable, and R-W* is high for a Read, and low for a Write.

STD Bus Theory of Operation

The DRC190 is built around the STD bus, a standard 8 bit microcomputer bus. The STD standard calls for boards to be 4.5 x 6.5 inches. Some boards in the DRC190 are expaned to 4.5 x 9.6 inches to allow for additional circuitry. Most standard STD bus boards can be used in the DRC190. Note, however, that the 6800 STD does not provide for memory refresh. Dynamic RAM boards with on-board refresh (such as those available from Systek) can be used in the DRC190. Those dynamic memory boards that expect the processor to do memory refresh cannot be used. Most I/O boards can be used.

The pin out of the STD bus as used in the DRC190 is listed below.

Component Side		Solder Side	
1	+5 VDC Power	2	+5 VDC Power
3	Ground	4	Ground
5	No Connection	6	No Connection
7	D3	8	D7 - Most Significant Data Bit
9	D2	10	D6
11	D1	12	D5
13	D0 - Least Significant Data Bit	14	D4
15	A7	16	A15 - Most Significant Address Bit
17	A6	18	A14
19	A5	20	A13
21	A4	22	A12
23	A3	24	A11
25	A2	26	A10
27	A1	28	A9
29	A0 - Least Significant Address Bit	30	A8
31	WR* - Active Low Write Strobe	32	RD* - Active Low Read Strobe
33	IORQ* - Active Low I/O Request	34	MEMRQ* - Active Low Memory Request
35	IOEXP - Grounded in DRC190	36	MEMEXP - Grounded in DRC190
37	No Connection	38	Phase 2* - Active low system clock
39	No Connection	40	R/W* - Hi Read, Lo Write
41	No Connection	42	No Connection
43	No Connection	44	IRQ* - Active Low Interrupt Request
45	No Connection	46	NMI* - Non-Maskable Interrupt
47	SYSRES* - Active Low Reset	48	PBRES* - Grounded by reset button
49	No Connection	50	No Connection
51	No Connection	52	No Connection
53	Ground	54	Ground
55	+12 VDC	56	-12 VDC

Many of these pins are self explanatory. For those less familiar with microcomputer operation, the following descriptions are provided.

The data bus (D7-D0) allow bi-directional communications between the processor board and other boards in the system. When the processor board is sending data to another board, the R/W* line is low, indicating the processor is trying to write data. When the processor wants to input data, it leaves the R/W* line high, indicating it wants to read. The R/W* line is driven by the processor whether it wants to access devices on the processor board or off the board, or whether the device being accessed is a memory or Input/Output device. Data is passed on D7-D0 8 bits at a time, with D7 being the most significant bit.

A15-A0 form the address bus. The processor drives this bus. The processor sets up the address that it wishes to communicate with (perhaps reading an instruction or data, or writing data to either a memory location or an I/O device), sets IORQ* low if the address corresponds to an I/O device, or sets MEMRQ* low if the address corresponds to a memory location, sets up the data, if this is a processor write, then strobes RD* or WR* for a read or write strobe. The processor latches up the read data on the trailing positive edge of the RD*. The external device latches up the data on the trailing edge of the WR* strobe.

The 6802 processor uses memory mapped I/O. This allows the same STAA (store accumulator A) instruction to store the contents of an accumulator in a memory location, or in one of the registers of an I/O device. Other processors (Intel and Zilog series) use separate OUT instructions.

The memory map for the DRC190 is shown below:

Address Range		Device
0000 - 1FFF		U21, 8 Kbyte RAM chip on processor board
2000 - 27FF		U3A, 2 Kbyte RAM chip on optional RAM board
2800 - 2FFF		U3B, 2 Kbyte RAM chip on optional RAM board
3000 - 37FF		U4A, 2 Kbyte RAM chip on optional RAM board
3800 - 3FFF		U4B, 2 Kbyte RAM chip on optional RAM board
4000 - 47FF		U5A, 2 Kbyte RAM chip on optional RAM board
4800 - 4FFF		U5B, 2 Kbyte RAM chip on optional RAM board
5000 - 57FF		U6A, 2 Kbyte RAM chip on optional RAM board
5800 - 5FFF	*	U6B, 2 Kbyte RAM chip on optional RAM board
6000 - 67FF	*	U7A, 2 Kbyte RAM chip on optional RAM board
6800 - 6FFF	*	U7B, 2 Kbyte RAM chip on optional RAM board
7000 - 77FF	*	U8A, 2 Kbyte RAM chip on optional RAM board
7800 - 7FFF	*	U8B, 2 Kbyte RAM chip on optional RAM board
8000 - 87FF	*	U1A, 2 Kbyte RAM chip on optional RAM board
8800 - 8FFF	*	U1B, 2 Kbyte RAM chip on optional RAM board
9000 - 97FF	*	U2A, 2 Kbyte RAM chip on optional RAM board
9800 - 980F		Analog to digital converter board 0
9810 - 981F		Analog to digital converter board 1
9820 - 982F		Analog to digital converter board 2
9830 - 983F		Analog to digital converter board 3
9840 - 984F		Analog to digital converter board 4
9850 - 985F		Analog to digital converter board 5
9860 - 986F		Analog to digital converter board 6
9870 - 987F		Analog to digital converter board 7
9880 - 988F		Analog to digital converter board 8
9890 - 989F		Analog to digital converter board 9
98E0 - 98EF		Direct Connect Modem Board
98F0 - 98FF		Disk/Status I/O board
A000 - A7FF		U20 2 Kbyte EEPROM on processor board
A800 - A80F		U6 DUART on processor board
A900 - A90F		U16 VIA for IEEE488 and CW ID on processor board
AA00 - AA01		Front panel LCD
B000 - FFFF		U19 EPROM on processor board

* Locations 6000 - 97FF may be replaced with EEPROMs (X2816A) in some systems. This allows larger programs to be saved in EEPROM than can be saved in the "boot" EEPROM on the processor board. The program can be saved using SAVE

EEPROM 24576 or loaded using LOAD EEPROM 24576. In addition, if the RAM board is partially loaded with EEPROM, U6B on the processor RAM board needs to be left empty. On power up, Basic searches for the top of RAM. Leaving out U6B leaves a gap in the memory map insuring that the search for the top of RAM does not write over the EEPROM.

Theory of Operation 1441A Analog To Digital Converter Board

The analog to digital converter board selects external samples through reed relays and presents the selected sample to the analog to digital converter. The sample is measured and sent to the processor through the STD bus. In addition, processor instrucions can drive the control (Raise, Lower, Fail Safe, and Channel Select) outputs. The description of the circuit will be broken down by schematic page number for simplicity.

Page 1: STD Bus Interface, VIA

U02 (the Versatile Interface Adapter) is interfaced to the STD bus in somewhat standard fashion. U03 compares the address on the bus with that set up by the jumpers on P01. If these addresses agree and IORQ* (Input Output Request) is driven low by the memory map PROM on the processor board, BOARD-SEL* goes low. This enables the VIA U02 and the data transceiver U01. The direction of data transmission is determined by the STD-R/W* line on pin 1 of U01. This line is high when the processor board is attempting to read from an external device. If BOARD-SEL* is low, U01 will drive the STD bus with data provided by U02. If the STD-R/W* line is low, the processor is attempting to write to an external device. If BOARD-SEL* is low, U01 will take data from the bus and present it to U02.

In a similar manner, U02 watches BOARD-SEL* and STD-R/W* to determine if and in which direction data is to be sent. In addition, U02 uses PHASE 2 (STD-P2* inverted by U04A) to synchronize the data transfer with the processor. It also uses STD-A0 through STD-A3 (the four least significant processor address lines) to select which of the 16 registers in U02 is to be addressed.

U02 is programmed by the DRC software to generate an interrupt each time the A/D converter finishes a conversion. The IRQ* output of U02 goes low when U02 requests an interrupt. This is double inverted (giving sufficient drive to drive the bus) by U04B and U04C. Since U04 is an open-collector device, sections of R03 are used as pull-up resistors where necessary. A pull-up is not required on the STD-IRQ* since it is pulled up on the processor board.

The outputs of U02 drive the remainder of the circuitry. CHAN0 through CHAN9 are programmed high by the DRC program as necessary to select one of the ten channels of metering and control. DELAY-RES* floats high on power-up since the peripheral lines of U02 are set to input on reset. Once all initialization of the VIA is complete, DELAY-RES* is programmed low, enabling the 5 volt supply to the remainder of the board. This prevents all the reed relays and all Raise, Lower, and Channel Select output from being enabled on system reset.

FAILSAFE1 and FAILSAFE2 are driven by the DRC program as appropriate. If all failsafe requirements are being met, these two outputs are high, turning on the failsafe outputs. If a failsafe requirment is not being met (a required site is not responding), the failsafe outputs go low, turning off the failsafe outputs.

RAISE and LOWER are driven by the DRC software as needed to generate raise and lower control signals. These signals are NANDed with the channel select outputs to provide active low Raise and Lower outputs for each metering channel.

BUSY, BSY.CLK and POLARITY are inputs to U02. BUSY indicates that an A/D conversion is in process. U02 is programmed to generate an interrupt on the trailing negative edge of BUSY. The DRC program then takes the A/D data and sets up for the next conversion.

POLARITY indicates the polarity of the sample that has just been converted. If POLARITY is high, the reading is positive.

The A/D converter determines the digital conversion of the sample voltage by integrating the sample signal for 10,000 counts of a 125 KHz clock. It then counts clock pulses as it "de-integrates" a reference voltage until the integrator output crosses zero volts. During these two periods, the BUSY output of the A/D is high. If we count the clock pulses during the time that BUSY is high, and subtract 10,001 (an extra clock pulse sneaks in), we get the result of the A/D conversion.

The A/D board utilizes the counter in the VIA to count the pulses on PB6 (which is BUSY anded with CLOCK, hence BSY.CLK). On detecting the end of a conversion (by the interrupt generated by BUSY), the DRC program reads the counter in VIA, performs the required arithmetic, and reinitializes the counter. A little software trick here: The counter in the VIA counts down. By initializing the counter to 10,001, and having the A/D cause the counter to count down, the counter ends up with the resulting A/D conversion IF the reading is negative (10001-COUNT). If the reading is positive, the software takes the two's (binary) or ten's (decimal) complement to yield the conversion. On page five we'll see why this trick was used!

CB1 goes high at the beginning of a conversion and goes low at the end of a conversion. The negative edge generates an interrupt indicating that the conversion is finished. In addition, the VIA clocks the data on CB2 (which is open, so it floats high) into the least significant bit of a shift register on each positive edge CB1. At the start of a conversion, the shift register is cleared. At the end of a conversion, the software checks to insure that the shift register holds the number 1, and nothing higher. This insures that only one conversion has been accumulated in the counter. It is possible (when doing disk accesses, or other operations that leave interrupts disabled for relatively long periods of time), to miss the end of a conversion, resulting in an invalid conversion in the VIA counter. If this occurs, the shift register will have a number higher than 1, and the firmware will throw out the conversion. For this reason, CB1 <u>must</u> be left open or high.

<u>Pages 2 & 3: Channel Select Relays</u>

These circuitry on these pages selects one of the ten floating sample voltages and sends it to the A/D converter on page 5. The appropriate relay is driven by the appropriate section of U05 or U06 in response to a channel select signal from the VIA on page 1. The reed relays (K01 through K10) have internal spike suppression diodes to prevent high voltage transients that would appear when the relay is released. In addition, the high side of the relay coil is driven by +5-delay, generated by the circuitry on page 6. This prevents the reed relays from being activated until the system has been initialized.

R15, R16, R17 and R18 combine the switched samples to drive the A/D conveter (on page 5). These resistor networks provide some isolation between samples should a reed relay fail to release.

<u>Page 4: Control Line Drivers</u>

U14 through U22 are quad 2 input NAND gates with high current open collector outputs. One of the two inputs of each NAND gate is tied to the G input (an active high enable input). If, for example, CHANO is high, and RAISE is high, section A of U14 will pull RAISEO* low, driving external equipment. Lower commands are handled in a similar manner. The channel select lines are NANDed with a steady high (+5-DELAY after system initialization) to give the CHANOUT* outputs. These lines are pulled low when a particular channel is being accessed. These lines are typically used to drive tower select lines of antenna monitors.

U14 through U22 each include clamp diodes from the outputs to pins 2 and 7. These pins are tied to the CLAMP line, which has a 30 volt zener diode to ground (D03). These diodes and the zener conduct when the voltage on a control line exceeds 30 volts. This protects the output transistors in the chips from voltage transients from the external equipment.

Page 5: A/D Converter

U08 does the actual A/D coversion. Note that most of the devices on this page operate on a floating power supply provided by U07, which takes the +5 volt supply and converts it to a floating 12 volts.

The floating 12 volts is converted to a floating +/- 5 volts by D02, D01 and R09. U08 uses the +/- 5 volts (+FLOAT and -5-FLOAT) for its power supply. U10 operates on +FLOAT and FDIGITLGND, the floating digital ground. U10 operates on a net of 5 volts.

U09 is a temperature stabalized voltage reference. It contains a temperature controlled oven and a reference voltage circuit. The heater portion of U09 operates directly on the 12 volt output of U07 (+FLOAT and -FLOAT). The reference voltage generated by U09 (LM399H) is 6.95 volts. The top of this reference is connected to +FLOAT (the floating +5 volt supply). The low side of the reference (U09 pin 2) is a very stable 6.95 volts below the floating +5 volts. This 6.95 volt reference is divided down to 1 volt by precision resistors R06 and R07. R08 provides current through the reference from -FLOAT. We end up with about 6 volts dropped across R07 and 1 volt across R06. This gives us a precise 1 volt reference voltage across R07, with the high side being approximately at the floating digital ground, and the low side about 1 volt below that. The low side of the reference is connected to the analog ground and the negative sample input of the A/D (U08). This results in the analog ground (and the sample common mode) being about 1 volt below the digital ground, well within the common mode capabilities of the A/D.

The +SAMPLE and -SAMPLE are provided to the A/D from the sample selecting circuitry on pages 2 and 3. R20 (series) and R21 (shunt) form an optional voltage divider reducing the +SAMPLE..-SAMPLE signal to DIV SAMPLE..-SAMPLE. The A/D board is normally supplied with a jumper in the R20 position and R21 open. By adding these resistors (use ones with a very low tempco), samples higher than the maximum 2 volts the A/D will accept.

Other analog circuitry on page 5 includes the integrating capacitor C01, the auto-zero capacitor C02, the integrating resistor R05 and the reference capacitor C03.

During the sample integrate and reference de-integrate phases of the A/D conversion, C01 and R05 are used in a standard operational amplifier integrator. During the auto-zero phase, C02 is charged with a voltage to compensate for the offset voltages of the operational amplifiers and comparators in the analog to digital converter. Also during the auto-zero phase, C03 is charged to the reference voltage so that a floating reference is available for the reference de-integrate phase of the conversion. A floating reference is required since a different polarity of reference voltage must be applied to the integrator to cause it to integrate back towards zero output depending upon whether the polarity of the sample voltage caused the integrator output to go positive or negative during the sample integrate phase.

U10 is a high speed CMOS NAND Schmitt trigger (74HC132). This chip is operating on the 5 volts between +FLOAT and F-DIGITLGND.

U10A operates as an inverting Schmitt trigger oscillator at 120 KHz. C04 is charged and discharged between the high and low trigger points of the input of U10A through R10. This 120 KHz square wave provides the required clock to U08. It also is NANDed with F-BUSY (the floating busy signal) by U10D. The output of U10D (F-BSY.CLK*) drives the LED in opto-coupler U11 through current limiting resistor R13. U11 inverts the F-BSY.CLK* and "unfloats" it to

BSY.CLK. The VIA (U02 on page 1) counts these pulses to determine the A/D conversion.

U10B similarly drives U13 through R11. This unfloats F-BUSY to BUSY. The VIA uses BUSY to determine when a conversion has been completed.

U10C similarly drives U12 through R12. This unfloats F-POLARITY to POLARITY. The VIA uses POLARITY to determine whether the sample is positive or negative (a high indicates the sample voltage is positive).

Finally, R14 provides necessary pull-up resistors for U11, U12 and U13, since their outputs are open collector.

Use of the counter in U02 allows the complete A/D to float with only three opto-couplers. The alternative methods available to float the A/D would be to use about 11 opto-couplers to transfer the multiplexed BCD, busy and polarity signals, or to add circuitry to serialize the data.

Page 6: Output Connectors and Delayed+5 Generator

This page of the schematic shows the connections to the header connectors which then connect to the rear panel. Note that the numbering systems used on header connectors and D connectors (used on rear panel) are different.

R19 and Q01 take the +5-volts supply and provide +5-delay once the system has been initialized and the VIA has pulled DELAY-RES* low. +5-DELAY powers the reed relays and the control line drivers. These are not enabled until the system has been initialized, preventing all the reeds from being pulled in on reset.

Theory of Operation 1442B Processor Board

The 1442B processor board consists of the following portions of the DRC190 remote control system:

Processor
Dual Serial Port (1 for modem, 1 for RS232)
Parallel I/O (for modem control and keyboard scanning)
1200 Bit/Second half duplex modem
32K bytes of EPROM holding system program
2K bytes of EEPROM holding calibration and setup data
8K bytes of RAM holding temporary data, pointers, and Basic program
Parallel I/O providing IEEE488 port and CW ID for radio links
STD bus interface to remainder of system

We'll take the schematic in order of page number and give you an idea what's going on.

Page 1: Processor, Memory Map Decode, Control Decoder, Reset Generator

The processor receives a 3.6864 MHz clock from the DUART (dual universal asynchronous receiver transmitter) on the processor board. The NMI* (NonMaskable Interrupt) and IRQ* (maskable Interrupt ReQuest) lines are pulled up by R06 and R05. They are pulled low by I/O devices on this board and other boards when an interrupt is required. The processor (U01) divides the 3.6864 MHz clock down to 921.6 KHz, which is used as the system clock. This clock is called E or PHASE2 throughout the system. In data transfers throughout the system, data is latched by the receiving device on the trailing (falling) edge of E. E is a continuous 921.6 KHz clock. It continues even if the processor is not doing an external data transfer (this occurs if the processor is doing an internal calculation, such as calculating a relative address). To prevent invalid access to external devices, the processor provides a VMA (Valid Memory Address) line. This line is high if the processor intends to do a data transfer on this cycle of the E line. VMA is low if no transfer is to be done.

The processor also outputs the R/W* line. This line is high when the processor is reading data (inputting from an external device) and low when the processor is writing data (outputting to an external device). If there is no data access in this cycle, R/W* remains high.

The approximate timing of all the processor generated signals is:

0 nS Address and R/W* lines go to required state. VMA goes high if an external data transfer to occur in this cycle.
174 nS E goes high
244 nS Processor output data appears
418 nS Write data to external devices is valid
608 nS Read data from external devices must be valid
717 nS E goes low, latching data. E high for 543 nS
727 nS Read data from external devices can be released (hold time)
737 nS Address, R/W* and VMA lines go invalid
747 nS Write data to external devices goes invalid (hold time)
1086 nS Start at top of list again

U04 converts from the 65/68 bus (E, R/W*, VMA) to the Intel Bus (OE*, WE*). U04 is a 3 line to 8 line decoder with enables and active low outputs. When E is high, R-W* is high, VMA is high, and SYSRES (system reset) is low, The Y3 output of U04 goes low, driving the OE* (Output Enable) line for the rest of the system. For devices using the Intel control bus (the DUART and memory), the low OE* line causes these devices to put their data on the data bus. Similarly, if the same conditions are true except that R-W* is low, U04's Y2 output (WE* or Write Enable) goes low, causing the addressed device to latch the data then present on the data bus. The processor has written to the device.

SYSRES is included in the control decoding to prevent writes to the EEPROM should the system have under voltage. SYSRES is high immediately after power up, immediately after pushing the rear panel reset button, or if the power supply voltage goes below 4.5 volts. Operation of the processor with less than 4.5 volts can cause writes to locations that should not be rewritten (the EEPROM holding calibration and setup data). Including SYSRES in the control decoding of U04 insures that the EEPROM will not get any bad writes during processor shut down.

U02 is the memory map decoding PROM. Driven by the 5 most significant address lines, U02 breaks the 64 Kbyte address map into 32 blocks, each 2 Kbytes. Note that a Kbyte is considered to be 1024 bytes. U02 is enabled only when VMA* is low, which is when the processor is about to do an external data transfer. Chip Select signals are generated by U02 only when the processor is in an external data transfer cycle.

The outputs of U02 are the chip select lines (active low) to the various I/O and memory devices on the system bus. If VMA* is high, indicating that we do not want to select any devices, R01 and R02 pull all the chip select lines high, deselecting all devices.

The CS* lines select the below listed devices:

CS0*	On Board I/O
CS1*	Off board data transceiver
CS2*	U19 EPROM
CS3*	U20 EEPROM
CS4*	U21 RAM
CS5*	Not used
CS6*	Off board I/O (STD IORQ*)
CS7*	Off board memory (STD MEMRQ*)

The actual memory map is shown on the next page.

U03 further divides the 2 Kbyte block of address space selected by CS0* into 256 byte blocks. Each of these blocks is assigned to an I/O device on the processor board. These chip select lines are:

CS00*	DUART
CS01*	VIA for IEEE488 and CW ID
CS02*	Front panel LCD

DRC190 Memory Map PROM

System Address	PROM Address		PROM Data								
Hex	Hex	Octal	D7	D6	D5	D4	D3	D2	D1	D0	Selected Device
0000	00	00	1	1	1	0	1	1	1	1	On Board RAM
0800	01	01	1	1	1	0	1	1	1	1	On Board RAM
1000	02	02	1	1	1	0	1	1	1	1	On Board RAM
1800	03	03	1	1	1	0	1	1	1	1	On Board RAM
2000	04	04	0	1	1	1	1	1	0	1	Off Board RAM
2800	05	05	0	1	1	1	1	1	0	1	Off Board RAM
3000	06	06	0	1	1	1	1	1	0	1	Off Board RAM
3800	07	07	0	1	1	1	1	1	0	1	Off Board RAM
4000	08	10	0	1	1	1	1	1	0	1	Off Board RAM
4800	09	11	0	1	1	1	1	1	0	1	Off Board RAM
5000	0A	12	0	1	1	1	1	1	0	1	Off Board RAM
5800	0B	13	0	1	1	1	1	1	0	1	Off Board RAM
6000	0C	14	0	1	1	1	1	1	0	1	Off Board RAM
6800	0D	15	0	1	1	1	1	1	0	1	Off Board RAM
7000	0E	16	0	1	1	1	1	1	0	1	Off Board RAM
7800	0F	17	0	1	1	1	1	1	0	1	Off Board RAM
8000	10	20	0	1	1	1	1	1	0	1	Off Board RAM
8800	11	21	0	1	1	1	1	1	0	1	Off Board RAM
9000	12	22	0	1	1	1	1	1	0	1	Off Board RAM
9800	13	23	1	0	1	1	1	1	0	1	Off Board I/O
A000	14	24	1	1	1	1	0	1	1	1	U20 EEPROM
A800	15	25	1	1	1	1	1	1	1	0	On Board I/O
B000	16	26	1	1	1	1	1	0	1	1	U19 EPROM
B800	17	27	1	1	1	1	1	0	1	1	U19 EPROM
C000	18	30	1	1	1	1	1	0	1	1	U19 EPROM
C800	19	31	1	1	1	1	1	0	1	1	U19 EPROM
D000	2A	32	1	1	1	1	1	0	1	1	U19 EPROM
D800	2B	33	1	1	1	1	1	0	1	1	U19 EPROM
E000	2C	34	1	1	1	1	1	0	1	1	U19 EPROM
E800	2D	35	1	1	1	1	1	0	1	1	U19 EPROM
F000	2E	36	1	1	1	1	1	0	1	1	U19 EPROM
F800	2F	37	1	1	1	1	1	0	1	1	U19 EPROM

The only remaining circuitry on page 1 of the processor schematic involves U09, the reset generator. D02 and section 3 of R09 form a voltage reference providing about 0.7 volts on the inverting input of section A of U09 (IA-). R7 and section 2 of R09 form a voltage divider, dividing the +5 volt supply down to 0.8 volts. This is presented to the non-inverting input of section A of U09 (IA+). Since the non-inverting input is higher than the inverting input of comparator U09, the output (SYSRES*) is high, indicating we are not in a reset. C03 holds the non-inverting input low immediately after power up, presenting a reset to the system after power up. D01 insures that C03 is discharged immediately after the loss of power so that a reset is generated after every power loss. R08 provides positive feedback giving U09 hysteresis, insuring that even with a slowly moving input, the output of U09 will be a clean square waveform. Note that the rear panel reset button is connected across C03. Pressing the reset button simulates the loss of power supply, forcing the system into reset.

Page two: DUART, Keyboard Interface, RS-232 Interface, Clock

Page two of the 1442 schematic is centered on the DUART. The processor interface of the DUART is pretty standard. The address lines A0-A3 select one of 16 internal registers of the DUART. WE* and OE* cause the DUART to latch up data on the bus or to output data to the bus if the chip is selected (CS00* low). Sysres is normally low, but goes high when the system is reset. This resets the registers inside the DUART. U06 (the DUART) can request an interrupt by pulling the IRQ* line low. A timer in U06 is normally programmed to generate an IRQ* every 10 mS. Y01 works with the oscillator in U06 to generate a 3.6864 MHz clock. This is used by U06 to run the internal baud rate generators for the serial ports. The 3.6864 MHz signal is also bufferred by U05C and fed to the 6802 processor. The 6802 divides this frequency by four to arrive at the system clock (E or PHASE2).

Serial port A goes through U07 and U08, converting the data from TTL to RS232 levels, and the reverse. The RS232 data is presented to P02, which connects to J22 on the rear panel. Note that the pin numbers on J22 do not correspond with the pin numbers on P02 because of the different numbering methods for double row header connectors and D connectors.

Serial port B goes directly to the 1200 bit/second modem.

U06 also includes a parallel input port and a parallel output port. The input port serves the following purposes.

IP0 is driven by the KEY line. Between keyscans, the row lines of the keyboard are driven low. The column lines are pulled high by R03. If a key is pressed, one of the column lines will be pulled low by the short in the keyboard at the row/column intersection. This low will cause the output of U27A (key) to go high.

IP1 is driven by the demodulator portion of the modem. This line is high if the modem is receiving a carrier. This line is used to insure that modem data is accepted only when a carrier is present. It is also used in the contention avoidance system insuring that two DRC190s do not transmit data at the same time.

IP2 is driven by a 1 Hz signal from U28. U28 contains a very accurate crystal oscillator, the crystal, and a divider chain, outputting 1 Hz. The DRC190 program watches IP2 for a change. Based on this signal, the clock and calendar are updated.

IP3 through IP6 are driven by the column lines from the front panel keyboard. Once we have determined that a key is closed, the row ouput lines can be pulled low one at a time until a low shows up on one of the column lines. Once that happens, the low row and low column lines correspond to the row/column code of the position of the pressed key. The DRC190 program debounces this in software to insure the key closure is valid. It also converts the key code to ASCII for use elsewhere in the program.

U06 also has a parallel output port. This port serves these purposes.

OP0 drives the modulator portion of the modem. When this line (XMIT*) is low, the modem carrier is turned on.

OP1 drives U13C on page 6 of the schematic. When this line (SPKRCOM) is high, a reed relay (K01) moves the front panel speaker from the output of the speaker driving amplifier to the input of the line driving amplifier. The speaker then serves as a microphone in the intercom function of the DRC190.

OP2 drives U13B on page 6 of the schematic. When this line (LINE-EN) is high, reed relay (K02) connects the primaries of line driving transformers T03 and T04 to the line driving amplifier (U14). When the line is not enabled, the open primaries of T03 and T04 provide a high secondary impedance, preventing this unit from loading the line. This allows other units on the line to drive the line. This is a sort of audio "Tri-State" system.

OP3 drives U15 on page 7 of the schematic through D03 and R33. When this line (SPKRMUTE) is high, speaker driving amplifier U15 is driven into saturation, preventing audio from going through U15 to the speaker. This mutes the speaker. In addition, SPKRMUTE is pulsed low for 10 mS each time a key on the front panel keyboard is pressed. This gives an audible click each time a key is pressed, providing audible feedback.

OP4-OP7 drive the row lines on the keyboard. These idle low. When a key is pressed, a column line is pulled low, causing KEY, the output of U27A to go high. Once the key closure has been detected, OP4 through OP7 are pulsed low one at a time. When one of the row lines being low causes one of the column lines to go low, the key has been found. It is at the intersection of the low row and column lines.

This completes the description of the circuitry on page 2 of the 1442 processor board schematic.

Page 3: FSK Generator

This page covers the FSK generator. This is a simple application of the XR2206CP chip (U10).

C07 provides DC bypass required by U10.

The oscillator section of U10 oscillates at a frequency determined by C04 and R14 or R15. The selection of R14 or R15 is based on whether U10 pin 9 is high or low. If pin 9 is high (Mark condition), R15 is selected, causing U10 to operate at 2200 Hz. If pin 9 is low (Space condition), R14 is selected, causing U10 to operate at 1200 Hz.

The amplitude modulator in U10 is disabled (forced to 100% carrier) by gounding pin 1. A sine (rather than triangle) output is selected by setting R16 to 200 ohms.

Pin 3 is normally biased to +6 volts through R11, R12 and R10. C05 holds the junction of R11 and R12 at AC ground. R10 pulls pin 3 towards AC ground, reducing the audio level at pin 3. This serves as the output level adjustment.

U13F grounds pin 3 when no output is desired. This forces the output amplifier in U10 to cut off due to lack of bias. U13F has an open collector output, so it has no effect if the input of U13F (XMIT*) is low. Finally, the output amplifier in U10 takes the signal present on pin 3 and amplifies it by 1 (actually amplifies the current) and gives the output on pin 2. The DC component of this is removed by C06 and is presented to the line driving amplifier (on page 6) as TXAUDIO1.

Page 4: Input Summing Amplifier

The input summing amplifier is a simple unity gain summing amplifier that summs the communications line inputs from T01 and T02. The output is sent to the FSK demodulator (U12 on page 5) and the speaker amplifier (U15 on page 7) for use in the intercom mode.

R19, R18 and R17 determine the gain of the summing amplifier. R20 and R38 reduce the output of the summing amplifier down to a level suitable for the speaker driver amplifier (page 7), since the received audio is sent to the speaker in the intercom receive mode. C08 provides DC blocking between the input summing amplifier and the speaker amplifier.

C10 provides DC blocking between the input summing amplifier and the FSK demodulator. C10, when combined with the 20K input impedance of the XR2211 FSK demodulator, forms a high pass filter with a cut off frequency of about 600 Hz. This rejects any AC hum that may be present on the received audio.

Page 5: FSK Demodulator

Page 5 of the 1442 processor schematic is the FSK demodulator portion of the 1200 bit/second modem. This is a standard application of the XR2211CP demodulator chip.

The XR2211 is a phase locked loop FSK demodulator. The free-running frequency of the Voltage Controlled Oscillator (VCO) is set using C09, R22 and R21. The VCO normally free-runs at 1700 Hz.

The input signal from the summing amplifier (page 4) is applied to U12 pin 2. U12 amplifies this signal and applies it to two phase detectors (called the loop phase detector and the quadrature phase detector). The other input of each phase detector is driven by the VCO. The VCO signal fed to the quadrature phase detector is shifted by 90 degrees from the signal feeding the loop phase detector.

The preamp in U12 allows the chip to operate with receive levels between 10 mV RMS and 3 V RMS. This gives a minimum receive level of -37.8 dBm (600 ohm line).

The output of the loop phase detector appears on U12 pin 11. The internal resistance of this output along with C12 form the loop low pass filter, rejecting the double carrier output of the phase detector while allowing the 1200 bit/second data to pass around the loop. The filtered output of the phase detector goes through R26 to the VCO through pin 12. The value of R26 determines the loop gain, which sets the capture and lock range of the PLL. As the input signal changes frequency, the loop phase detector detects the changing phase relationship between the VCO and the input signal, generating a changing VCO control voltage, causing the VCO to track the input signal. Since the control voltage into the VCO is proportional to the frequency, the control voltage ends up being proportional to the incoming frequency. This is compared with a reference voltage (available on pin 10 of U12) with hysteresis provided by R27. In addition, R25 and C11 provide a data low pass filter, further reducing any 2 times carrier components present while allowing the 1200 bit/second data to pass.

The "sliced" demodulated FSK appears on pin 7 of U12. At this point, a Mark state (2200 Hz) is low, and a Space state (1200 Hz) is high.

The quadrature phase detector is used to detect the presence of a data carrier. This phase detector compares the incoming signal (U12 pin 2) with a 90 degree shifted output of the VCO. The resulting signal has two components. One component is an AC signal with a frequency of twice the VCO (the sum of the incoming signal and the VCO frequency). The other component (the "difference" frequency) is a DC component proportional to the amplitude of the incoming signal. The quadrature phase detector portion of U12 forms a synchronous AM demodulator. The output of the quadrature phase detector is available on pin 3 of U12. R23 determines the "gain" of the quadrature phase detector, while C10 filters the output of the phase detector, removing the double VCO component.

The filtered output of the quadrature phase detector is compared with the reference voltage (once again, available on pin 10) to determine if sufficient signal is present to indicate carrier presence. R24 provides positive feedback around the carrier detect comparator, providing hysteresis. This hysteresis insures that any double carrier frequency signal does not get through the comparator. The reference voltage is (V+/2)-650 mV. With our 5 volt supply, this works out to 1.850 volts. As long as the voltage on pin 3 is over 1.85 volts, U12 pin 6 (RX-CAR) will be high. It is quite interesting to watch the voltage on pin 3 with a DC coupled scope as the DRC190 operates. When the data carrier first comes up, the voltage on pin 3 approaches 5 volts, vastly exceeding the required 1.85 volts to indicate carrier presence. As the carrier is keyed (with FSK), a ripple appears in the voltage on pin 3. This is due to the slight delay in the VCO tracking the incoming signal, which results in a varying phase error out of the quadrature phase detector. As long as the minimum voltage of the ripple does not go below 1.85 volts, U12 will continue to indicate carrier presence. A solid carrier detect with keying is guaranteed with a -30 dBm input and typically achieved with -40 dBm.

The carrier detect portion of the FSK demodulator is quite critical to the proper operation of the system. Carrier detect is checked as each character of data is received from the modem. If valid carrier is not present, the data is thrown out and the modem receive routine is reset.

Page six: Line Driver, Speaker Driver, External TX Key

U14 of the PC1442 processor board acts as a simple summing amplifier. Most inputs use U14 as an inverting amplifier.

The CWID signal from the Morse code identifier is amplified by 5 (R29/R39) and presented to relay K02.

The TXAUDIO1 signal from the FSK modulator is amplified by 1 (R31/R29) and presented to relay K02.

When the SPKRCOM signal from the DUART is high, U13C activates relay K01. This switches the front panel speaker from the output of the speaker driver amplifier (COM-OUT) to the non-inverting input of U14. The speaker then acts as a microphone whose signal is amplified by 1001 (1+(R29/R30)) and applied to K02. C13 keeps U14 a voltage follower as far as DC is concerned, preventing excessive output offset voltage. R32 provides a bias path for the non-inverting input of U14 when K01 is not activated.

At this point, we can have one of various signals present at the input of K02 (LINE-OUT). When the LINE-EN output of the DUART (on page 2 of the schematic) goes high, U13B activates K02, connecting the output of U14 to the communications lines through T03 and T04. When K02 is released, the primaries of T03 and T04 are left open, presenting a high impedance to the communications lines. This allows other DRC190s in the system to drive the line withou contention. This forms a sort of audio "Tri-State" bus.

Finally, the output of U13B (LINE-EN*) is double inverted by U13D and U13E and brought out the rear panel of the DRC190 as TXKEY*. Note that U13 is an open collector device, so LINEEN** out of U13D is pulled up by section 4 of R28 on page 5 of the schematic. Since TXKEY* is low whenever the DRC190 is putting data on the communications line, this can be used to key an external UHF TRL transmitter. Due to transmit bring-up and receiver squelch delays, it is necessary to increase the site delay time from the normal 0.05 seconds to about 0.25 seconds when such a system is used. This is handled in the calibration and setup of the DRC190.

Page 7: Speaker Driver Amp

The speaker driver amp is an almost standard application of the LM380-8 (U15). The non-inverting input of U15 is driven by the output of the input summing amplifier (U11 on page 4) divided down by R20 and R38 (also on page 4). Further, C08 on page 4 blocks DC allowing the internal biasing network of U15 to operate.

The output of U15 passes through C15, removing the +6 volt DC component. R35 prevents C15 from discharging due to leakage when the speaker is not connected. The COM-OUT signal from C15 goes to K01 on page 6 where it is sent to the front panel speaker unless we are in the intercom talk mode (when the speaker is used as a microphone and the output of U15 is ignored).

Finally, the SPKRMUTE output of the DUART on page 2 drives D03, R33, R34, and C14. SPKRMUTE is at +5 volts when the speaker is to be muted (which is most of the time). This is divided down by R33 and R34 and applied to the inverting input of U15. This forces the output of U15 to ground, preventing any audio present on COM-IN from driving the speaker.

When SPKRMUTE is low, the inverting input of U15 drops to 0 volts, allowing the signal on COM-IN to be amplified by U15 and drive the speaker. Since SPKRMUTE is driven by a transistor in the DUART, it cannot actually go all the way to ground. It does, however, go close enough to ground to not exceed the "knee voltage" of D03, insuring that the voltage into the inverting input of U15 is indeed 0 when the speaker is unmuted.

SPKRMUTE is pulsed low for about 10 mS each time a key on the keyboard is pressed to provide audible feedback of the keyboard operation. This "clicks" the speaker as U15 is pulled out and then back into saturation. C14 provides some pulse shaping. Since the keyboard pulse is actually unmuting the speaker, any data present on the input to the DRC will be heard through the speaker during the speaker click. This causes the occasional chirp instead of click during keyboard operation.

Page 8: IEEE488, Display Connector, CW ID

This page of the schematic is built around U16 (with the exception of P04, the display connector).

P04 provides the required data and address lines to the front panel display module. In addition, a chip select line (CS02) is provided by U05E. U05E inverts the active low chip select (CS02*) from the on board I/O decoder U03 on page 1. Also, CS02* is inverted by U05E, NANDed with the processor clock E by U27B, and finally inverted by U05F. This signal (CS02.E) is applied to pin 14 of P04. P04 is a 16 pin socket that will accept cables from a variety of displays. Some displays (such as IEE Daystar) require 16 pin connectors. These displays have an active high chip select input that can be driven by the memory map decoding (inverted by U05E). Some displays (such as the Sonicor) require only 14 pins. These displays do not have a separate chip select input. The E input needs to be gated with the chip select line. This is handled by U27B and U05F. When a 14 pin display is used, the bottom two pins (pins 8 & 9) of P04 are not used. The display appears to be a standard I/O device to the processor. Control (display clear, cursor positioning, etc.) and display data are written to the display. Status of the display can be read by the processor. R37 and R36 provide a variable bias to the display to adjust the viewing angle. The multiplexed LCD has a limited viewing angle. Adjustment of R37 optimizes the contrast at the desired viewing angle.

The left side of U16 is the standard processor bus interface. In addition, the IRQ* output of U16 is tied to the IRQ* input of the processor on page 1, allowing U16 to request an interrupt.

The majority of U16 is devoted to the IEEE488 instrumentation port. DIO1 through DIO8 are the 8 parallel bi-directional data lines for the bus. These lines are bufferred by U17 and U18 before being presented to the rear panel connector (J25) through P05 and the associated cable.

The other IEEE488 lines (DataAVailable, NotDataACcepted, EndOrIdentify, InterFaceClear, ATtentioN, RemoteENable, ServiceReQuest and NotReadyForData) are similarly bufferred by U17 and U18. The HI-TALK output of U16 drives the direction select lines of U17 and U18, allowing the lines to be biidirectional.

At this writing (10 November 1985), the software for the IEEE488 interface has not been written. A more detailed description of the operation of this portion of the circuit will be written when it works!

U16 is also used to generate a 750 Hz tone for the Morse Code CW Identifier. U16 divides the system clock (921.6 KHz on the E line) to 750 Hz and presents it to PB7 (CWTONE). In addition, under software control, the CWTONE output is enabled and disabled as required to generate the Morse code identifier. The DC component of the 5 volt square wave CWTONE is removed by C20. R40 and C21 form a low pass filter, turning the square wave to a triangle wave. This amplitude of this tone is set by R41 and sent to the line driving amplifier on page 6.

Page 9: Memory

The three memory devices are put directly on the processor bus. Note that the 6802 bus does not have WE* or OE* signals, so these are generated by the control decoder U04 on page 1.

U19 is A 27256 EPROM (32 Kbytes). Note that not necessarily all of the chip is enabled. It can be brought up in 2 Kbyte blocks as necessary by the memory map PROM U02 on page 1.

U20 is a Xicor 2816A (or equivalent) Electrically Erasable Programmable Read Only Memory. Non-volatile memory device includes on chip high voltage generation, address and data latches, and timing required for programming. To the remainder of the system, it appears to be static RAM, except that the chip is disabled by its internal timer for 10 mS after a write. Because of this, an interrupt routine handles writes to U20. This chip is used to hold the label and units characters for each metering channel, the metering curve (linear, square law, etc.) and the calibration scaling factor for each metering channel. In addition, this chip holds the site delay, site number, maximum site number, fail safe site numbers, control enable site numbers, communications and terminal bit rates, and the Morse CW identifier. It also holds a small Basic "boot" program that is executed on power up. This program is saved to EEPROM using SAVE EEPROM and loaded on power up or using LOAD EEPROM.

U21 is a 8 Kbyte static RAM. It holds temporary data and pointers for the whole system. In addition, some memory is left over for a small Basic program for control and logging of stations. The system RAM is expanded by plugging an additional memory board into the system STD bus.

Page 10: STD Bus Drivers

U24 and U25 drive the STD bus with the addresses out of the processor at all times (no off board device can access memory on the processor board). U26 drives the control lines of the STD bus with the appropriate lines from the processor or derived from the processor.

U23 acts as a bi-directional data transciever, passing data to and from the STD bus. The direction of data transfer is determined by the R/W* line on pin 1. When R/W* is high, data is taken from the bus and sent to the processor. When R/W* is low, data is taken from the processor and sent to the STD bus. Pin 19 enables U23. All the data lines on both sides of the chip are in a high impedance state unless pin 19 is low. Pin 19 is driven low by CS1* from U02 on page 1 when the processor is accessing an off-board address. The rest of the time, U23 allows the STD bus to float, and allows other devices on the processor board to drive the processor bus.

Theory of Operation: PC1443A Interface

The PC1443A (revision A) interface board serves a couple of purposes. It handles the interface between the system and the various power supplies and it provides a couple of serial interfaces for operating the disc drives and status panels.

Power Supply Interface

The power supply interface portion of the PC1443 takes power from the main supply and feeds it to the system backplane. In addition, it charges the UPS battery and provides under-voltage shutdown to protect the battery from deep discharge.

The battery is charged by regulating 24 volts DC (taken across the +12VDC and -12VDC outputs of the main power supply) down to the required 14.00 volt charging voltage. This is accomplished by programmable regulator U01 along with R13, R02, and R03. R01 provides current limiting on the regulator.

The UPS module converts 14 VDC to about +/- 100 volts DC. This is connected to the filter capacitors of the off-line rectifier circuit of the main power supply. When the AC line is present, it charges these capacitors to approximately 160 volts. This back biases diodes in the UPS module, preventing it from providing any power. When the UPS module is idling, it draws about 50 mA from the 14 volt supply. The negative 14 volt input to the UPS module is routed through R11 on the 1443 board. When the UPS is idling, the voltage developed across R11 is not enough to cause Q01 to conduct, so the voltage at the adjust input of U01 is determined by the voltage divider action of R01, R02 and R03. This allows the 1443 board to charge the battery when the UPS is idling (the AC line is present).

Should the AC line fail, the voltage across the main power supply input capacitors drops to 100 volts, forward biasing the output diodes in the UPS module. The UPS module takes over providing DC to the primary switcher in the main power supply. This increased load on the output of the UPS module increases its input current to above 100 mA. This current causes sufficient voltage drop across R11 to cause Q01 to conduct, pulling down the adjustment input of U01. This drops the output voltage of the charging circuit, back biasing D01, preventing the battery from being charged when there is no AC line present. This prevents power from the battery being used to charge the battery. . . a losing proposition.

In an extended power failure, the battery will eventually be unable to provide sufficient power to keep the system running. When this occurs, the +12 volt output of the main power supply will fall, causing K01 to drop out, disconnecting the battery from the system. This prevents excessively deep discharge of the battery. When the AC line is restored, the +12 volt output of the main power supply returns, and K01 pulls in. The battery is then connected to the remainder of the circuit, allowing it to charge and back the system.

P01 of the 1443 board connects to the battery, the UPS module and the main power supply. In addition, it connects to the rear panel system reset button. The PBRESET* (Push Button Reset, active low) line is routed directly from P01 to the back plane through the 1443 edge connector. In addition, R04 provides current limited +5 volts to run the LED in the reset button.

The wiring of P01 is listed below.

P01 pin	Wire Color	Connects to
1	Brown	+5 volts from power supply
3	Orange	Common from power supply
5	Green	+12 volts from power supply
6	Blue	-12 volts from power supply
7	Violet	Battery +
8	Gray	Battery -
9	White	UPS + input
10	Black	UPS - input
11	Black/White	Reset switch
12	Black/White	Reset switch
13	Red/White	Reset switch LED +
14	Orange/White	Reset switch LED

Disc Interface

The remainder of the 1443 board provides interface to disc drives and status panels. Each of these uses a serial data bus. They are driven with software through U04.

P03 drives the Commodore serial bus, which is used to interface with a disk drive. U04 drives the bus through sections of U05 under software control. The Commodore serial bus signals are described below:

The ATN output of U04 is inverted and converted to an open collector output with a pull-up resistor by U05A and a portion of R07. The active low ATN* output is driven low when the DRC190 is sending an interface instruction to a disk drive or other device on the bus. These interface instructions tell specific devices to listen or talk on the bus. An actual bus transaction is described a bit later in this section.

The CLK output of U04 is inverted by U05B yielding CLK*, another open collector bus signal pulled up by another section of R07. The clock line is a bidirectional line that can be driven by the DRC190 or by the outside device. When the CLK* line is driven by an external device, its state is read by the PB2 input of U04. The DRC190 drives PB1 when it wishes to drive the clock line. When the DRC190 is to receive clock signals, PB1 is set low, allowing CLK* to go high. The external device is now able to drive the CLK* line to the desired state, which is detected on the PB2 input of U04.

In a similar manner, PB3 and PB4 of U04 and U05C drive the DATA* line and receive data off the line.

A request for a directory from the disk drive might appear as follows:

We will send a command to the disk drive (typically device 8, the primary address), with a secondary address of 0, telling it to listen to the bus. The command is typically expressed as MLA8,0 (My Listen Address 8, Secondary address 0).

Sending this command on the serial bus goes like this. Through the VIA on the disk drive board (U04), we force the serial bus ATN* line low, telling other devices on the bus we are about to send a command character. We then send the primary address over the bus.

To send a character over the bus, we do the following. We start with a bus handshake to insure that all the devices on the bus are ready to receive the character. This same handshake and character transmit sequence is used to transmit any character over the bus. Since the ATN* line is low, the devices on the bus know that this is a command character.

The serial bus "talk" handshake goes like this. The CLK* line is forced low and the DATA* line is released. The other devices on the bus should pull the DATA* line low. If this does not occur, the DRC190 indicates a serial bus error (SB error). The CLK* line is released. The DRC190 then waits for all devices on the bus to release the DATA* line. Since all devices are open collector, any device can hold the DATA* line low, indicating it is not yet ready to receive a character. When all devices are ready, DATA* is released, the DRC190 detect this, forces the CLK* line low, and starts transmitting the character.

A byte is sent over the bus following the talk handshake. The byte is sent in this manner. The least significant bit of the byte to be transmitted is placed on DATA*. If the LSB is 0, DATA* is driven low. If the LSB is 1, DATA* is released, and is pulled up by the pull up resistor. The CLK* line is pulsed high, then low, causing the devices on the bus to capture the bit. The next bit of the byte is put on the DATA* line, and the CLK* line is pulsed high. The byte is transmitted with no handshake between the bits. After the last bit (the MSB of the byte) has been transmitted, the DRC190 releases the DATA* line and waits for the external device to pull DATA* low. This "frame handshake" indicates that the 8 bits have been received properly. If the frame handshake fails, an SB error results.

We have just sent the primary address of the device we are telling to listen over the bus. We then send a secondary address of 0 (actually, $F0 is sent, indicating a secondary address of 0). The ATN* line is then released, indicating we are finished sending the command.

Device 8, the disk drive, is now listening to the serial bus. We transmit the character "$", which indicates we want the directory, without driving ATN* low, since this is data instead of a command. The $ is sent in a manner similar to the other characters sent over the bus, except that since this is the only data character to be sent, we flag it with an "EOI" (End or Identify, here we are using it to identify the end of the message). We use a talk EOI handshake instead of the previously discussed talk handshake. The two handshake sequences are the same except that when we release the DATA* line and wait for the listening devices to let the DATA* line go high, we do not immediately respond to the high DATA* by forcing the CLK* line low. Instead, we leave DATA* high. This "not pulling" CLK* low is recognized by the addressed device as an indication of EOI. It acknowledges the EOI by pulsing DATA* low. When DATA* is again released by the receiving device, the "talk EOI" handshake is complete. The character is then transmitted in the normal manner.

In going further in our directory command, the addressed drive is now told to "unlisten". The UNL command does not include an address, and it tells all devices to stop listening. The UNL command consists of a $3F sent over the bus with ATN* low.

The disk drive is then given permission to talk on the bus. This command (MTA8,0) is sent in the same manner as the previous MLA command (with ATN* low). The MTA command adds $40 to the primary address (yielding $48 for drive 8), and, as previously, adds $F0 to the secondary address.

Since the disk drive had just received a command, and is now expected to talk on the bus, a bus turn-around sequence is called. This turn-around sequence goes as follows: The DATA* line is forced low by the DRC190. The CLK* line is released. We wait for the addressed device to pull the CLK* line low (if this fails, an SB error results). This completes the bus turn-around, with the addressed drive ready to send data.

The DRC190 then receives a byte off the serial bus. The DATA* line is driven low by the DRC190. The CLK* line is released (although it is being held low by the disk drive waiting to talk). We then wait for the disk drive to release the CLK* line. On detecting this, we handshake with the drive by releasing the DATA* line. We then wait for the device to pull CLK* low. If this does not occur within 200 uS, the drive is sending an EOI, indicating that the next character to be sent is the last one in the message. If we detect EOI, it is acknowledged by pulsing DATA* low, then high. Finally, we wait for CLK* to go high, indicating the drive has placed a bit of data on DATA*. This bit (the LSB of the byte being received) is taken off the DATA* line. The bit is a 1 if DATA* is high. The DRC190 then waits for CLK* to go low, then high again, indicating the next bit is on the DATA* line. This repeats eight times to capture the 8 bits. The frame handshake occurs at the beginning of the next received character.

The first two bytes sent by the drive are thrown out, since they represent a load address that is not used in the DRC190.

The Commodore 1541 sends the directory in the same form as a Commodore Basic program (which is different from the DRC190 form) where the line number represents the number of block used by the file. In a line of Basic, the first 2 bytes represent a link to the next line of the program. On a 6502 based machine (such as Commodore or Apple), the link is in low byte, high byte format. On 680x machines, such as the DRC190, the link is stored in high byte, low byte form. The link is address of the next link in the program. During a line search (such as in a GOTO or GOSUB), the program jumps from link to link checking line numbers until the desired line is found. The use of links makes line finding faster, since each line need not be scanned.

The next two bytes of a Basic program represent the line number in low byte, high byte format for a Commodore, or high byte, low byte format in a DRC190 program. The line number is followed by the tokens and ascii characters of the program line, followed by a 00 byte, indicating the end of the line. The 00 byte is followed by the link of the next line. If the link has both bytes 00, the previous line was the end of the program. DRC190 Basic follows the 0000 link with another 00, indicating the end of a zero length line.

The directory command takes the data coming down the serial bus, evaluates it, and displays it. The evaluation consists of throwing out the first two bytes sent (the load address), then evaluating each line as it is sent.

As each line is received, the first two bytes represent the link to the next line. These are thrown out by the DRC190 in the directory command. The next two bytes represent the line number, which represents the size of the file about to be listed, in low byte, high byte form. These two bytes are reversed, converted to decimal, then displayed. Each byte from the serial bus is then sent to the display, to show the name of the file and its attributes, as sent by the disk drive. When a 00 byte is detected, indicating the end of a line, a carriage return line feed sequence is sent to the display. The next two link bytes are checked to see if they are 00, and thrown out if they are not. If they are 00, the directory listing is completed.

On completion of the directory listing, the DRC190 sends an UNT (untalk) command to the disk drives. The UNT command consists of a $5F with ATN* low. This takes all devices off the bus.

A "close file 0" command is then sent to the drive. This takes the form of MLA8 (my listen address 8), followed by a $E0 sent with ATN* low. Finally, an UNL command is sent.

This description should give you some idea of how communications with disk drives operate. For further information, refer to the bibliography section.

Status Interface

The status interface uses another serial bus, although this is much simpler than the interface to the disk drives. The disk drive interface needs to send and receive variable length messages to and from various devices on the bus. It includes extensive handshaking, since the disk drive cannot be constantly watching the bus (it has to look at the disk now and then). The status panel, on the other hand, transmits and receives constant length messages with no handshaking. The bus requires more wires, but is quite simple.

U04 contains an 8 bit shift register that can be used to serially transmit or receive data. In each case, this data is sent or received through the CB2 line. U04 sends the shift register clock out on CB1. This is used to drive the shift registers in the status panels.

When status is being sent by the DRC190 to a local status display, PB5 is set low, indicating we are doing an output. This causes U06A to release STATUS-I-O so that it can be driven by U04 CB2. CB1 (the status clock) is programmed low, the desired logic state is put on CB2 (the STATUS-I-O) by the DRC190, then CB1 is programmed high. The status panel receive shift registers capture the data bit on the positive edge of CB1 (which is inverted by U05F, causing the status panel shift register to capture the data on the negative edge of SCLK*). This is repeated until 5 bytes (40 bits) are sent to the status panel. The first 4 bytes are 32 channels of status. The last byte is the site number of the site the status information was received from. On completion of the transmission of these 5 bytes, LOAD-OUT is pulsed high (pulsing LOAD-OUT* low). The status panel routes LOAD-OUT* through a binary comparator that compares the from site number to the site number the particular panel is to display. If they match, the LOAD-OUT* pulse is passed on to the other receive shift registers, allowing them to latch the data that has been received. The latched data is used to drive the status panel front panel LEDs and drive the status outputs. If the address did not match, the shifted in data is not latched, and is ignored.

Once per second, the DRC190 reads in the current status of its associated transmit status panel (one DRC190 can have several receive status panels showing the status of various sites in the system, but only one can transmit). To read the status, the DRC190 programs U04 to treat CB2 as a shift register input and sets ST-IN-OUT* high, indicating we are about to do an input. U06A now routes the incoming status (from the transmitting status panel's transmit shift register) to STATUS-I-O. U04 pulses the LOAD-IN line high, causing the transmit shift register in the status panel to latch the current state of all 32 status lines. The status is then clocked out of the status panel shift register into the shift register in U04 in a manner similar to the shift out. Four bytes, representing the 32 status channels are shifted in. These are compared to the last status that was shifted in. If there is a difference, the new status is transmitted to all sites by the DRC190.

STD Interface

Interface to the backplane (the STD bus) is provided by U02, U03 and U07A.

U03 compares the address present on the address bus with the board address set up on P02 (normally $A8F0). If the compared address lines match AND STD-IORQ is low, Board-Sel* goes low, enabling the remainder of the STD bus interface. Note that STD-IORQ is driven by the memory map PROM on the processor board. It is low when the processor is addressing off board I/O devices.

If Board-Sel* is low, U02 is allowed to transmit data. The direction of the data flow through this non-inverting transceiver is determined by the STD-R-W* line. If the line is high, the processor board is trying to do a READ, and U02 will take the data present on the A side (the board side) and present it to the B side (the STD bus side). If the STD-R-W* line is low, the processor is doing a write. The data present on the STD data bus will be presented to the data lines of U04.

If Board-Sel* is low, U04 is also enabled. Otherwise, it ignores the bus activity.

U07A inverts STD-P2* (the inverted processor phase 2 line, also called the E line) to become Phase2. This is presented to the Phase2 input of U04.

During a write to U04, the processor sets up the address lines to represent an address of U04 ($A8F0 to $A8FF). A few nS later, the memory map PROM on the processor board drives STD-IORQ* low, indicating we are addressing an off-board I/O device. A few nS later, the Board-Sel* output of U03 goes low. At the same time the address lines were set up, the STD-R-W* line was driven low by the processor to indicate we are going to do a write.

About 170 nS after the address and R-W* lines were set up, Phase2 goes high. About 250 nS after this, valid data from the processor is presented to the data lines of U04. After the Phase2 line has been high for 540 nS, it goes low, causing U04 to latch the data that was present on its data lines. The result of this write into one of the 16 registers inside U04 depends upon which register was written to (determined by the address lines connected to the RS Register Select inputs of U04). U04 could just present the written data to one of its output ports (port A or port B), or could use the data internally to set a timer, load a shift register, or set up the operation of the timers, shift registers and ports.

A read operation from the processor operates in a similar manner. The only real difference is that that STD-R-W* line is now high, causing U04 to drive the data lines instead of receiving data off the lines. These lines then drive U02, which then drives the STD bus with the data.

The final portion of the STD-Bus interface is handled by U07B and U07C. If U04 is programmed to generate an interrupt, it pulls IRQ* low when the interrupt needs servicing. This is inverted twice by U07B and U07C to increase the drive capability to drive the STD bus. Since U07 is an open collector inverter, R07 is used pull up the lines where necessary.

Theory of Operation 1444 Subcarrier Transceiver

The 1444 subcarrier transceiver generates and demodulates subcarriers suitable for control and metering on STL and FM broadcast stations. The theory of operation will be covered by page number of the schematic.

Page 1: Subcarrier Modulator

U01 is a function generator that is set up to generate a sine wave at the desired subcarrier frequency. R01 adjusts the subcarrier waveform symmetry. R02 adjusts the triangle to sine wave converter to minimize the distortion of the subcarrier waveform. R01 and R02 are adjusted for minimum harmonic distortion of the subcarrier carrier, as measured with a THD analyzer or a spectrum analyzer.

Pin 1, the AM input, is grounded, forcing full subcarrier output.

R03, R05 and C02 provide a bias voltage at half supply. C02 places this bias voltage at AC ground. R04 provides the bias voltage to the input of the output amplifier while pulling that point towards AC ground to adjust the output level (subcarrier injection).

The subcarrier frequency is determined by C03, R06, R07, R08 and R09. Since R09 is small compared with the other resistors, it has little effect on the frequency. R06 is used to get close to the desired frequency while R07 sets the precise frequency. R08 prevents damage to U01 should R06 and R07 be set to zero ohms.

The audio to be carried on the subcarrier (TX-AUDIO) is fed to R10 and R09. The large division ratio of this voltage divider provides the required low level audio to give the desired deviation of the subcarrier (typically 1 KHz/volt). R09 is used to adjust the subcarrier deviation.

C04 removes the DC component of the resulting subcarrier.

Page 2: Subcarrier Receive Mixer

The subcarrier receiver or demodulator uses the super-hetrodyne principle. The entire base-band is up-converted to 455 KHz. At 455 KHz, the desired subcarrier is pulled out with a ceramic filter. Using an up-conversion places the image response of the receiver above 455 KHz, above the response available on STLs or FM receivers.

This page of the schematic covers the local oscillator - mixer (or converter) portion of the receiver.

As in the transmit portion of the subcarrier transceiver, U02 here generates a sine wave. R14 adjusts the triangle to sine converter in U02 to provide the sine wave. The frequency of the oscillator is determined by C06, R16, R17 and an AFC voltage from the discriminator on page 4. With no input signal, the AFC voltage is zero, allowing R17 to be adjusted so that the correct local oscillator frequency is measured on P05. Note that P05 normally has a shorting plug across it. This must be removed before measuring the LO frequency. After the LO has been measured, the jumper is replaced to improve the local oscillator null. Pin 11 of U02 is an open collector output of the oscillator signal. R15 provides pull up, making the test signal available on P05 a 12 volt P-P square wave.

The local oscillator frequency should be adjusted to 455 KHz - SCA where SCA is the desired receive frequency. For example, if we wish to receive 110 KHz, the local oscillator frequency would be 345 KHz. The mixer uses low side injection to insure that the required frequency is always within range of U02. In addition, the AFC circuit is designed with low side injection in mind. If the local oscillator were changed to high side injection without a change in the AFC circuitry, the AFC would push the receiver away from the desired frequency (not a good idea!).

U02 includes a balanced AM modulator (one capable of generating double sideband supressed carrier). R11 adjusts the DC bias to the input of the balanced modulator so that there is no output on pin 2 when no subcarrier is being received.

When a subcarrier is received, the AC voltage is analog multiplied by the local oscillator signal in the balanced modulator, resulting in a signal at SCA + LO and one at LO - SCA where SCA is the received subcarrier frequency and LO is the local oscillator signal. The SUM signal ends up at 455 KHz, while the difference signal ends up substantially below 455 KHz. C05 AC couples the received subcarrier into the balanced modulator.

R12 and R13 provide the required +supply/2 bias to the input of the output amplifier of U02.

The mixer output appears on pin 2 of U02. The DC component is removed by C08.

Page 3: IF Amplifier

The mixer output from U02 on page 1 (with the DC component removed by C08) is applied to the IF filter Y01 through R18. R18 along with the 600 ohm output impedance of U02 provides the required driving impedance for Y01. Y01 pulls the subcarrier out of the up-converted base-band while rejecting the rest of the base-band.

Q01 along with the associated resistors form the IF amplifier. The output of the IF amplifier is coupled to the discriminator through C09.

The discriminator is formed by R23, R24, D01, D02 and Y02. The DC (or audio) voltage on DISCRIM-IO varies with the frequency of the IF signal applied to DISCRIM-IO. R25 and C11 form a low pass filter, allowing DC and audio to pass while rejecting the IF frequency.

The output of the low pass filter is amplified by 10 by U03A. The output of U03A is coupled through C13 to the subcarrier demodulator output. The demodulated output is sent through a low pass filter formed by R32 and C16 to further remove any IF components.

The output of U03A is also filtered through R29 and C12 to remove the audio while leaving a DC voltage proportional to the frequency deviation from the center of the discriminator. This voltage is bufferred by U03B and applied to the local oscillator as an Automatic Frequency Control signal, insuring that any drift in the local oscillator frequency will not cause loss of the received subcarrier.

Page 5: Power Supply Filters

R30, R31, C14, and C15 filter the power for the subcarrier transceiver insuring that power supply and processor noise do not interfere with the operation of the subcarrier circuitry.

Theory of Operation 1445 Direct Connect Modem

The 1445 direct connect modem allows the DRC190 to communicate with data terminals or computers over dial up telephone lines. The board provides an auto-dial, auto-answer, 300/1200 bit per second modem. It also provides an RS-232 port and several TTL level I/O lines.

Page 1: DUART

This page includes the STD bus interface circuitry and the Dual Universal Asynchornous Receiver Transmitter.

U02 drives BOARDSEL* low if the STD-IORQ* line is low and the address on STD-A4 through STD-A7 matches the selected address set up on ADDRSEL4 through ADDRSEL7, as set up on P01. R01 provides pull-ups to +5-VOLTS when a pair of pins on P01 is left open.

BOARDSEL* enables U01 (the data bus transceiver) and U03 (the DUART) when the processor is addressing this board. STD-R/W* selects the direction of the data transfer. If STD-R/W* is high, the processor is doing a read and U01 inputs data on D0 to D7 (the on-board data bus) and outputs it to STD-D0 to STD-D7 (the STD data bus). This takes data from the DUART and sends it to the processor through the STD bus.

If the STD-R/W* line is low, the processor is doing a write and the data from the STD bus is sent to the data inputs of the DUART.

U03 is the DUART. STD-A0 to STD-A3 (the least significant address lines) are used to select registers in the DUART. STD-WR* is driven low by the processor board when it wants to write data to an external device (such as the DUART). STD-RD* is driven low by the processor when it wants to read data from an external device, such as the DUART. The STD-RD* and STD-WR* signals are ignored unless BOARDSEL* is low, indicating the processor wants to talk to this particular chip.

U07A inverts the STD-RES* signal, creating the active-high RESET signal, which resets the registers in U03 on power up.

The IRQ* output of U03 goes low when U03 wishes to interrupt the processor. This is doubled inverted and bufferred by U07B and U07C and put on the STD bus. U07 is an open collector device, so R03 provides required pull ups. The current DRC firmware does not use the interrupt capability of the modem card.

Y01 and C01 are used by the baud rate oscillator in U03. U03 contains programmable dividers that use the 3.6864 MHz signal to provide the different available communications rates.

P04 allows several of the DUART parallel input and output lines to be brought outside the DRC190 for user applications.

To program an output bit on P04, use the statements listed below:

Output Bit	Program High/Low	Statement
1	Low	POKE 39150, 2
1	High	POKE 39151, 2
2	Low	POKE 39150, 4
2	High	POKE 39151, 4
3	Low	POKE 39150, 8
3	High	POKE 39151, 8
4	Low	POKE 39150, 16
4	High	POKE 39151, 16
5	Low	POKE 39150, 32
5	High	POKE 39151, 32

6	Low	POKE 39150, 64
6	High	POKE 39151, 64
7	Low	POKE 39150, 128
7	High	POKE 39151, 128

To read the input lines from P04, use a PEEK(39149). ANDing the result of the peek with a mask will yield a 0 result if the selected bit is low, and a non-zero result if the selected bit is high. Examples appear below.

```
IF (PEEK(39149) AND 2) <> 0 THEN DISPLAY "Input 1 is high"
IF (PEEK(39149) AND 4) <> 0 THEN DISPLAY "Input 2 is high"
IF (PEEK(39149) AND 8) <> 0 THEN DISPLAY "Input 3 is high"
IF (PEEK(39149) AND 16) <> 0 THEN DISPLAY "Input 4 is high"
IF (PEEK(39149) AND 32) <> 0 THEN DISPLAY "Input 5 is high"
IF (PEEK(39149) AND 64) <> 0 THEN DISPLAY "Input 6 is high"
```

Input 0 of the DUART is driven by the HI-SPEED* output of the modem module. This line is used by the firmware to determine whether the modem is in the 1200 or 300 bit per second mode.

Output 0 of the DUART drives the SW1 input of the modem module, determining the number of bits sent. On reset, SW1 is high, causing the modem to sent 8 data bits (7 bits of ASCII with a mark parity bit).

Page 2: Modem & RS232 Interface

This page shows the modem module and the RS232 interface.

U04, the modem module, operates on +5 and -5 volts. R02 and D01 derive the -5 volts from the -12 volt supply. The serial output of the DUART is sent to the modem, and the serial output of the modem is sent to the DUART. Unused modem lines are left open. The HI-SPEED* output of the modem reflects the speed of the current call. This is sent to the DUART. Note, however, that HI-SPEED is changed only during a call. The DRC190 firmware initializes the modem at 300 bits per second on power up. It then changes the modem speed in response to MDMSPD statements. It also changes the speed of the DUART serial port in response to a control-N W status message from the modem, indicating it has received a call at the wrong speed, and is about to change speed.

The modem RESET line is pulsed high on system reset, resetting the modem to its default conditions. The telephone line connects to the tip and ring connections of P03, which appears on J21 on the rear panel.

The second serial port of the DUART is converted to RS232 levels by U05 and U06. This appears on the rear panel at J23.

For further information on the modem, see the modem manual, reprinted in the back of this manual.

Direct Connect Modem Operation

The H&F 1445 Direct Connect Modem card is built around the Cermetek CH1770 modem module. This FCC approved module provides 300/1200 bit per second modem functions meeting Bell 103 and Bell 212 standards. It also provides auto anwer and dialing. Dialing is available with either tone or pulse. The manual on the CH1770 is printed following this page.

The CH1770 communicates with the DRC190 through a serial port on the modem card. Data is sent to the modem with a PRINT #2, statement. Commands are sent to the modem with a PRINT #2, CHR$(14); statement. The CH1770 looks for CHR$(14), then treats the following characters as a command. The PRINT statement should not be terminated with a comma or semicolon so that a carriage return line feed sequence is sent to the CH1770 to terminate the command.

Data can be received from the CH1770 using INPUT#2, statements, or INKEY$(2) functions. Note that the INPUT#2 statement will wait for a carriage return to be received, which can prevent the DRC190 from taking other required actions. It is therefore suggested that most modem input be done using the INKEY$(2) function, and string inputs built using this function. After long periods of inactivity, the string building routine can be terminated.

The CH1770 sends unsolicited status messages on a change in its status. These are preceeded by a control-N. The DRC190 firmware traps these messages and stores the latest one in MDMSTAT$. The line feed carriage return sequence following the status message is not trapped, and reaches the INKEY$(2) or INPUT#2 function or statement. For this reason, it is suggested that upon noting that a call has been answered, the receive buffer be cleared by executing a loop containing INKEY$(2). Otherwise, a remaining carriage return may terminate an input statement.

MDMSTAT$ has a character in it with the following meaning:

D - Modem has disconnected due to remote disconnect or disconnect command

R - The modem has sensed the line it is connected to is ringing

A - The modem has answered an incoming call or has sensed that an outgoing call has been answered

N - An outgoing call has not been answered

Note that MDMSTAT$ holds the last received status character. It may hold an R if the line rang once, but is not currently ringing. It may hold an N indicating that the last call was not answered, but there is no status on the current call yet. Most MDMSTAT$ characters are due to unsolicited status messages received from the CH1770. The exceptions are that when an END command (control-N E) is sent to the modem, MDMSTAT$ is changed to a D, even though the CH1770 does not return a status message. In addition, the control-N W message does not ever appear in MDMSTAT$, INKEY$(2) or INPUT#2. This message indicates that the CH1770 has sensed that the remote modem is at the wrong speed, and the CH1770 is changing speed. This message is trapped by the DRC190 firmware, and changes the speed of the serial port driving the CH1770. It also updates the variable MDMSPD.

It is suggested that prior to starting a call an END command be sent to the modem, forcing MDMSTAT$ to D. You can then check for N or A for the current call.

MDMSPD indicates the current speed of the modem in hundreds of bits per second. It can be either 3 (for 300 BPS) or 12 (for 1200 BPS). MDMSPD can be read or written to. An example of a read is:

```
        IF MDMSPD=3 THEN GOTO 1234    :REM Skip long message if low speed
```

A write to MDMSPD changes the speed of the serial port running the CH1770 and runs the modem through a speed change training sequence. This is typically used to originate a call to a specific terminal at a specified speed. An example of such a write is:

```
        MDMSPD = 12                   :REM Change to high speed
```

A summary of the modem commands is listed below. For more detail, see the CH1770 manual, reprinted following this section.

ANSWER   PRINT #2, CHR$(14); "A"

Force the modem into answer mode. This might be used to change a voice call into a data call (answer a line that is not ringing), or to answer a ringing line before the ring counter reaches the programmed value.

BREAK   PRINT #2, CHR$(14); "B 9"

Send a break (long space) for a multiple of 250 mS. The example sends a break for 9 250 mS periods (2.25 seconds). Valid numbers to use in the break command are 1 to 9.

COUNT   PRINT #2, CHR$(14); "C 9"

Sets the ring counter. In the example, incoming calls will be answered in after 9 rings, while outgoing calls will be terminated if not answered in 13 rings (n+4). The COUNT command can use a number between 0 and 9. If 0 is used, the modem will not auto-answer calls. On power up, count is set to 2.

DIAL   PRINT #2, CHR$(14); "D 'TB(805)541-0200'"

This example will originate a call using tone dialing to H&F. The TB indicates use of tone dialing. ()-@ are place holding characters and are ignored by the modem. No spaces are allowed in the dialing string (other than the required one after the D). If the call is answered by a modem within the programmed number of rings, MDMSTAT$ will change to A. If the call is not answered in the required number of rings, MDMSTAT$ will change to an N.

PRINT #2, CHR$(14); "D 'PB(805)541-0200'"

This example will originate a call using pulse dialing.

PRINT #2, CHR$(14); "D 'TB541-6116B765538B(805)541-0200'"

This example places a call through an alternate long distance carrier. The B characters in the dialing string cause a 2 second pause, allowing for answer of the access line, and then allowing for account number verification. Up to 32 digits may be included in a number string.

PRINT #2, CHR$(14); "D 'TB541-0200' 9"

This example will use tone dialing to call H&F and will retry 9 times, or until the call is answered. The number of retrys can be between 0 and 9. There is a 2 second pause between retries.

PRINT #2, CHR$(14); "D 'TB541-0200Z'"

This example puts the modem in the sleep mode (zzzz) after placing the call. The modem stays off hook and does not bring up its modem call. This may be useful to autodial a voice call.

PRINT #2, CHR$(14); "D"

Redial the last number dialed and wait for answer.

PRINT #2, CHR$(14); "D 9"

Similar to above, but the call is retried 9 times. The number of retries can range from 0 to 9. There is a 2 second pause between retries.

END PRINT #2, CHR$(14); "E"

End or terminate the current call. This puts the modem on-hook and sets MDMSTAT$ to D.

ORIGINATE PRINT#2, CHR$(14); "O"

Force the modem off hook and into the originate mode. This could be used to convert a manually dialed voice call to a data call with the modem in originate mode (instead of answer mode). If no answer tone is heard from the remote modem within 17 seconds, MDMSTAT$ is changed to N. If an answer is heard within 17 seconds, MDMSTAT$ becomes A.

This list of modem commands should cover most requirements. There are a few other commands listed in the following documentation from Cermetek, but they must be used carefully, as the DRC190 firmware counts on the modem to be programmed in a certain manner. For example, using the NEW command to change the command character from control-N will prevent MDMSTAT$ or MDMSPD from being updated when required, as the DRC190 firmware traps control-N messages. The PROGRAM command can be used, as outlined in the Cermetek manual, but unsolicited status messages should not be disabled. The QUERY command can be used, although the character immediately following the command character (control-N) will be trapped by the DRC190 firmware (and thrown out, since it is not a valid character for MDMSTAT$) and not appear in the character string sent to INKEY$(2) or INPUT#2. RESET should not be used, as the DRC190 firmware goes through a power up initialization sequence to program the CH1770, and this will not be repeated when sending a RESET. The TEST mode can be run to check the modem operation, as outlined in the CH1770 manual. Use of the UNLISTEN command is not suggested. The ZZZ command can be used, if desired.

The manual provided by Cermetek on the CH1770 modem module is reprinted (with permission) starting on the next page. Note that our agreement with Cermetek requires modem problems be referred to H&F rather than Cermetek.

27-7

Cermetek
microelectronics

DATA MANUAL

CH1770

BELL 212A-TYPE 110/300/1200 BPS INTELLIGENT MODEM COMPONENTS

DOC #82 Printed in USA

SUMMARY OF CONTENTS

Contents Page

1.0 FEATURES

- Small Size, PCB mount
  2.54"x3.74"x.75"
- 110/300/1200 bps Operation, Bell 212A and 103 compatible
- FCC Part 68 Registered Telephone Line Interface (DAA)
- Serial Modem Command Interpreter
- Intelligent Command Protocol
- Auto/Manual Dialing
- Auto/Manual Answer
- Auto Speed Select
- Auto Parity Select
- 3-Dialing Procedures
  (Dial Last, Immediate, Repeat Dial)
- DTMF and Pulse Dialing
- Diagnostic Test Mode
- Voice/Data Operation
- Asynchronous Operation
- TTL Host Interface Levels With RS-232C Type Lines
- Power: +/-5V

2.0 GENERAL DESCRIPTION

2.1 Configuration

The CH1770 is a Bell 212A-type modem component that uses the latest in LSI technology to implement a highly intelligent 110/300/1200 bps modem component in less than 9 square inches.

The CH1770 employs resident firmware to control every function of the modem. The CH1770 masks this firmware directly onto its resident controller.

2.2 Interface

The CH1770 interfaces to the telephone line through a built-in FCC registered data access arrangement (DAA) that directly connects to the telephone line through a user supplied RJ-11C jack. Because the DAA partially powers itself from the telephone line's loop current, telephone line connection must be made for correct DAA performance. Terminating TIP and RING lines with just a 600 ohm resistor for testing purposes is not adequate since no loop current is provided. An 18 volt D.C. floating power supply in series with a 600 ohm resistor will provide a suitable test termination.

Since the CH1770 is entirely controlled through the exchange of asynchronous commands on its serial data lines, TXD and RXD, it can be easily software controlled through the host's UART/USART without requiring additional serial or parallel control ports. Commands may be sent at either 110, 300 or 1200 bits per second (bps) using an ASCII format. Status is returned serially to the host using terse ASCII messages, making them easily decoded by the host's application software. The serial data interface is implemented using RS-232C lines but at TTL levels.

2.3 Operation

The CH1770 supports 13 different host commands, enabling such functions as:

- Auto-dialing
- Auto-answer
- Echo or no echo of commands
- Modem test diagnostics
- and many more

Dialing can be commanded to use either DTMF tone or rotary pulse dialing. Pause characters can be used to direct the CH1770 to pause during dialing. This enables the CH1770 to dial through PBX's, which commonly require the dialer to pause for an outside line, after dialing 9, before continuing to dial the rest of the number.

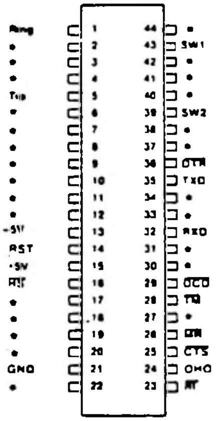
2.4 Data Speed and Parity

The CH1770 automatically adapts to the host's speed (110/300/1200 bps) and parity (odd, even, mark, and space) by using a simple learning sequence. If, however, a remote modem calls the CH1770 at a different speed and automatic speed adaption is enabled in the program register, it will automatically adapt to the remote modem's speed. The selected speed is indicated to the host at pin HS (high speed), and through a terse status message on RXD at the old speed.

3.0 MECHANICAL SPECIFICATION

3.1 Pin Configuration

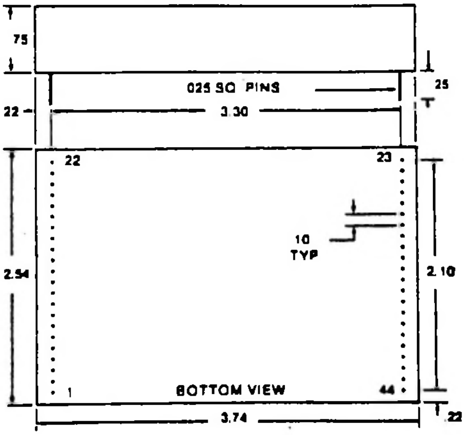
CH1770



NOTES: * Indicates a factory test point.
Make no connection to these pins.

3.2 Physical Demensions

CH1770



4.0 PIN DESCRIPTION

Telephone Line Interface:

RING; pin 1
TIP; pin 5 : direct connects to the telephone line's RING and TIP leads through a user-supplied RJ-11C telephone line jack.

OHO; pin 24 : OFF HOOK output. A high indicates that the modem is off hook.

Drive Capability: 3 LSTTL Loads

$\overline{\text{RI}}$; pin 23 : ring indication output. A low level indicates that the telephone line is ringing. The modem will answer at the end of the number of ring signals which is set in the "count" command (see Section 7.0). At power up the modem auto-answers after the second ring. A ring signal must be greater than 100 msec. in duration with greater than 500 msec. between rings.

Drive Capability: 3 LSTTL Loads

POWER

+5V; pin 15 : +5v power supply input
-5V; pin 13 : -5v power supply input
GND; pin 21 : signal and power ground return

MODE CONTROL LINES:

SW1; pin 43 Input SW1 is used by the CH1770 to set the component's serial data format as shown below:

SW1	Serial Format
LOW	9 data bits (including parity)
HIGH	8 bits (including parity) or 7 bits with 2 stop bits

The CH1770's asynchronous data format requires one start bit and at least one stop bit. For 7 bit data selection, 2 stop bits minimum are required. The number of data bits selected includes parity.

Input Load: 1 LSTTL Load.

SW2; pin 39 : Input SW2 is used to set the operation of the modem interface lines $\overline{\text{CTS}}$, $\overline{\text{DSR}}$, and $\overline{\text{DCD}}$. If SW2 is asserted low, these lines are asserted to the state of the $\overline{\text{DTR}}$ input, pin 36. If SW2 is forced high, normal RS-232C line signal sequencing is supported (see CH1770 Handshake Timing Diagram).

Input Load: 1 LSTTL Load.

SERIAL HOST INTERFACE:

TXD; pin 35 : serial transmit data input. Marking or a binary 1 condition is transmitted when high is asserted.

Input Load: 1 LSTTL Load

RXD; pin 32 : serial receive data output. Received marking or binary 1 condition is indicated by a high output.

Drive Capability: 2 LSTTL Loads

$\overline{\text{MR}}$; pin 26 : data set ready output. A low output on this pin indicates the modem is OFF HOOK in the data mode. If $\overline{\text{MR}}$ is set to follow $\overline{\text{DTR}}$, this pin will indicate when $\overline{\text{DTR}}$ has been asserted ON.

Drive Capability: 2 LSTTL Loads

$\overline{\text{CTS}}$; pin 25 : clear to send data output. When this signal is set low the CH1770 has set up the data call and is ready to transmit data.

Drive Capability: 2 LSTTL Loads

$\overline{\text{DCD}}$; pin 29 : receive data carrier detect output. When this output is set low, the received data carrier is present on the telephone line.

Drive Capability: 2 LSTTL Loads

$\overline{\text{DTR}}$; pin 36 : data terminal ready input. This input must be set low before the modem can answer or initiate calls. Once a call has been established this line can be used to disconnect the call by setting $\overline{\text{DTR}}$ high for greater than 50ms.

Input Load: 1 LSTTL Load

$\overline{\text{TM}}$; pin 28 : test mode output. This output is set low whenever the CH1770 is placed in the analog loopback test mode.

Drive Capability: 2 LSTTL Loads

$\overline{\text{HS}}$; pin 16 : high speed select output. A low on this level indicates the CH1770 is operating at 1200 bps. If $\overline{\text{HS}}$ is high, the modem is operating at 110 or 300 bps.

Drive Capability: 3 LSTTL Loads

MISCELLANEOUS SIGNALS:

RST; pin 14 : CH1770 reset input. A high applied on RST resets the modem to the idle state and asserts the phone line off hook. At power up, this pin must be asserted high for a minimum of 10 ms after the 5 volt supply has reached its operating region.

Input Load: 2 LSTTL Loads

Note: This pin internally has a 10K ohm resistor connected to GND and a 10uf capacitor connected to +5 volts.

5.0 TYPICAL APPLICATIONS

Because of the CH1770's small size and component configuration, it is ideally suited to integrate modem communications into a host data product. The addition of communications to such products as CRT terminals, personal or business computers, or workstations, can vastly expand the capability of the product, allowing it to support electronic mail, data base access, remote diagnostics, distributed networking and other such functions.

As can be seen in Figure 1, the CH1770 provides an efficient solution to 212A-type modem integration. Since its serial host interface supports RS-232C type lines at TTL levels, the CH1770 directly interfaces to virually any UART. All that is needed to complete the modem integration function is to provide power to the CH1770 and connect it to the telephone line through a modular telephone jack (RJ-11C-type). The telephone line interface is FCC registered and a registration label is included with each CH1770 for application to the outside of the host product. No recertification is necessary.

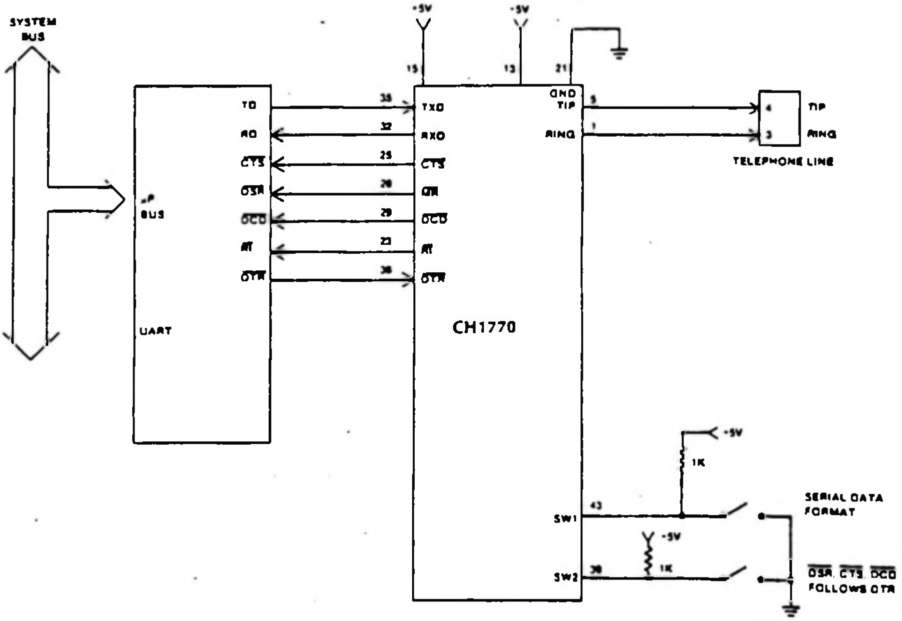
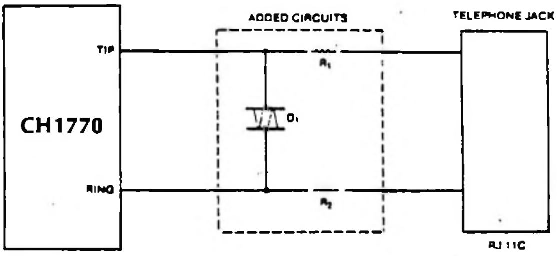


FIGURE 1: TYPICAL INTEGRAL 212A-TYPE MODEM

The telephone interface type is permissive, which means the product user is permitted to make the telephone line connection through a modular RJ-11C-type telephone jack. The FCC, however, requires that the end product user be provided with the installation rules and regulations from their Part 68 so it is necessary to have such information as presented in SECTION 8.0 (FCC GENERAL INFORMATION) in the end product's User's Manual.

The CH1770 can additionally be approved for Canadian telephone line connection. This must be done after the modem is installed in the host. The host system must then be submitted to Canadian DOC (Department of Communications) for approval. Because the DOC requires additional protection, the following additional telephone line interface circuitry, (shown in the dotted box), is needed. This circuitry is optional for FCC Part 68 registration in the U.S.A. (see Figure 2).



WHERE: R1, R2 = 10 ¼W CARBON FILM RESISTORS
D1 = 250V VARISTOR (PANASONIC ERZ-C10DK391)

FIGURE 2: ADDITIONAL CIRCUITRY NEEDED FOR CANADIAN DOC REGISTRATION

The integral modem of Figure 1 is controlled by the host product through the exchange of serial asynchronous commands, as detailed in SECTION 6.0 (MODEM CONTROL). Not all the CH1770 host interface lines need to be used. All that is needed by the CH1770 is transmit and receive serial data: TXD and RXD. All the other lines can be left unconnected, except $\overline{\text{DTR}}$ which must be low or ON before the modem can operate. This allows the CH1770 to adapt to virtually any UART environment.

The SW2 input allows further specification of the operation of the interface lines: CTS, DSR and DCD. When SW2 is asserted high, these lines follow the normal EIA-RS-232C specified handshake format. When SW2 is asserted low, however, CTS, DSR and DCD follow the state of the DTR input. This unique option allows the CH1770 to operate with the most stubborn "smart" terminals. Many times "smart" terminals (and also the IBM PC) insist on CTS, DSR and DCD all to be ON before they will enable the serial data interface. After a data call is set up, all of these lines are indeed ON so serial communication can take place. Since none of these lines are ON before a call is set up, the host's serial data channel is disabled, therefore making it impossible to send serial commands to the CH1770 to auto-dial, for example. This is also a common problem experienced by intelligent standalone modems. The CH1770 option to make CTS, DSR and DCD follow DTR solves this problem.

The CH1770 can also implement a stand alone 212A-type modem. Figure 3 displays the minimum configuration modem. The entire standalone modem consists only of the CH1770, some RS-232C level converters, and LED an display driver.

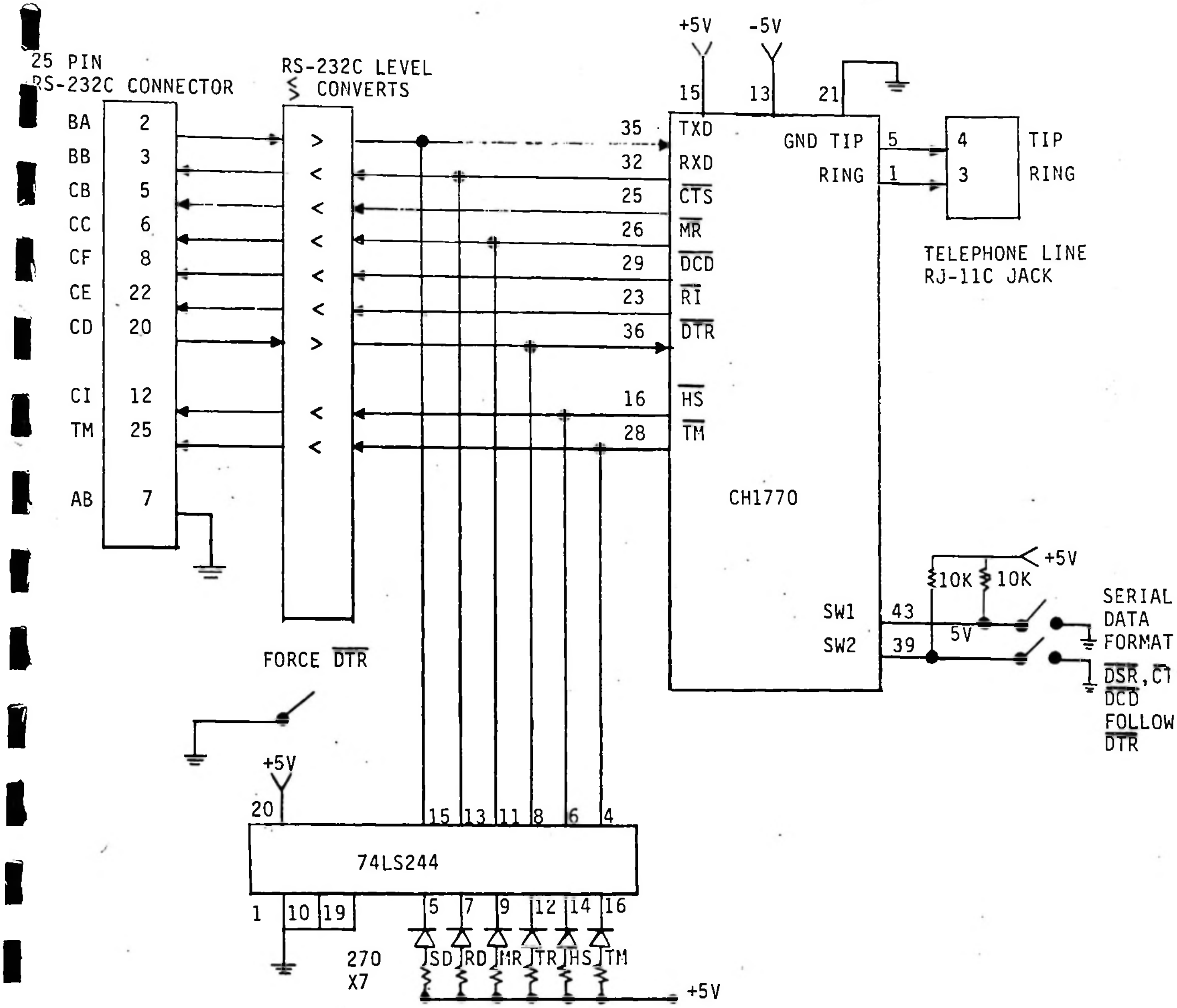


FIGURE 3: INTELLIGENT STANDALONE 212A-TYPE MODEM
(Minimum Configuration)

6.0 MODEM CONTROL

The CH1770 supports serial asynchronous communication. Before a call has entered the data mode, the CH1770 enables a built-in asynchronous command interpreter that allows the host product to issue modem commands serially over the transmit data line, TXD. Similarly, the CH1770 returns its status to the host over the receive asynchronous data line, RXD.

If the CH1770's $\overline{\text{DTR}}$ line is asserted ON (low) and no call is in progress, the command interpreter LISTEN's to data sent by the host in an effort to decode a command. If a command is recognized, it is immediately executed and a completion status is returned to the host.

After a data call is in progress, modem commands and data are sent over the same interface. It is therefore important for the CH1770 to understand when it should interpret the serial transmit data for commands. Use of the UNLISTEN command allows this discrimination.

6.1 Choosing Speed and Parity

The host can set the speed and parity settings of the CH1770 by sending it a four (4) character training sequence. The current speed is indicated by the CH1770 on output pin, $\overline{\text{HS}}$.

For the CH1770 to adapt to parity or speed it must first be idle. It cannot be in the midst of a data call or currently executing a command.

The $\overline{\text{HS}}$ pin is valid only during a connection. This pin is normally only used after answering a call since the modem adapts to the speed of the calling remote modem which is unknown to the local host.

The CH1770 has both a high and a low speed data channel. The high speed channel is set at 1200 bps, whereas the low speed channel is either 110 or 300 bps.

To set just the speed of the modem (110, 300, or 1200 bps), the host should send the CH1770 a <space><space> sequence. This will allow speed selection of either 110, 300, or 1200 bps. If both speed and parity are to be changed, the following sequence should be sent: <space><space>XY. X and Y must be upper case, and <space> is the character sent when the keyboard's space bar key is typed (decimal ASCII code 32).

Each training character sent must be followed by a 200 millisecond minimum pause where no characters are sent.

TRAINING CHARACTER	DESCRIPTION
<space><space>	adapt to host speed
<space><space>XY	adapt to host speed & parity

The CH1770 uses the trained speed to originate calls. On answer calls, however, the CH1770 adapts to the remote modem's speed. Answer speed selection is made between either high speed (1200 bps) or low speed (110 or 300 bps). The selection of either 110 or 300 bps in the low speed channel is controlled by the last low speed trained. For example, if the CH1770 had previously been trained to 110 bps, the modem will auto-select the answer speed at either 110 or 1200 bps. If 300 bps was the last trained low speed, as is the default state at power up, the modem will auto-answer select between 300 and 1200 bps.

6.2 Serial Line Protocol

All commands issued to the CH1770 are encoded in ASCII and are preceded by a single command character, <com>. Each command line is terminated by a carriage return, <CR>. At power-up <com> defaults to control N (decimal ASCII 14). It may be redefined, however, using the NEW command (see Section 7.0).

Multiple commands can be placed on one line separated by commas. The commands themselves consist of the command character followed by the command word, a delimiter, all arguments, and then the closing carriage return or comma. The maximum length of any command line, however, is 40 characters. If multiple commands are issued on a single line, only the first command shall be proceeded by a command character.

Two examples of commands would then look like the following:

```
<com>DIAL 'TB(408)555-1010'<CR>
<com>DIAL 'PB(408)555-1010',QUERY<CR>
```

In both cases, the notation <com> is equivalent to the single command character. The delimiter separating the command from the arguments is always a space. Only the first character of the command itself is significant. All remaining characters are ignored up to the first space following the command.

If the argument is not given in the command, the command will assume the default value as the argument. The arguments are all ASCII numbers and/or characters. The numbers themselves are ASCII encoded hex ('0' - '9' and 'A' - 'F').

Commands can be aborted while in progress by sending the CH1770 another command. The CH1770 will abort the current command upon receiving the new command's <com> character and then begin executing the new command after receiving the complete command line.

The command character itself can be transmitted by sending it twice in a row:

<com><com>

This would send the character a single time, if the modem is in the middle of a data call. There are two other methods of transmitting the command character. The first is to change the command character to another character and then transmit the former command character. The second way is to place the modem in the UNLISTEN mode and then transmit the character.

The CH1770 absorbs all commands without sending them on through to the telephone line.

The CH1770 returns its status to the host over the receive data line, RXD. All status messages are framed as shown below:

<com><status character><LF><CR>

A command character, <com>, precedes each message to let the host know that this is a status message and not data from the remote modem.

Two types of status messages exist; 1) Solicited, and 2) Unsolicited. Messages that result from the execution of a command are called solicited. These messages generally provide information about command completion.

Unsolicited messages result from external events such as telephone line ringing, telephone line hang-ups due to loss of carrier, and auto-answer line connection. On power-up or after execution of a RESET instruction, the unsolicited status messages are disabled. They may be enabled through the use of the PROGRAM command (see Section 7.0). Exact status message format is detailed in SECTION 7.0 (DETAILED COMMAND AND STATUS DESCRIPTION). The symbols <CR> and <LF> represent 'carriage return' and 'line feed,' respectively.

27-23

6.3 CH1770 Command Summary

COMMAND	DESCRIPTION
<com>Answer<CR>	force off-hook and answer call
<com>Break n<CR>	send break n x 250 msec
<com>Count n<CR>	ring and ringback counter 0 ignore ring signal, no auto answer 1-9 answer after n rings give up dialing after n+4 ringback signals
<com>Dial s<CR>	dial last, immediate, or until answered
<com>End<CR>	hang up
<com>New n<CR>	set new value of command character <com> to n
<com>Originate<CR>	force OFF HOOK and enter originate data mode
<com>Program n<CR>	set internal modem options
<com>Query<CR>	return modem status
<com>Reset<CR>	reset modem options to defaults
<com>Test n<CR>	start/stop the modem test
<com>Unlisten n<CR>	set CH1770 to LISTEN or UNLISTEN to commands during data transmission
<com>Zzzz<CR>	make modem quiet

6.4 CH1770 Status Summary

<com>A<LF><CR>	data call answered
<com>D<LF><CR>	modem disconnect
<com>N<LF><CR>	no answer or command execution failed

<com>R<LF><CR>	ring signal received
<com>W<LF><CR>	modem answer but host is at wrong speed
<com><LF><CR>	command complete acknowledgement
<com>?<LF><CR>	command entry error
<com><DIALED NUMBER><LF><CR>	number dialed status
<com><$H_1 H_2$><LF><CR>	$H_1 H_2$ represent hex status of the program register

6.5 Disconnect Sources

The CH1770 can be disconnected from a number of different sources once a data call has been established. The following events will disconnect a call:

- received long space (optional)
- END command
- $\overline{\text{DTR}}$ asserted off (high)
- RESET command
- hardware reset, pin 14

7.0 DETAILED COMMAND AND STATUS DESCRIPTION

To concisely describe the CH1770's commands and status messages, a few symbols will be used as defined below:

SYMBOL	DESCRIPTION
<com>	command character. Defaults to control N at power-up, but may be set to another character with the NEW command.
<LF>	line feed
<CR>	carriage return
[ ]	optional parameter
[ ]^n	optional parameters that may occur 0 to n times
n...m	specifies the inclusive set n through m
<SP>	space
<letter>	A...Z a...z
<command>	command including argument if needed
<n>	number
<quote>	single quote or apostrophe
<number>	dialed number string

Each CH1770 command follows the following syntax:

```
<com><command>[,<command>]<CR>
```

Single or multiple commands may be given to the CH1770 as described in SECTION 6.0 (MODEM CONTROL) but each command line must fit within the 40 character command buffer. If this buffer is exceeded, an entry error is returned to the host:

```
<com>?<LF><CR>
```

The following command description will detail the correct syntax, function and an example for each command.

ANSWER

SYNTAX: <com>A[NSWER]<CR>

FUNCTION: Takes the CH1770 OFF HOOK, pauses nominally 2 seconds, then sends answer tone (2225Hz) on the telephone line.

If after nominally 17 seconds the originating modem has not completed the handshake connect sequence, the call is aborted, returning the status:

<com>N<LF><CR>

If the calling modem completes the handshake connect sequence and is at the same speed as the local CH1770, the following status is returned to the host:

<com>A<LF><CR>

If the handshake sequence is completed but the calling modem is at a different speed (110/300 or 1200 bps), the CH1770 returns the following status at the host's old speed:

<com>W<LF><CR>

then switches to the other speed. The CH1770 can be optioned to disable automatic speed, however. After returning the W status it maintains its speed setting and waits for a host command, such as END. This enables the host, if unable to switch speeds, to command an END to the connection before the speed switch is made (see the PROGRAM Command).

The same CH1770 operation occurs if a call is auto-answered, except the answer sequence is initiated automatically by telephone line ringing instead of from the ANSWER command. The same status messages result but because the answer operation was not initiated by the local host, these messages are classified as unsolicited and therefore only occur if unsolicited status messages are enabled (see the PROGRAM Command).

EXAMPLE: <com>ANSWER<CR>
or <com>answer<CR>
or <com>A<CR>
or <com>a<CR>

BREAK

SYNTAX: <com>B[REAK][<SP><n>]<CR>

Where <n> is 1...9

FUNCTION: Sends a break (long space) condition for n times 250 milliseconds. (If the BREAK command is given without an argument, an argument of 1 is assumed.) After the break is complete the following status is returned:

<com><LF><CR>

EXAMPLE: <com>B<SP>1<CR>

Sends a 250ms space

COUNT

SYNTAX: <com>C[OUNT]<SP><n><CR>

Where <n> is 0...9

FUNCTION: Sets the RING and RING-BACK counter. Incoming calls are auto-answered after n RINGS whereas auto-dialed calls give up waiting for answer tone after n + 4 RING-BACK tone cycles. If n is specified as 0, the CH1770 will not auto-answer calls. The completion of the command is signified by the returned status:

<com><LF><CR>

The counter power-up default count is 2.

EXAMPLE: <com>C<SP>4<CR>

The CH1770 auto-answers after the fourth RING and gives up auto-dialing after the eighth.

DIAL

There are 2 basic variations of the dialing command: dial last, and dial immediate. All dialing commands can be directed to dial using either DTMF tone or rotary-type pulse dialing.

27-28

Each dialing variation refers to a sequence of numbers (hereafter referred to as a "number string") as a source for the number to be dialed. For immediate-type dialing commands the number string is supplied with the command, whereas the dial last command refers to a number string previously entered in the CH1770.

Along with telephone number digits the number string can also contain control characters that direct the CH1770 to dial using tone or pulse dialing. Also, pause or wait characters can be inserted that enables tandem dialing through PBX's.

Each number string can be 32 digits long. If control characters are used, fewer numbers may be entered. Each digit occupies one number string position, whereas control characters occupy 2 or 4 positions.

The following characters are allowed in the number string and occupy the indicated number string positions:

Character	Number String Positions	Meaning
0...9	1	dialed digits
TB	4	dials the digits in the number string using DTMF tones.
PB	4	dials the digits in the number string using rotary-type pulses.
B	2	inserts a 2 second pause in the dialing sequence.
Z	2	if placed as the last character in the number string, the CH1770 terminates the dialing command without going into the originate data call mode. The CH1770 stays OFF HOOK with its modulator squelched.
@	2	place holding characters
)	1	
(	1	
-	1	
<SP>		spaces are illegal

27-29

The following examples are typical number strings:

Number Strings	Meaning
TB767-1111	dials 767-1111 using tones after waiting 2 seconds.
PB767-1111	dials 767-1111 using pulses after waiting 2 seconds.
PB767-1111Z	same as the previous number string except after dialing 767-1111 the modem does not go into the originate data mode. It stays OFF HOOK with its modulator squelched waiting for the next command. This is very useful for placing voice calls.
TB7771234BB12345678408767111199	dials 777-1234 using tones, then a 4 second pause is inserted before the number 12345678 is dialed. Finally, the number (408)767-1111 is dialed, followed by a two digit access code, 99. This is a typical number string used for calling through long distance carrier facilities.

After a dialing command is given to the CH1770, the appropriate number string is interpreted to determine how the dialing process should proceed. As the number is dialed, the dialed number string is returned to the host in the form of a status message:

<com><NUMBER><LF><CR>

This enables the host to follow the progress of the number dialed. If the number string is terminated with Z however, no further action is taken by the CH1770. The modem is left OFF HOOK with the modulator squelched. This method of dialing is important if dialing is intended to reach a non-modem party.

After the completion of dialing, the CH1770 monitors the telephone line for modem answer tone.

A 'no answer' status message will be returned if the answer tone or some other timer reset signal is not received prior to timeout.

The following status messages are returned immediately after the dialed number message:

Call Progress Status	Meaning
<com>A<LF><CR>	data call answered
<com>N<LF><CR>	no answer, results from no modem answer tone within a 17 second period.

A. Dial last:

SYNTAX: <com>D[IAL][<SP><n>]<CR>

Where <n> is 0...9

FUNCTION: Dials the last number dialed and if <n> is specified, the CH1770 will retry the number n times or until answered. There is a 2 second pause between retries.

EXAMPLE: <com>D<CR>

Dials the last number dialed

<com>D<SP>5<CR>

Dials the last number dialed and retries up to 5 times if the call is unanswered.

B. Dial Immediate:

SYNTAX: <com>D[IAL]<SP><QUOTE><NUMBER><QUOTE>[<SP><n>]<CR>

Where <n> is 0...9
<NUMBER> is a telephone number of 32 digits or less, including control characters.

FUNCTION: Dials the telephone number specified in the command and if <n> is specified, the CH1770 will retry the number n times or until answered. There is a 2 second pause between retries.

EXAMPLE: <com>D<SP>'TB761-1111'<CR>

Dials 767-1111 using tone dialing.

END

SYNTAX: <com>E[ND]<CR>

FUNCTION: Ends the call in progress. The following status is returned to indicate command completion:

<com><LF><CR>

NEW

SYNTAX: <com>N[EW]<SP><n><CR>

FUNCTION: The command character <com> is replaced by the new command character <n> specified in the command argument. The power-up default command character is control-N. On command completion, the following status is returned using the new <com> character.

<com><LF><CR>

EXAMPLE: <com>N<SP>/<CR>

Replaces the current command character with / (slash). This is useful when the CH1770 is being controlled by a human through a 'dumb' terminal, since / is a printing character.

ORIGINATE

SYNTAX: <com>O[RIGINATE]<CR>

FUNCTION: Takes the CH1770 OFF HOOK and forces an originate mode call. This command is useful if a call is manually dialed and later a data call is desired.

If the remote modem's answer tone is not detected after nominally 17 seconds, the CH1770 returns the status:

<com>N<LF><CR>

If the call is answered, the following status is returned:

<com>A<LF><CR>

EXAMPLE: <com>O<CR>

Forces an originate call sequence at the modem's currently trained speed.

PROGRAM

SYNTAX: <com>P[ROGRAM][<SP>H_1H_2]<CR>

FUNCTION: Sets or displays the CH1770's internal option parameters. The argument, H_1H_2, is a two digit hex number that specifies the option configuration.

```
  H1        H2
 _______  _______                 PARAMETER DESCRIPTION
|2^3| | |2^0||2^3| | |2^0|
 | | | |  | | | |
 | | | |  | | | 0..disconnects on loss of receive carrier
 | | | |  | | | 1..does not disconnect on loss of receive carrier
 | | | |  | | |
 | | | |  | | 0....does not assert OFF HOOK during modem test
 | | | |  | | 1....does assert OFF HOOK during modem test
 | | | |  | |
 | | | |  | 0......enables sending & receiving of long space on disconnect
 | | | |  | 1......disables sending & receiving of long space on disconnect
 | | | |  |
 | | | |  0........must always be zero
 | | | |
 | | | 0..........enables answer mode speed switching
 | | | 1..........disables answer mode speed switching
 | | |
 | | 0............disables unsolicited status messages
 | | 1............enables unsolicited status messages
 | |
 | 0..............disables echoing of commands
 | 1..............enables echoing of commands
 |
 0................must always be zero
```

If the command is entered without an argument, the current program configuration is returned as status:

<com>H_1H_2<LF><CR>

27-33

If an argument is specified, the internal option changes are made and the command complete status is returned to the host:

<com><LF><CR>

The power-up default status is 00.

H_1 position 2, enables unsolicited status messages if programmed to be a 1. Messages are considered to be unsolicited if they result from an occurrence other than a host command. Such messages occur as a result of auto-answer, incoming telephone line ringing, and disconnect resulting from loss of carrier or received long space.

EXAMPLE: <com>P<CR>

Displays the current configuration

<com>P<SP>04<CR>

Programs the CH1770 to not disconnect on long received spaces and not send long spaces on disconnect. All other parameters are at their power-up default state.

QUERY

SYNTAX: <com>Q[UERY]<CR>

FUNCTION: Commands the CH1770 to return its current status.

The following message is returned:

<com>OCHSAUX./X_1 X_2<LF><CR>

Where:

O	OFF HOOK asserted
C	carrier detected
H,M,L	high speed (H:1200,M:300,L:110 bps)
S	self test enabled
A	analog loop test enabled
U	unlisten mode enabled
X	modem echoes commands
/X_1 X_2	current hex TEST error count

If a period (.) replaces any of the status characters it implies the negative or opposite status condition.

EXAMPLE: <com>Q<CR>

The CH1770's status is queried during a normal data transmission and the following is returned:

```
<com>OCH.....X./00<CR>
```

This indicates that the modem is OFF HOOK, detecting high speed carrier and set to echo all transmitted commands back to the host.

RESET

SYNTAX: <com>R[ESET]<CR>

FUNCTION: Resets the CH1770 to its power-up default condition (see individual commands for default condition).

EXAMPLE: <com>R<CR>

The modem resets to its power up default state then waits for a speed and parity training sequence. A minimum 300 ms pause after this command response must proceed the new training sequence.

TEST

SYNTAX: <com>T[EST][<SP><n>]<CR>

FUNCTION: Commands the entry and exit of the test mode. To enter the test mode, the TEST command is issued with the argument <n> which indicates any of 4 different test modes:

<n>	Test Mode
0	Analog Loop, originate mode
1	Analog Loop, answer mode
2	Analog Loop Self Test, originate mode
3	Analog Loop Self Test, answer mode
(absent)	Exit the test mode.

To exit any of the TEST modes the TEST command should be given again without an argument. The resulting status sent back to the host is exactly as described in the QUERY command.

TESTS	DESCRIPTION
0,1: Analog Loop	modulates the transmit data asserted on TXD and loops it back through its own demodulator and outputs the resultant receive data on RXD. This allows the host to test the entire CH1770 by checking to see if the receive data matches the transmit data.
2,3: Analog Loop Self Test	same as Analog Loop except the CH1770 transmits a pseudo random data pattern in place of the host data. It then checks to see that the correct data pattern is received. Detected errors cause the CH1770 to increment its self test error register. Since the self test register is cleared upon the start of the test mode, the error count status returned at the end of the test command represents the number of data errors encountered during the test. The register maximum count is FF hex.

EXAMPLE: <com>T<SP><n><CR>

Enter test mode specified by <n>

<com>T<CR>

Exit test mode

UNLISTEN

SYNTAX: <com>U[NLISTEN][<SP><n>]<CR>

Where <n> is 0 or 1 or absent

FUNCTION: Enables or disables the CH1770 command interpreter in the data transmission mode. The command interpreter is always enabled when the CH1770 is not in the middle of a data call. Once a call is set up, three different UNLISTEN modes are possible. At power-up, the default state of the command interpreter is in the LISTEN mode.

<n>	Description
argument absent	listens for commands in the data mode
0	does not listen for commands in the data mode until the host transmits a break (start bit, data bits all zero, and a zero stop bit). Thereafter, the CH1770 listens for host commands until it is commanded to again UNLISTEN. The host's break signal is absorbed by the command interpreter and does not pass on through to the telephone line.
1	does not listen for commands in the data mode. The only way the host can command the CH1770 to disconnect at the end of the call in this mode is to assert $\overline{\text{DTR}}$ OFF (high). The command assures that inadvertent embedded commands in blocks of data sent through the CH1770 will not inappropriately affect the data transmission. Both binary and ASCII files can therefore be passed through the modem with complete data transparency assured. The CH1770 returns the following status to show command completion: <com><LF><CR>

EXAMPLE: <com>U<CR>

Sets the CH1770 to listen to commands in the data mode.

<com>U<SP>0<CR>

Sets the CH1770 to not listen (or unlisten) to commands during the data mode until it receives a break.

<com>U<SP>1<CR>

Sets the CH1770 to not listen to commands during the data mode.

Z Z Z Z

SYNTAX: <com>Z[ZZZ]<CR>

FUNCTION: Makes the modem quiet. When the command is given, the modem squelches its transmitter and stays OFF HOOK. To escape this command, either the ORIGINATE, ANSWER or END commands should be issued.

If both modems on either end are optioned to not disconnect on carrier loss, this command can enable the telephone connection to be used for voice and data switching under host control.

The following status is returned on completion of the ZZZZ command:

<com><LF><CR>

8.0 FCC GENERAL INFORMATION

FCC rules and regulations under part 68, requires the following information be provided to the user of FCC-registered terminal equipment such as the Cermetek CH1770.

Section 68.100 GENERAL

Terminal equipment may be directly connected to the telephone network in accordance with the rules and regulations...of this part.

Section 68.104 STANDARD PLUGS AND JACKS

(a) General

> "Except for telephone company-provided ringers, all connections to the telephone network shall be made through standard (USOC) plugs and standard telephone company-provided jacks, in such a manner as to allow for easy and immediate disconnection of the terminal equipment. Standard jacks shall be so arranged that if the plug connected thereto is withdrawn, no interference to the operation of equipment at the customer's premises which remains connected to the telephone network shall occur by reason of such withdrawal."

Section 68.106 NOTIFICATION TO TELEPHONE COMPANY

"Customers connecting terminal equipment or protective circuitry to the telephone network shall, before such connection is made, give notice to the telephone company of the particular line(s) to which such connection is to be made, and shall provide to the telephone company the FCC Registration Number and Ringer Equivalence of the registered terminal equipment or protective circuitry. The customer shall give notice to the telephone company upon final disconnection of such equipment or circuitry from the particular lines(s)."

Section 68.108 INCIDENCE OF HARM

"Should terminal equipment or protective circuitry cause harm to the telephone network, the telephone company shall, where practicable, notify the customer that temporary discontinuance of service may be required; however, where prior notice is not practicable, the telephone company may temporarily disconnect service forthwith, if such action is reasonable in the circumstances. In case of such temporary discontinuance, the telephone company shall (1) promptly notify the customer of such temporary discontinuance, (2) afford the customer the opportunity to correct the situation which gave rise to the temporary discontinuance, and (3) inform the customer of the right to bring a complaint to the Commission pursuant to the procedures set forth in Subpart E of this Part."

Section 68.216 REPAIR OF REGISTERED TERMINAL EQUIPMENT AND REGISTERED PROTECTIVE CIRCUITRY

"Repair of registered terminal equipment and registered protective circuitry shall be accomplished only by the manufacturer or assembler thereof or by their authorized agent; however, routine repairs may be performed by a user, in accordance with the instruction manual if the applicant certifies that such routine repairs will not result in noncompliance with the rules in Subpart D of this Part." (This applies anytime during and after the factory warranty period. Faulty equipment should be returned to the Cermetek Distributor or Cermetek Factory for repair.)

Section 68.218(b) ADDITIONAL INSTRUCTIONS TO USER

1. "...registered terminal equipment or protective circuitry may not be used with coin lines."

2. "...when trouble is experienced, the customer shall disconnect the registered equipment from the telephone line to determine if the registered equipment is malfunctioning, and...if the registered equipment is malfunctioning, the use of such equipment shall be discontinued until the problem has been corrected."

3. "...the user must give notice to the telephone company in accordance with the requirements of Section 68.106..." for connecting the CH1770 to the telephone line.

27-40

9.0 ELECTRICAL SPECIFICATIONS

Power: +5V +/-5%, -5V +/-5%

TA: 0-60 degrees C

PARAMETERS	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
LOGIC I/O LINES						
Input high	Vih		2.0			V
Input Low	Vil		-.3		.8	v
Input current high	Iih				500	µA
Input current low	Iil				-500	µA
Output high	Voh	OH = .2mA	2.4	3.5		V
Output low	Vol	OL = .2mA	0.0	.2	.45	V
TELEPHONE LINE INTERFACE						
AC Impedance	Zline			600		ohm
Surge Protection		Conforms to all FCC Part 68 surge, hazardous voltage, and leakage				
Carrier Transmit Level	Ptx	600 ohm line termination	-11	-10	-9	dbm
Carrier Receive Sensitivity	Rcar	OFF to ON detection		-43		dB
		ON to OFF drop out		-48		dB
ON HOOK Impedance	Zonhk		20M			ohm
LOOP CURRENT	Iloop			20	100	mA
FCC Registration Number		B468NR-68618-DM-E				
RINGER EQUIVALENCE				0.4B		

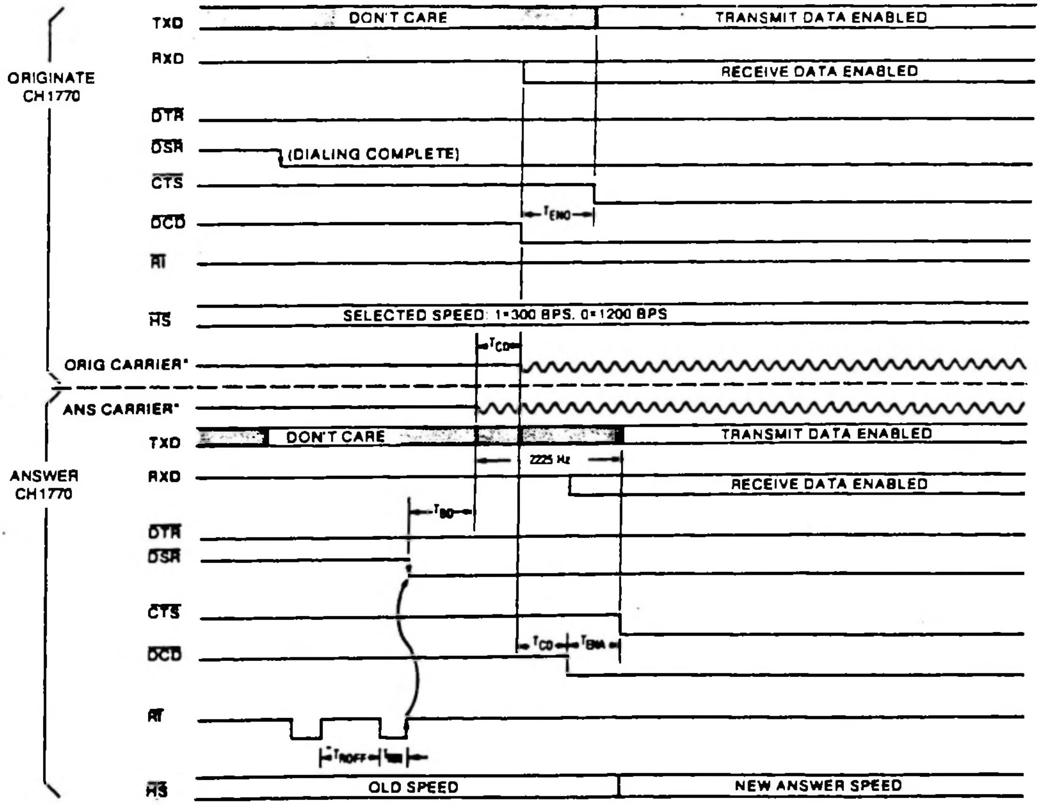
27-41

PARAMETERS	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
DIALING						
DTMF LEVEL			-8	-6	-4	dBm
DTMF FREQ ACCURACY			-1.0		+1.0	%
DTMF ON TIME			90	100	110	ms
DTMF OFF TIME			90	100		ms
PULSE SPEED			9	10	11	PPS
PULSE RATIO		MAKE/BREAK RATIO		40/60		%
PULSE INTERDIGIT			650	700	750	ms
DTE INTERFACE TIMING						
Carrier Detect	Tcd		150	--	300	ms
Clear to Send Delay (Answer)	Tena	110 or 300 BPS	175	200	225	ms
		1200 BPS	750	775	800	ms
Clear to Send Delay (Originate)	Teno	110 or 300 BPS	125	150	175	ms
		1200 BPS	1400	1475	1500	ms
Billing Delay	Tbd		2.0		2.2	Sec
Ring Cycle ON	Tron		0.1	2.0	3.0	Sec
Ring Cycle OFF	Troff		0.5	4.C	6.0	Sec
DISCONNECT TIMING						
DTR Forced	Tdtr	DTR asserted OFF (high)	50			ms
		DTR asserted ON (low)		10		
Long Received Space	Tlrs	Optional	1.6			Sec
Loss of Carrier	Tlc	Carrier drop out, Optional	77			ms
Send Long Space	Tsls	Optional	4		4.5	Sec
POWER						
	Icc	Current at +5 volts		300	350	mA
	Ihmv	Current at -5 volts		30	45	mA

27-4

PARAMETERS	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
WEIGHT					.3	lb
SIGNALING RATE		1200 BPS PSK Asynchronous	-2.5		+1.0	%
		110 or 300 BPS, FSK	-2.5		+2.5	%
BIT Error Rate		Average Line, 9dB S/N 1200 BPS			10^{-5}	Errors/Sec
		Average Line 5dB S/N 110 or 300 BPS			10^{-5}	Errors/Sec

CH1770 HANDSHAKE TIMING DIAGRAM



32 Kbyte RAM Board

This section describes the optional Matrix 7911/RPC memory board. The manual put out by Matrix follows.

The RAM board is typically loaded with 16 2 Kbyte static RAM chips. When used with a revision A processor board, the RAM board is set to provide 32 K of RAM from address 0 up (the bottom 2 K overlays the RAM on the processor board since the RAM board allocates in 4 K blocks). When used with a revision B processor board, the RAM board provides 32 K of RAM starting up 8 K (address $2000). Finally, the RAM board can be partially loaded with EEPROM (X2816A), allowing larger programs to be held in EEPROM. If loaded as shown below, a Basic program can be saved to the RAM board EEPROM using the statement SAVE EEPROM 24576. The program can be loaded using the statement LOAD EEPROM 24576. Note that if the board is partially loaded with EEPROM, it is necessary to leave the "6B" socket on the RAM board empty. This provides a "hole" in the memory map that Basic finds on initialization. If this hole were not present, Basic would try to use the EEPROM as RAM, overwriting the program.

7911/RPC Jumpers for use with Rev A Processor

J2 - J4	strap none
J3 - J5	strap 1-8

7911/RPC Jumpers for use with Rev B Processor (all RAM)

J2 - J4	strap 3-8
J3 - J5	strap 1-2

7911/RPC Jumpers for use with Rev B Processor with EEPROM

J2 - J4	strap 3-8
J3 - J5	strap 1-2

Load with EEPROM:	1A, 1B, 2A, 2B, 7A, 7B, 8A, 8B
Load with RAM:	3A, 3B, 4A, 4B, 5A, 5B, 6A
Leave Empty:	6B

Chips suitable for use in the 7911/RPC include:

Type	H&F P/N	Manufacturer	Part Number
RAM	3130-2016	Toshiba	TMM2016P-2
		Mostek	MK4802-N-3
		Mitsubishi	M58725P
EEPROM	3130-2816	Xicor	X2816AD-45
		Seeq	DQ2816A-350

2 8-2

7911/RPC

User's Manual

MATRIX CORPORATION RALEIGH, NC (919)833-2837

28-3

7911/RPC RAM PROM CARD

REVISION A

This manual has been carefully checked for accuracy and is thought to be fully consistent with the product it describes. Although Matrix Corporation assumes full responsibility for this manual's content, we do not assume liability. Furthermore, the information contained herein is of a proprietary nature and is not to be reproduced without prior written consent of Matrix Corporation.

TABLE OF CONTENTS

TABLE OF CONTENTS (Con't)

SECTION 6

Appendix

SECTION 1

GENERAL INFORMATION

1.1 Introduction

This Users manual contains detailed information on the preparation and use of the 7911/RPC, RAM PROM Card. Also included in this manual is documentation concerning theory of operation, component layout, and the electrical schematic. All address and data references preceded by a $(dollar) sign are in hexadecimal notation.

1.2 General Description

The 7911/RPC is a 32K Byte Random Access or Read Only Memory board, configured as 8-4K byte, relocatable blocks of memory space.

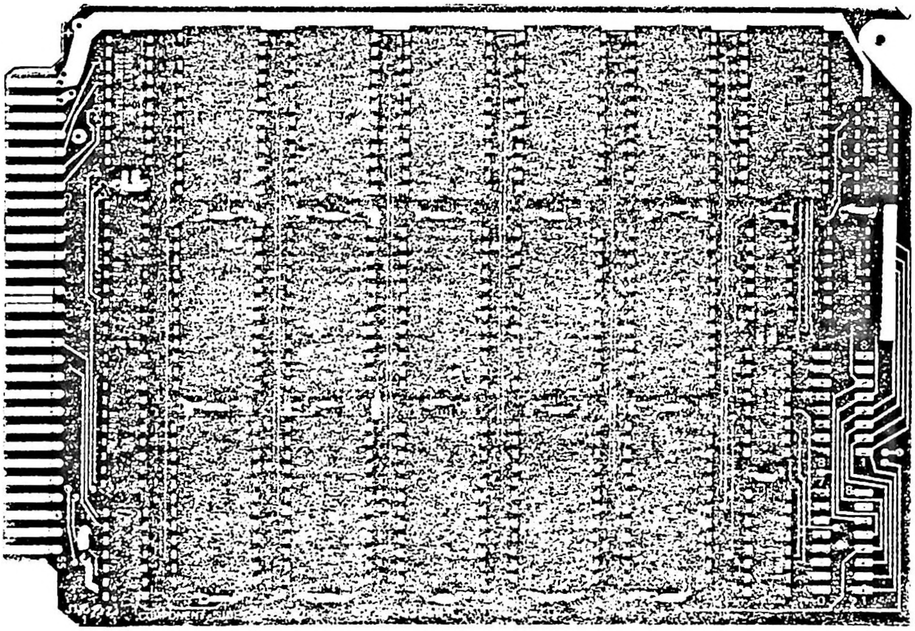
This card conforms to all the STD BUS standards and provides a complete 64K memory system with only 2 cards.

1.3 General Design

The 7911/RPC is designed to provide a high density memory unit, as versatile as possible, meeting all specifications of an STD BUS system. The address lines are decoded in a manner that permits 16 4K positions in the 64K address space, each 4K block consists of two memory ICs. Each IC uses the industry standard 2716 pin outs.

1.4 Block Diagram

The principal function of the 7911/RPC is to provide the User with high density memory space available for static RAM or EPROM on a single card. Figure 1.1 shows, functionally, how it was done. Configuration of the lines to the memory ICs was selected so the largest number of types of ICs could be accommodated. All clocks, address, control and data lines are buffered from the STD BUS.



7911/RPC RAM PROM CARD

FUNCTIONAL BLOCK DIAGRAM

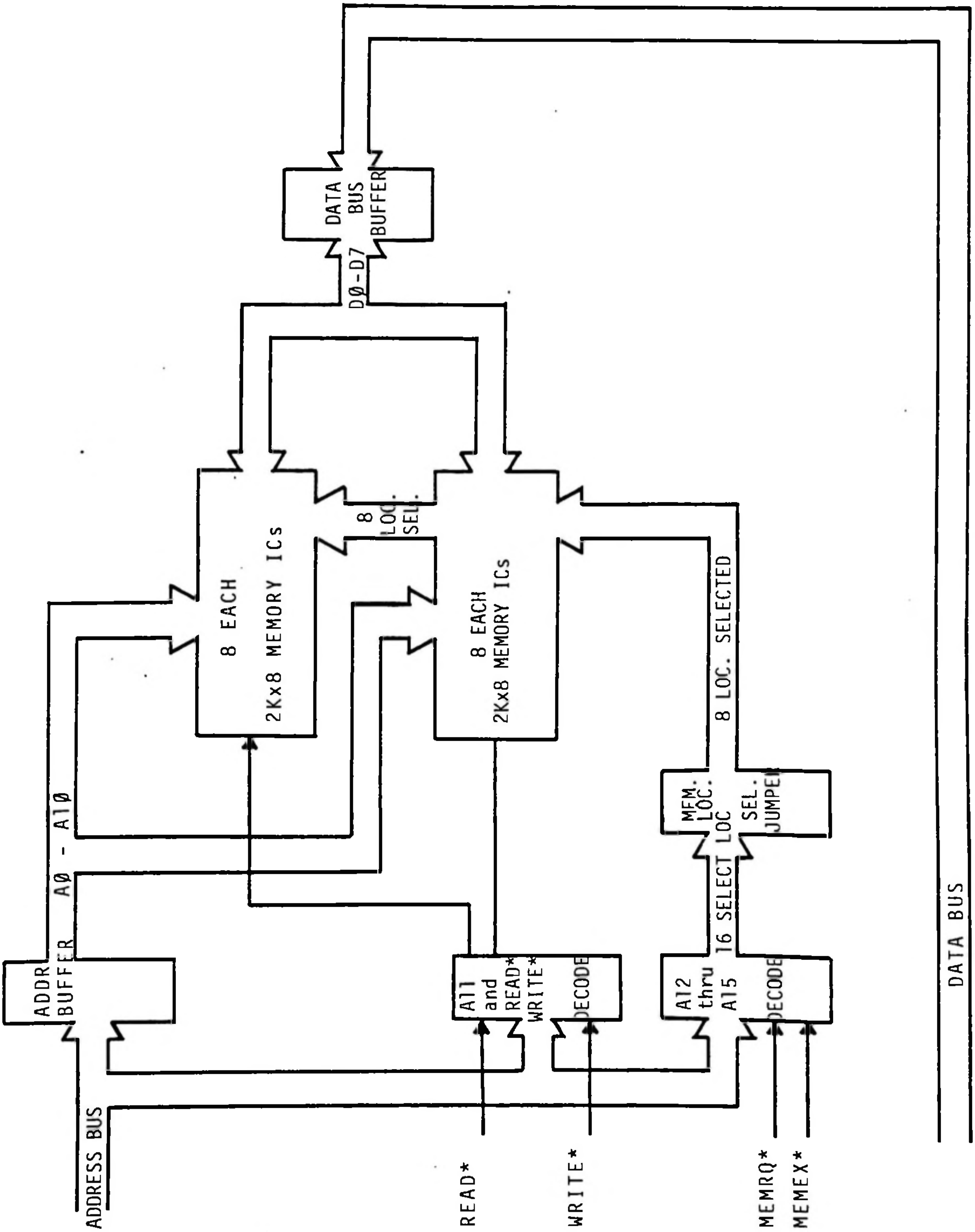


FIGURE 1.1

SECTION 2

PREPARATION AND INSTALLATION

2.1 General

This section provides instructions to prepare the 7911/RPC memory board for use in STD BUS Systems. Included are unpacking instructions, for the proper environmental and electrical requirements of the 7911/RPC, and strapping instructions, for all operational options of the unit. Installation instructions and procedures are included in this section.

2.2 Unpacking and Inspection

Carefully unpack the 7911/RPC and remove the anti-static shipping bag. It is recommended that the anti-static bag be retained for future use in storing the unit should it be removed from service. Closely inspect the unit for any visible damage (i.e., cracked board, damaged or missing ICs, or other components, etc.) and note any discrepancies. Note: If a damaged unit is received, contact Matrix Corporation immediately for handling instructions.

2.3 Operation Requirements

The 7911/RPC operates at temperatures of from 0°C to 55°C in relative humidity not to exceed 95%. The unit requires +5VDC at 450mA, maximum (fully populated RPC-32).

2.4 Card Addressing

Each 4K block of memory may be addressed at any location in a 64K address space by means of wire wrap jumpers between J2-J4 and J3-J5. Each connection on J4-J5 corresponds to its numbered block of memory, and each connection of J2-J3 corresponds to a decoded 4K address space. J2 decodes 8 4K blocks of address space for $0000-7FFF and J3 decodes 8 4K blocks of address space for $8000-FFFF.

2.5 Memory Expansion

Jumper J1 has been provided to allow the 7911/RPC to be used in the "primary" or "expanded" memory maps. With J1 installed from 1 to 2, the "primary" memory map is enabled (MEMEX*Low). Installing J1 from 2 to 3 enables the 7911/RPC for "expanded" memory operation (MEMEX* High).

2.6 Installation Procedures

Installation of STD BUS products, including the 7911/RPC, is uncomplicated. It is suggested that the following precautions and checks be observed when installing the 7911/RPC in the STD BUS shelf:

1. Do not attempt to insert or remove units from the bus while the bus is powered. Always remove power from the bus before any mechanical operations are performed.

2. Check to see that all jumpers are installed for correct addressing of the installed memory, and the expansion jumper is selected for proper enabling.

To install the 7911/RPC in the STD BUS rack, simply grasp the card, align the card with the correct shelf slots, push the card in place and firmly seat the card in the connector at the rear of the shelf. Note: Although the 7911/RPC will function equally well at any position in the STD BUS shelf, it is recommended that it be placed to conform to a STD BUS shelf plan of the users own design.

Should you have any problems or questions about the 7911/RPC or its operation, please contact Matrix Corporation for assistance and/or advice.

SECTION 3

THEORY OF OPERATION

3.1 General

The 7911/RPC memory is designed as main memory for use in a STD BUS system. Each unit consists of up to and including 32K bytes of memory in 2K byte increments. Because of the availability of a large variety of memory devices with the same pin configuration, the 2716 EPROM was used as the interconnect model for this system.

3.2 Operation

All lines from the 7911/RPC to the STD BUS connector are fully buffered. This unit uses 8 data lines, 16 address lines and the WRITE*, READ*, MEMRQ* and MEMEX* lines.

3.2.1 Data Bus

The data lines are connected to the data pins of the memory devices through a Bidirectional Octal Bus Buffer. The direction of data flow is controlled by the buffered READ* line. The Data Buffer is enabled by a signal that is generated by the MEMRQ*, MEMEX*, and the M. S. byte Address Decoders.

3.2.2 Address Bus

The Address bus is buffered by two Octal Bus Buffers.

AØ through A1Ø is connected through these buffers to the AØ through A1Ø pins of the memory devices. A-11 is decoded with READ* and WRITE* for OUTPUT ENABLE*, or the WRITE ENABLE* respectively, of one group of 8 memory devices when A-11 is logic 1, and a second group of 8 memory devices when A-11 is logic Ø. A12 through A14 are decoded by One of Two Decoders, IC20 and IC21, selected by A15, and connected to the chip select pins of the memory devices in pairs, through an address select jumper connector to permit location of any 4K memory block on any 4K boundary, anywhere within a 64K memory map.

3.2.3 Control Bus

MEMEX* is provided with a strapping option to permit the User the option of using this unit as "primary" memory or "expanded" memory, if the User system supports these functions. The selected MEMEX* and MEMRQ* are used to enable the A12 through A14 Address Decoders and,with these selected decoders,output to enable the Data Bus Buffer.

SECTION 4
CIRCUIT DESCRIPTION

4.1 General

This section provides a technical discussion of each of the functional areas of the 7911/RPC memory unit. Refer to Figure 1.2, Block Diagram, and the Schematic Diagram in the Appendix for the following discussion.

4.2 Address Buffers and Decoders

The address lines AØ through A15 are connected to Octal Buffers, IC2 and IC3. Address lines AØ through A1Ø from IC2 and IC3, connect directly to the address lines AØ through A1Ø of the memory devices, ICs 5-19 and 23. Address line A-11 from IC2 is used by gates in IC22 and IC25, along with READ* and WRITE*, to generate an OUTPUT-ENABLE* and WRITE-ENABLE* respectively, according to Table 4.1.

Address lines A12 through A15 from IC3 are connected to Decoders IC20 and IC21. A15, when at logic 0, selects One of Eight Decoder IC20 which in turn decodes addresses A12 through A14 to provide the ENABLE* signals to J3 accordinq to Table 4.2. J2, therefore, corresponds to the lower 32K of memory, addresses $0000 through $7FFF, and J3 corresponds to the upper 32K, addresses $8000 through $FFFF.

These jumpers, together with the status of the MEMEX* line provide the complete range of address selection. J4 and J5 can be connected to J2 and J3 with wire wrap jumpers to enable any 4K block of memory at any 4K address space.

4.3 Control Logic

MEMRQ*, MEMEX*, READ*, and WRITE* are buffered by Hex Inverter IC1. MEMRQ* and MEMEX* are used to generate a data bus enable signal along with a valid decoded address by a gate of IC22. MEMRQ* and MEMEX* through a gate of IC25 enables the Address Decoders IC2Ø and IC21. READ* and WRITE* signals from IC1 are used with A-11 as explained in 4.2, except that READ* is also used to control data direction in the Data Bus Buffer.

4.4 Data Transceivers

Data is gated to and from the bus by Bidirectional Octal Bus Buffer IC4. A low on pin 1 of IC4, caused by the READ* line of the STD BUS being low, directs the data from the memory devices to the bus. At all other times data are directed from the bus to the memory devices. Data Bus Buffer outputs are not enabled until a low signal from a gate of IC22 is provided as explained previously in 4.3.

READ/WRITE ENABLING				
A-11	WRITE*	READ*	WRITE-ENABLE*	OUTPUT-ENABLE*
1	0	1		ICs 5,8,11,14, 17,10,16,23
1	1	0	ICs 5,8,11,14 17,10,16,23	
0	0	1		ICs 6,9,12,15, 18,7,13,19
0	1	0	ICs 5,9,12,15, 7,13,19	

TABLE 4.1

ADDRESS ENABLING					
A15	A14	A13	A12	J2 LOW	J3 LOW
Ø	Ø	Ø	Ø	Ø	
Ø	Ø	Ø	1	1	
Ø	Ø	1	Ø	2	
Ø	Ø	1	1	3	
Ø	1	Ø	Ø	4	
Ø	1	Ø	1	5	
Ø	1	1	Ø	6	
Ø	1	1	1	7	
1	Ø	Ø	Ø		8
1	Ø	Ø	1		9
1	Ø	1	Ø		A
1	Ø	1	1		B
1	1	Ø	Ø		C
1	1	Ø	1		D
1	1	1	Ø		E
1	1	1	1		F

TABLE 4.2

MEMORY BLOCK ENABLING

J4 & J5 Pin No.	Enable	
	Lower 2K IC	Upper 2K IC
1	19	23
2	13	16
3	7	10
4	18	17
5	15	14
6	12	11
7	9	8
8	6	6

TABLE 4.3

28-18

SECTION 5

SPECIFICATIONS

5.1 Power Supplies

5.1.1 Unit Power

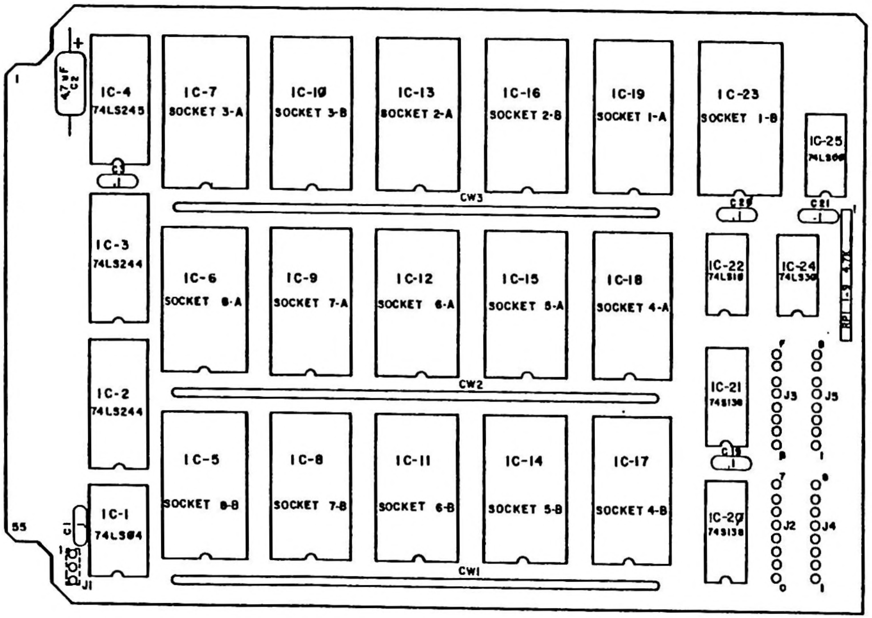
Voltage:	+5VDC, ±5%
Current:	450mA[1], max.
Regulation:	10%
Filtering:	60 db ripple rejection
Voltage Rise Time at Turn-on:	50 ms or faster

[1] Depending on memory devices used.

5.2 Environmental

Storage Temperature:	-25°C to +85°C
Operating Temperature:	0°C to +55°C
Operating Humidity:	Not to exceed 95% relative humidity

REVISED	BY	DESCRIPTION
APRIL 9, 82		REV.A ADD BUS BARS/CHANGE IC S 20 & 21



MATRIX CORP.
RALEIGH, NC

7911 STD SYSTEM

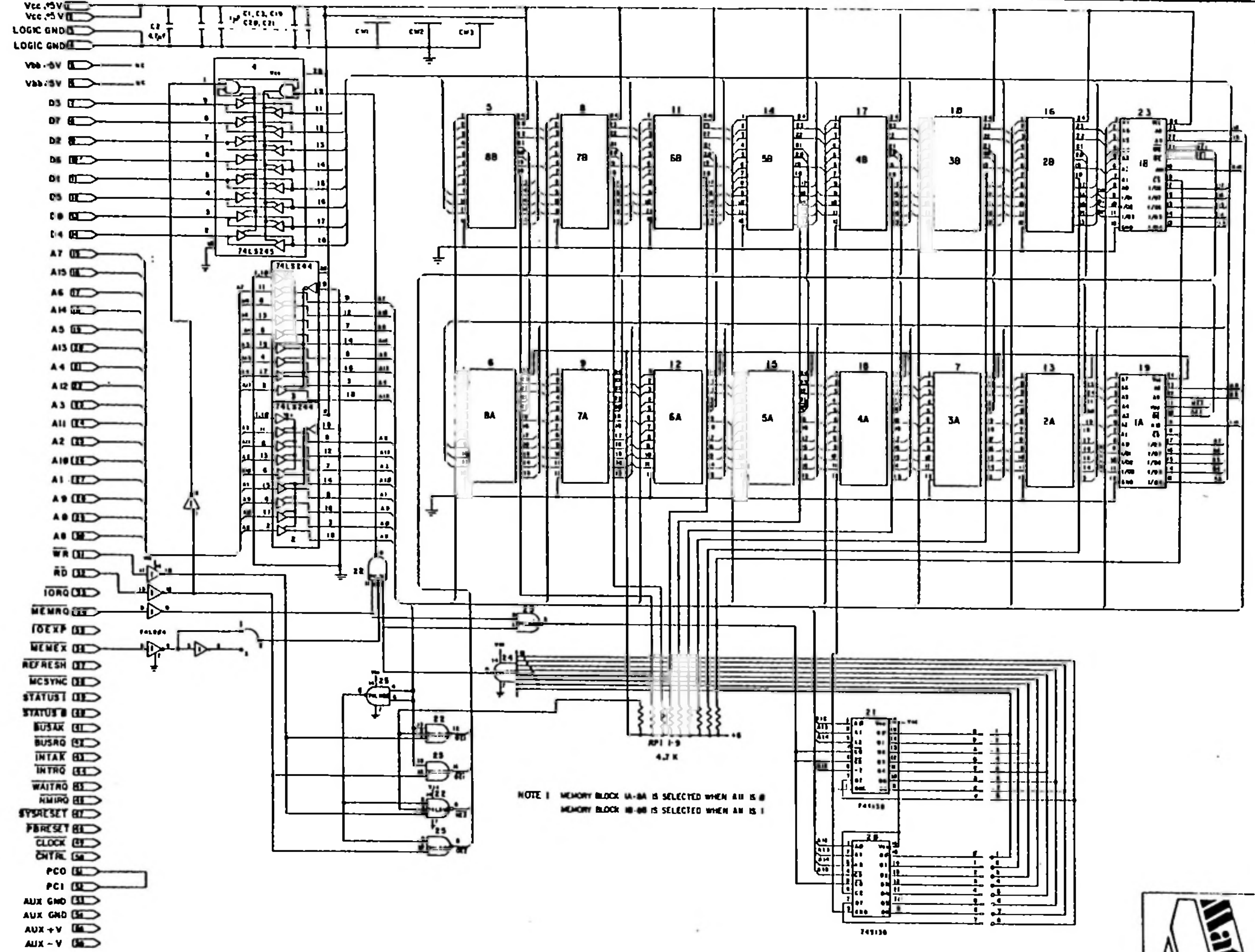
PARTS LAYOUT

32K STATIC RAM/PROM BOARD

(2X) | 6-24-81 | C | 7911-089 | REV. A

28-19

REVISED	BY	DESCRIPTION
APRIL 9, 82	J	REV A ADD BUS BARS · CHANGE .C5 28-2[illegible]



NOTE 1 MEMORY BLOCK 1A-8A IS SELECTED WHEN A11 IS 0
MEMORY BLOCK 1B-8B IS SELECTED WHEN A11 IS 1

MATRIX CORP.

STD SYSTEMS

SCHEMATIC

32 K STATIC RAM/PROM

7-30-81 | D | 7911-086 | REV. A

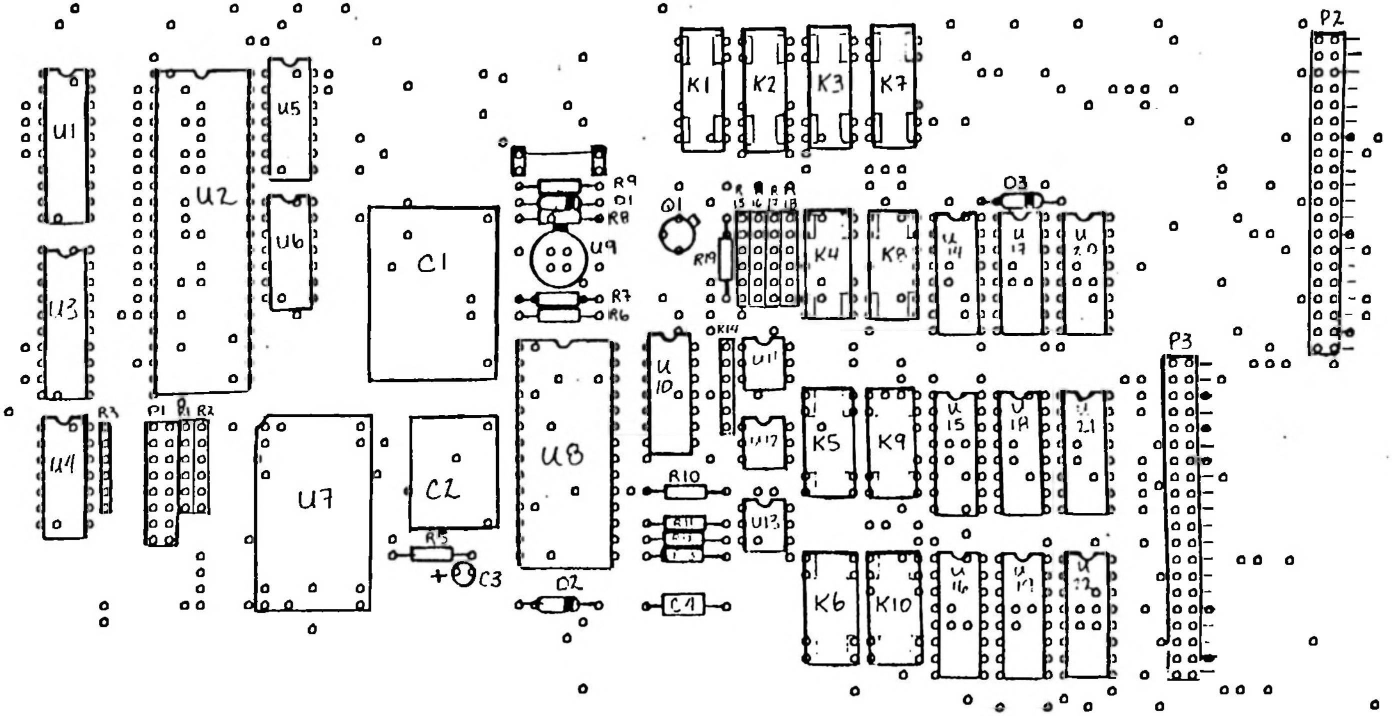
28-20

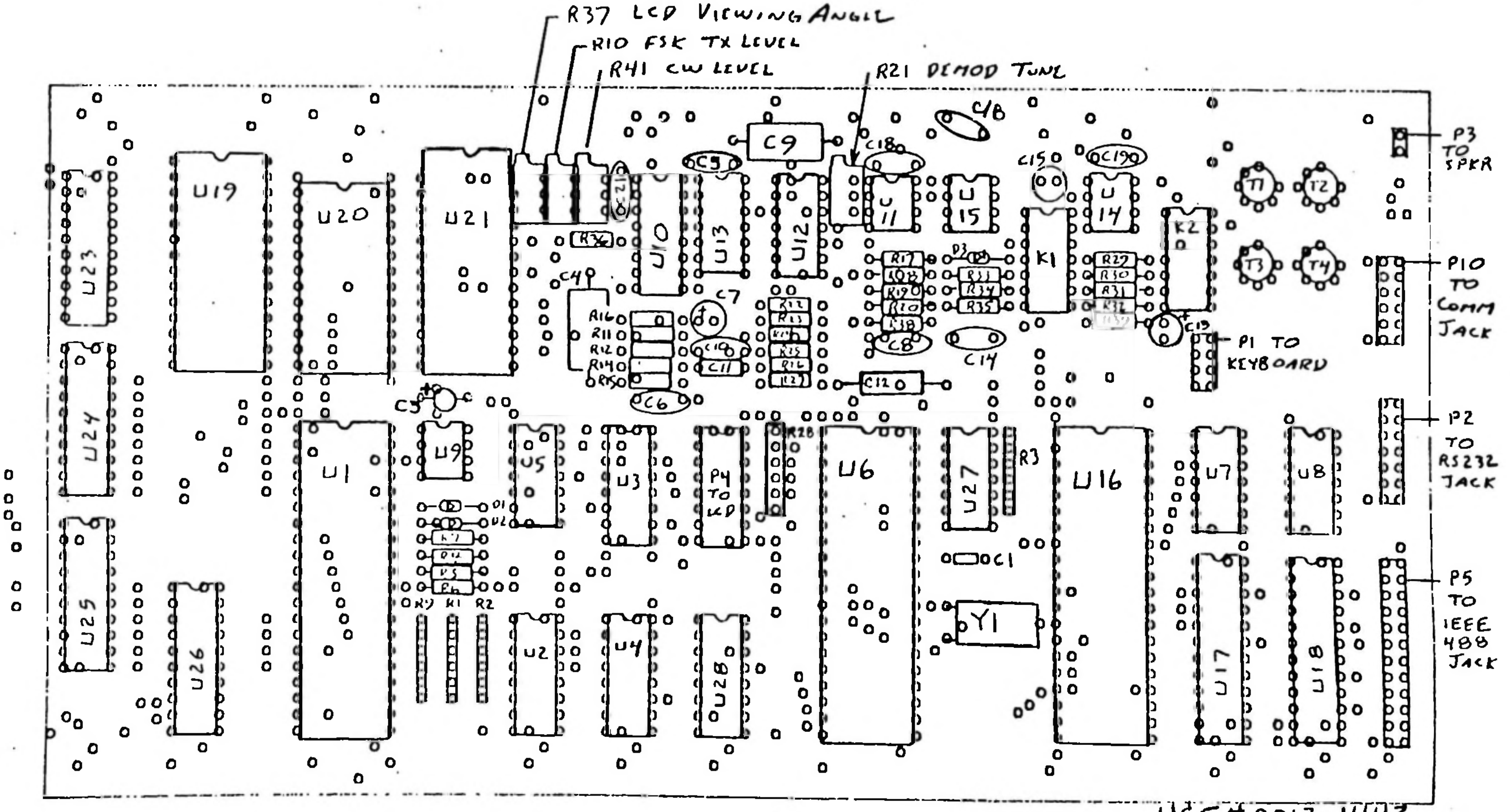
WARRANTY

Matrix Corporation warrants its 7911/Series products to be free from defects in material or workmanship over a period of one year from date of shipment provided it it properly used and not modified by non-Matrix personnel. Matrix shall repair or replace, at its discretion, a defective product provided it is returned to Matrix freight prepaid. No other warranty is expressed or implied.

BE 10/9/85

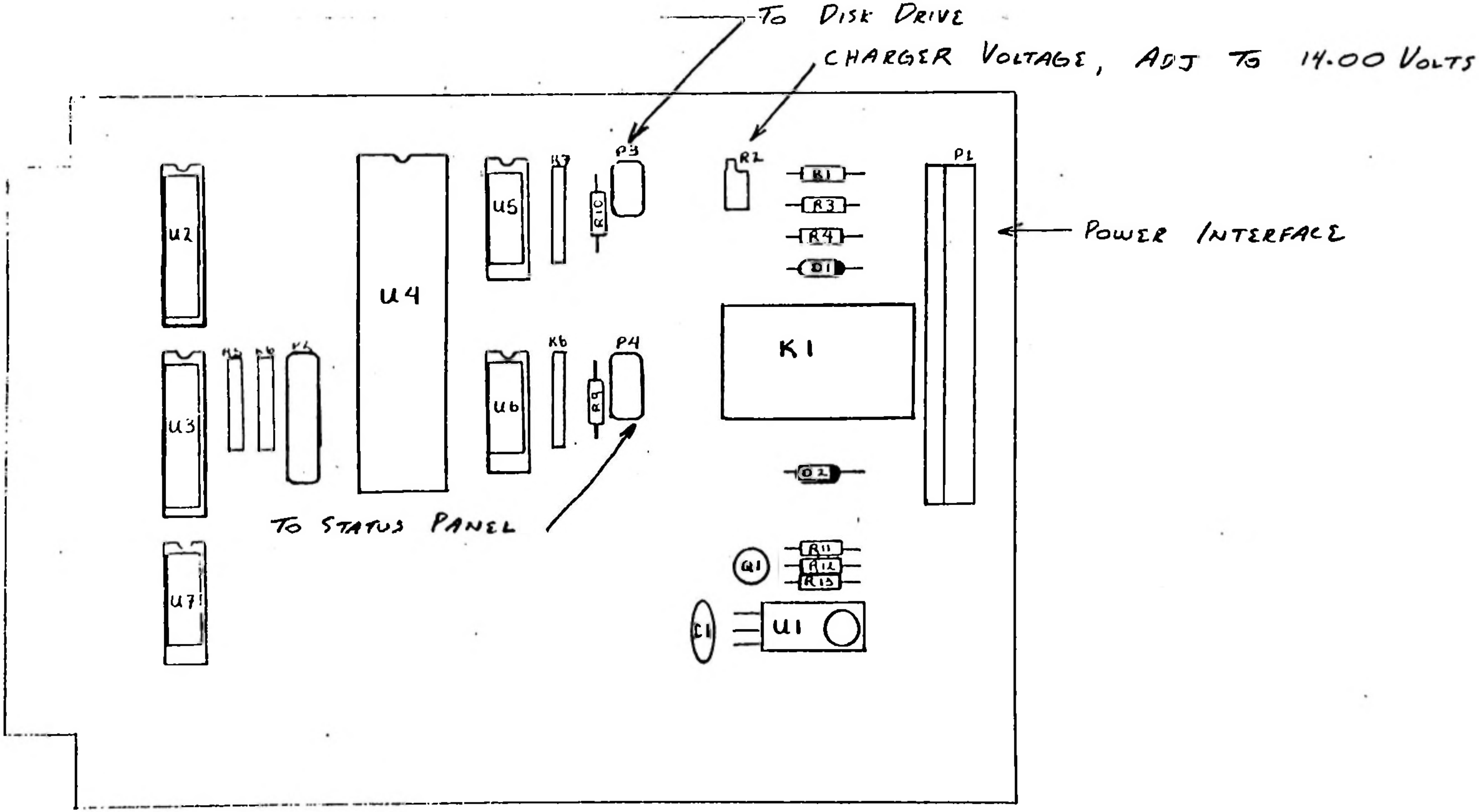
H&F #0010-1441 Component Placement
DRC 190 A/D Sub-assy A-190-1



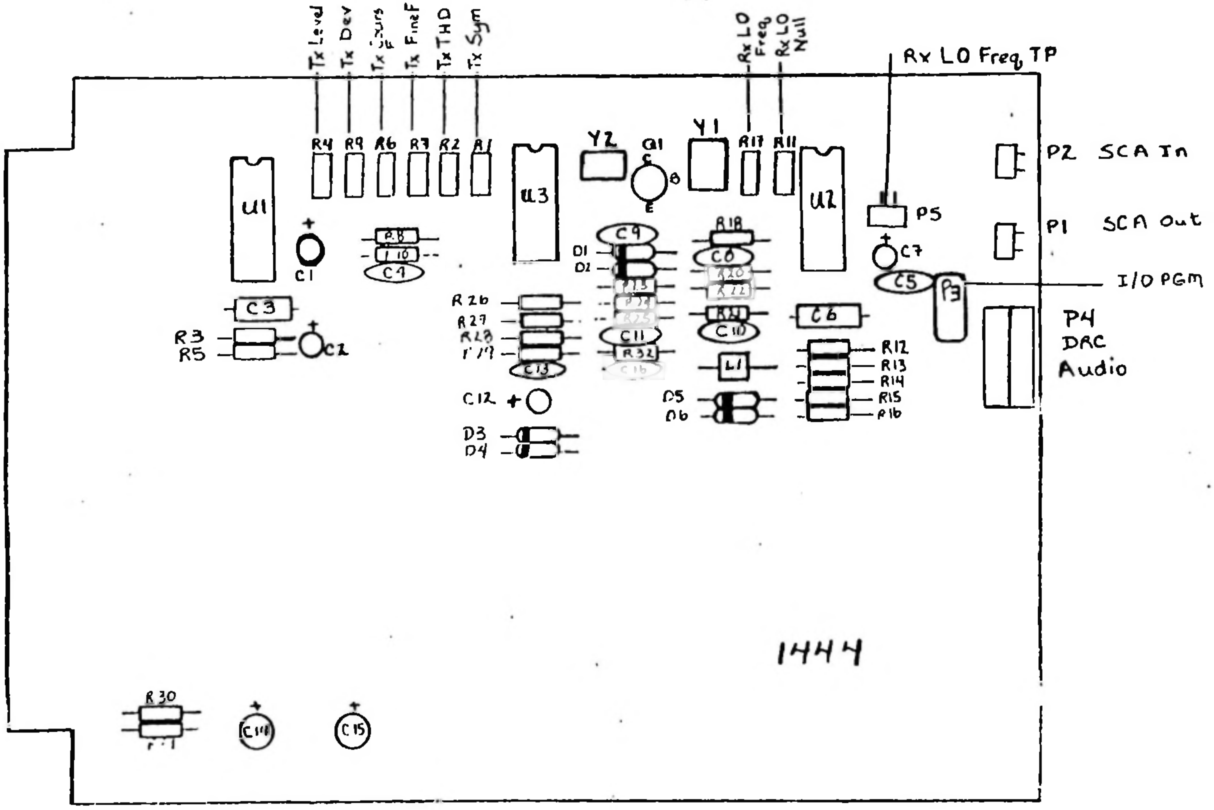


1442 PROCESSOR COMPONENT PLACEMENT HEF #0013-1442

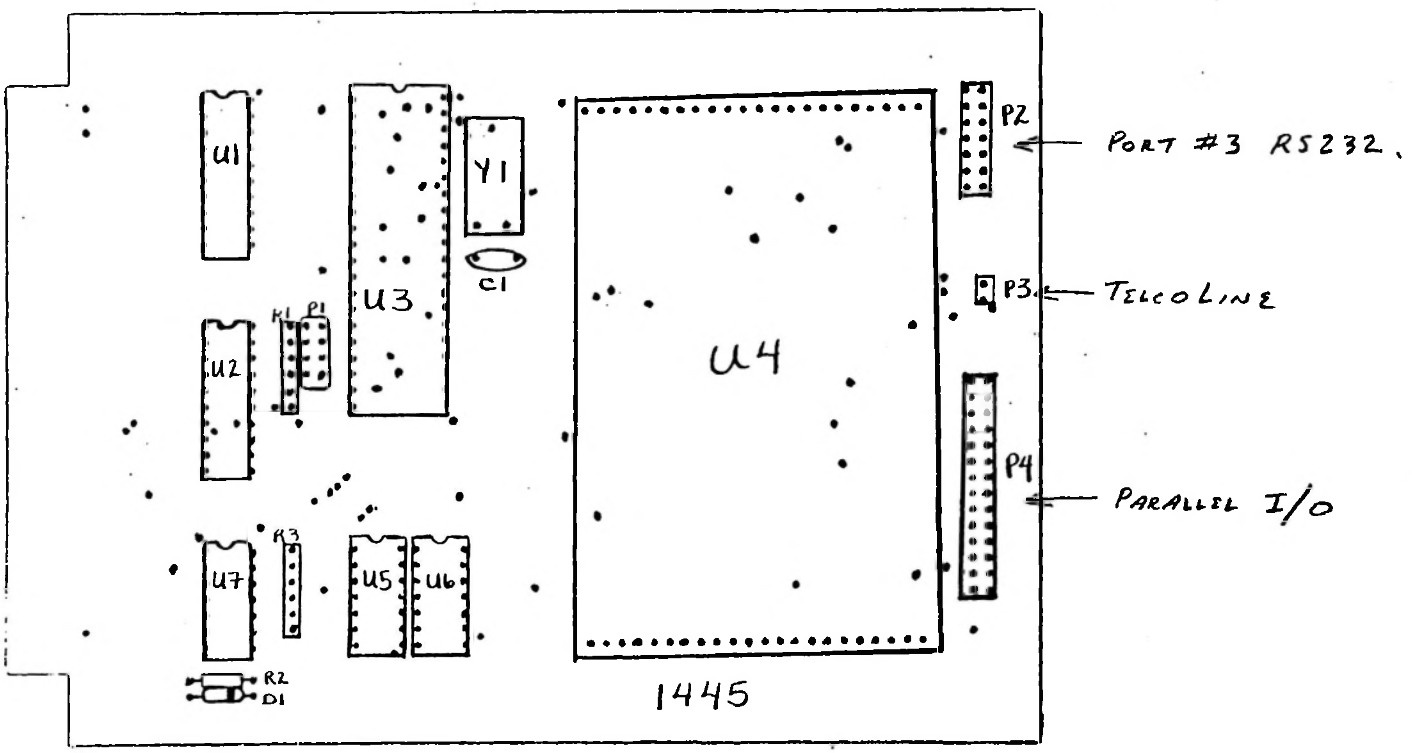
A-203-1



Component Placement – Power, Disk, Status Board 1443



COMPONENT PLACEMENT: SUBCARRIER TRANSCEIVER 1444
H&F AUDIO-1444

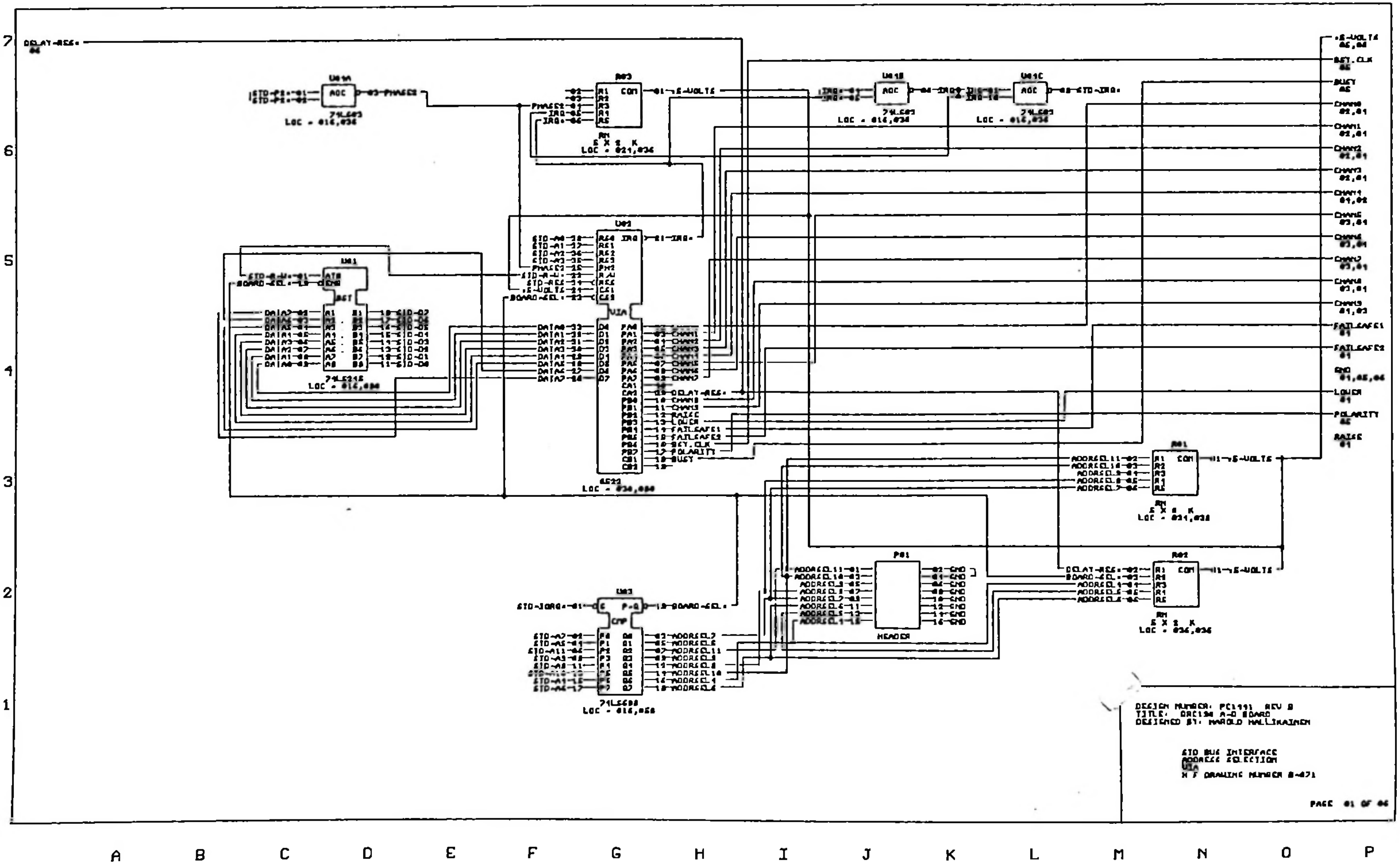


29-5

BE 10/16/85

DRC 190
H & F #0010-1445

Direct Connect
Modem

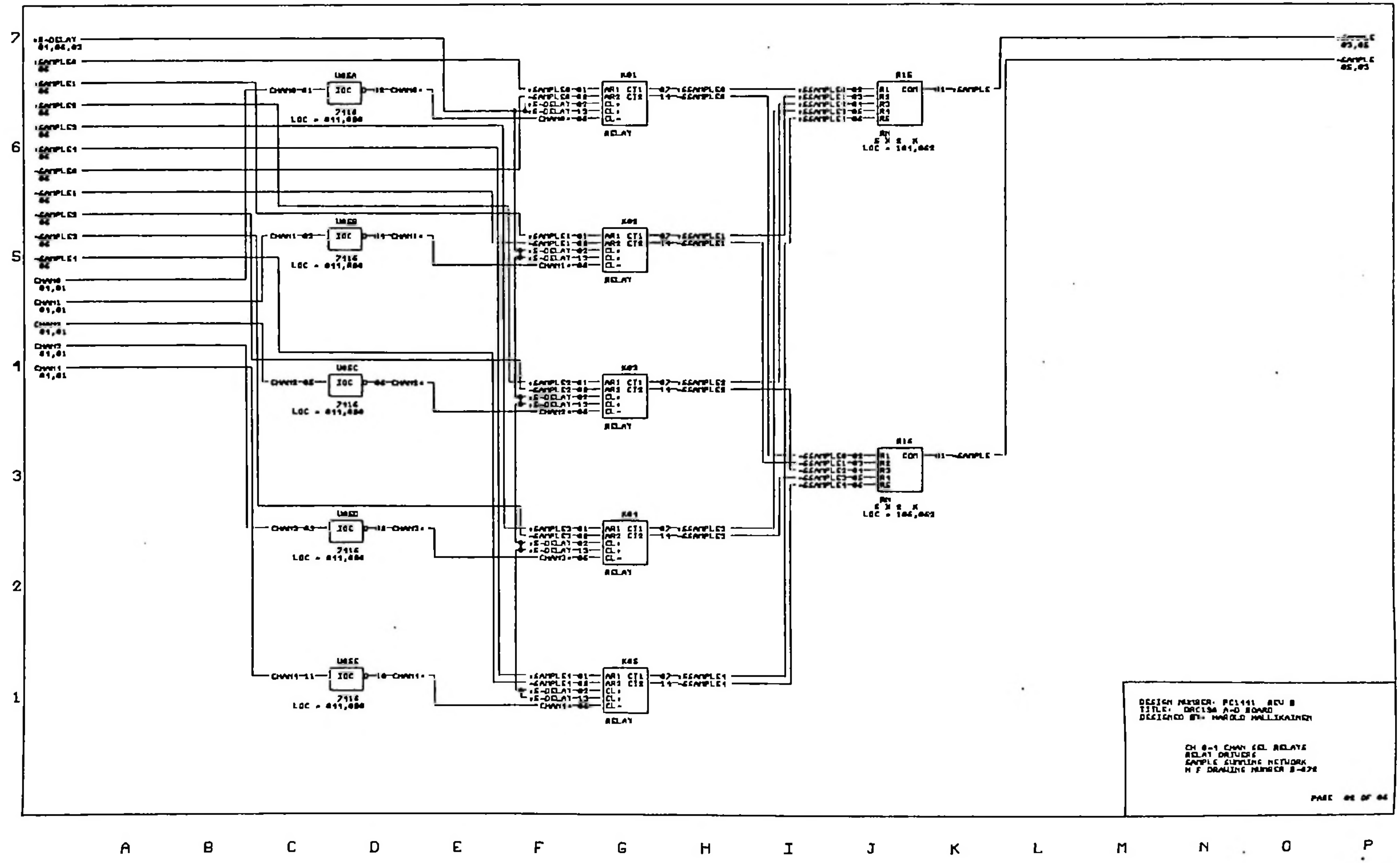


DESIGN NUMBER: PC1491 REV B
TITLE: ORC134 A-D BOARD
DESIGNED BY: HAROLD HALLIKAINEN

STD BUS INTERFACE
ADDRESS SELECTION
VIA
H F DRAWING NUMBER B-471

PAGE 01 OF 06

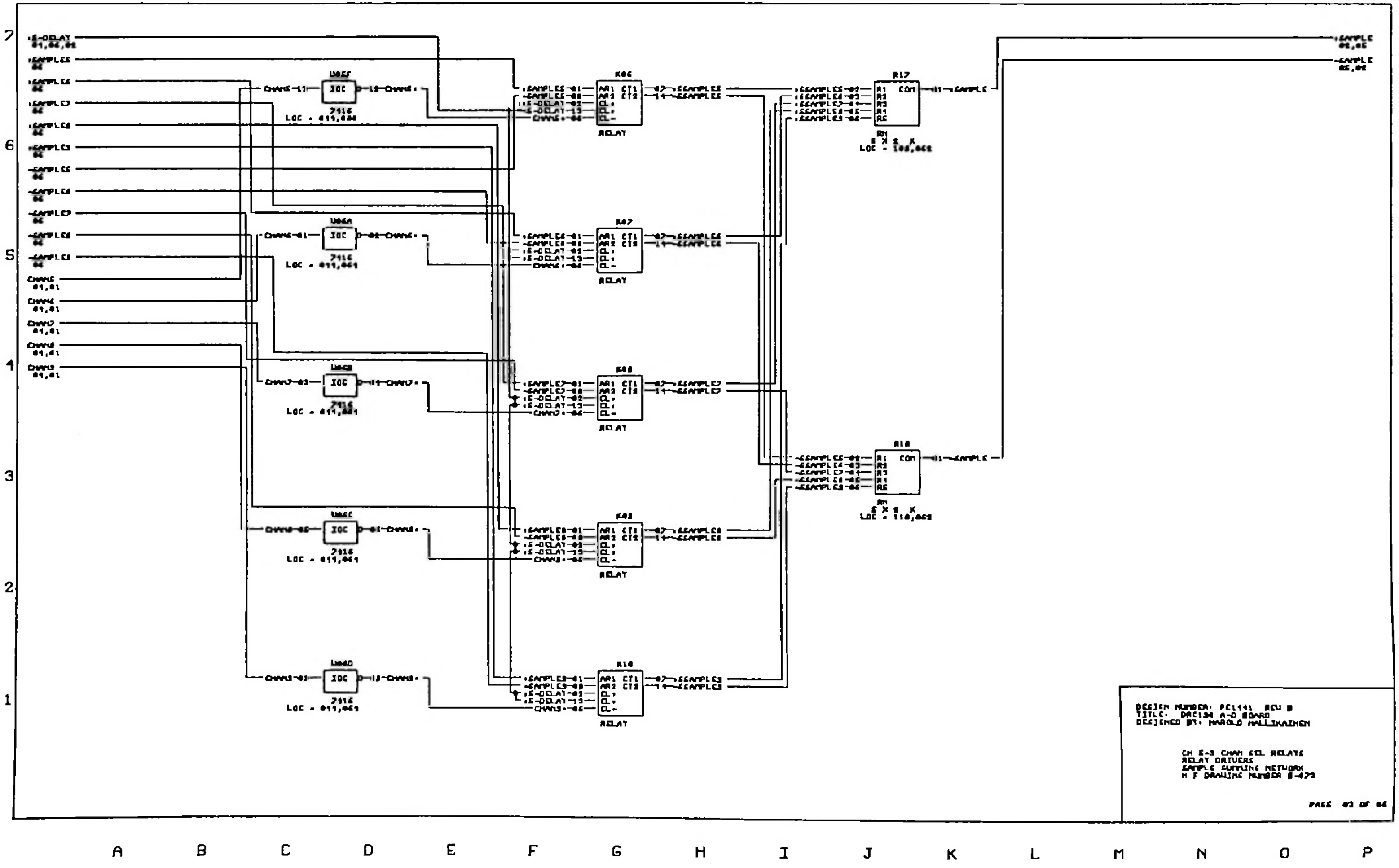
30-1



DESIGN NUMBER: PC1441 REV B
TITLE: DAC136 A-D BOARD
DESIGNED BY: HAROLD HALLIKAINEN

CH 0-4 CHAN SEL RELAYS
RELAY DRIVERS
SAMPLE SUMMING NETWORK
H F DRAWING NUMBER B-478

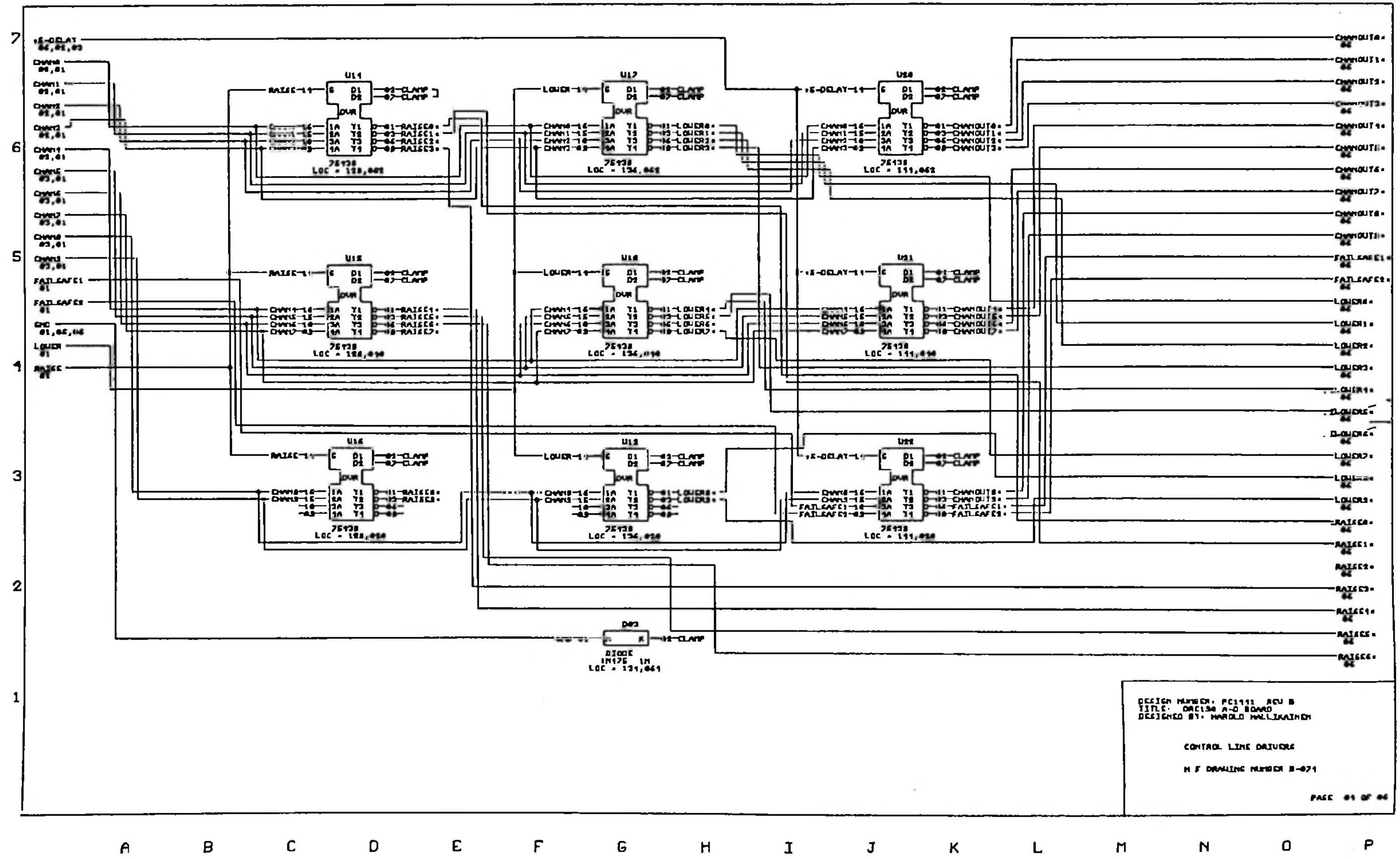
PAGE 04 OF 04



DESIGN NUMBER: PC1141 REV B
TITLE: DAC130 A-D BOARD
DESIGNED BY: HAROLD HALLIKAINEN

CH 5-9 CHAN SEL RELAYS
RELAY DRIVERS
SAMPLE SUMMING NETWORK
H F DRAWING NUMBER B-473

PAGE 03 OF 04



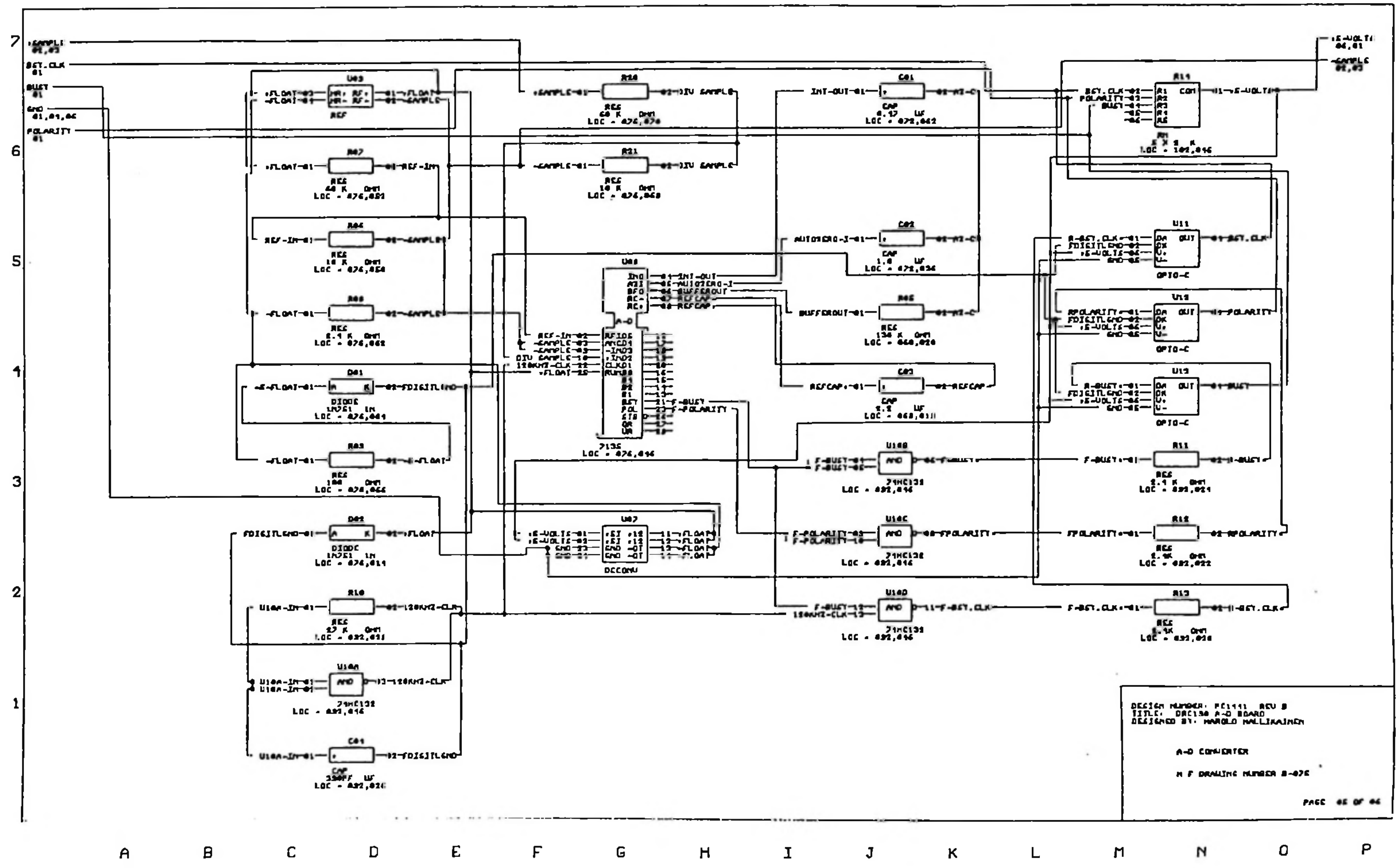
DESIGN NUMBER: PC1111 REV B
TITLE: ORC136 A-D BOARD
DESIGNED BY: HAROLD HALLIKAINEN

CONTROL LINE DRIVERS

H F DRAWING NUMBER B-071

PAGE 01 OF 06

30-H

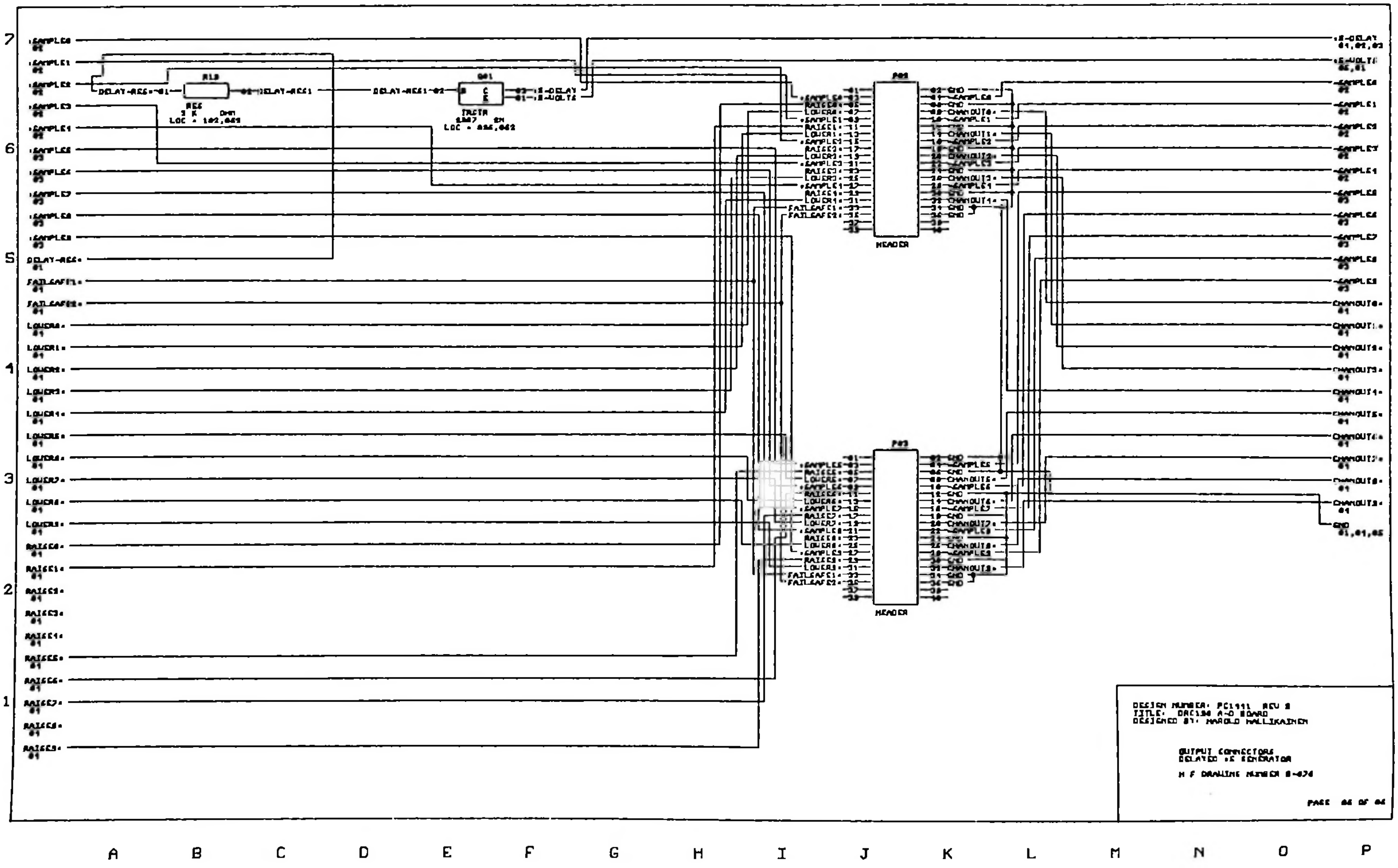


DESIGN NUMBER: FC1111 REV B
TITLE: DAC150 A-D BOARD
DESIGNED BY: HAROLD HALLIKAINEN

A-D CONVERTER

H F DRAWING NUMBER B-076

PAGE 05 OF 06



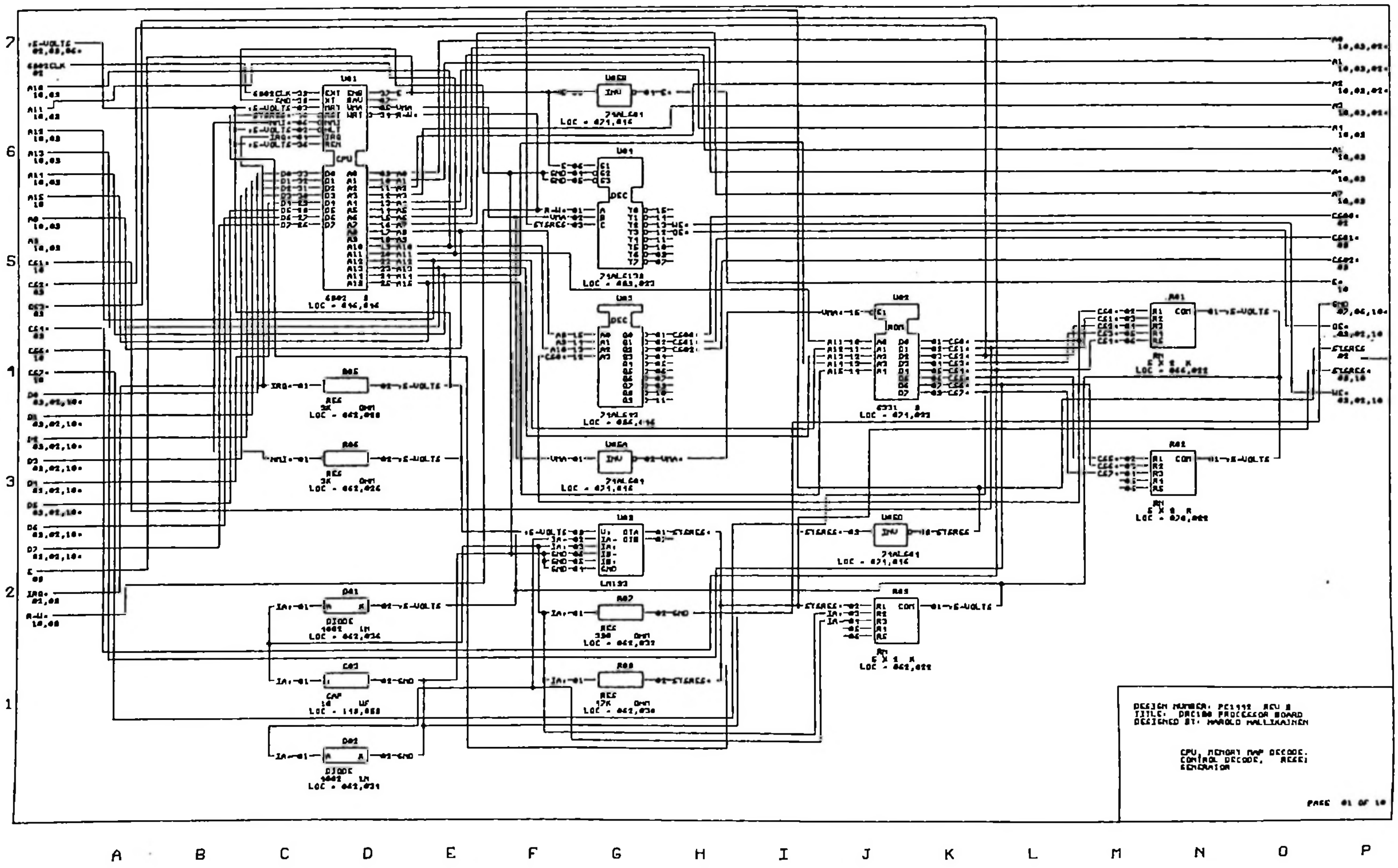
DESIGN NUMBER: PC1911 REV 2
TITLE: DRC190 A-D BOARD
DESIGNED BY: HAROLD HALLIKAINEN

OUTPUT CONNECTORS
DELAYED +5 GENERATOR

H F DRAWING NUMBER 8-474

PAGE 04 OF 04

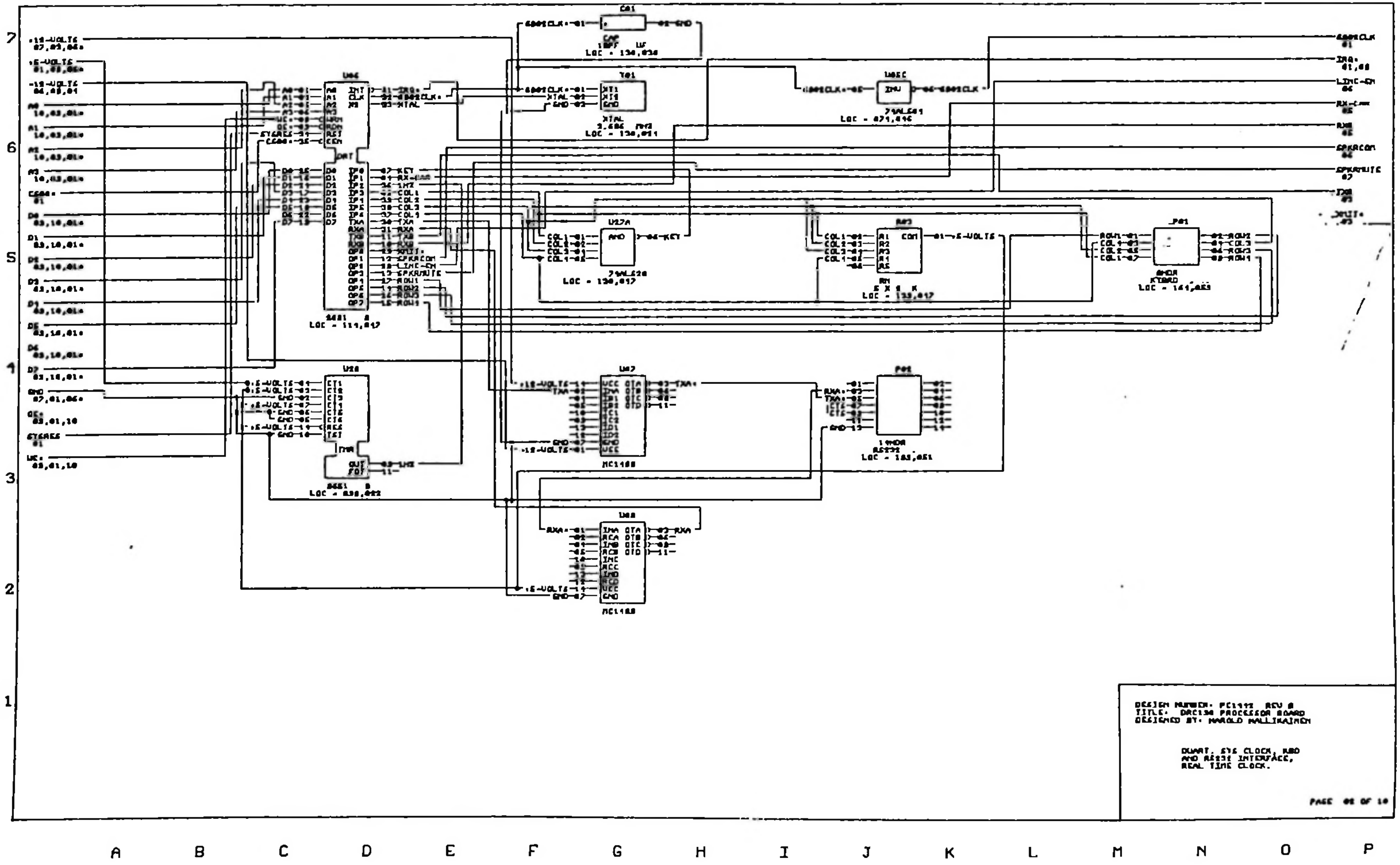
30-6



DESIGN NUMBER: PC1992 REV 8
TITLE: DRC100 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

CPU, MEMORY MAP DECODE,
CONTROL DECODE, RESET
GENERATOR

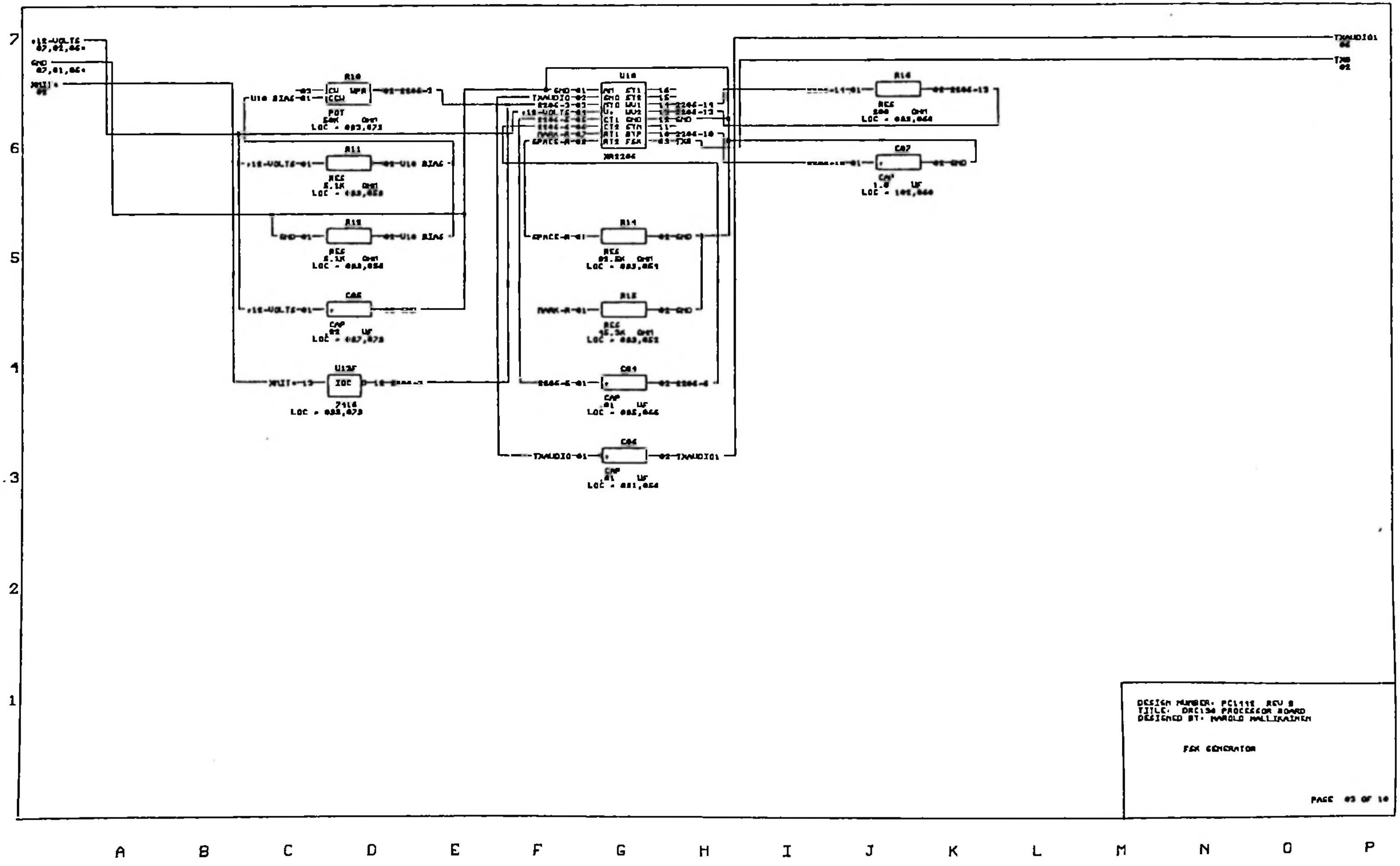
PAGE 01 OF 10



DESIGN NUMBER: PC1112 REV B
TITLE: DRC130 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

DUART, SYS CLOCK, KBD
AND RS232 INTERFACE,
REAL TIME CLOCK.

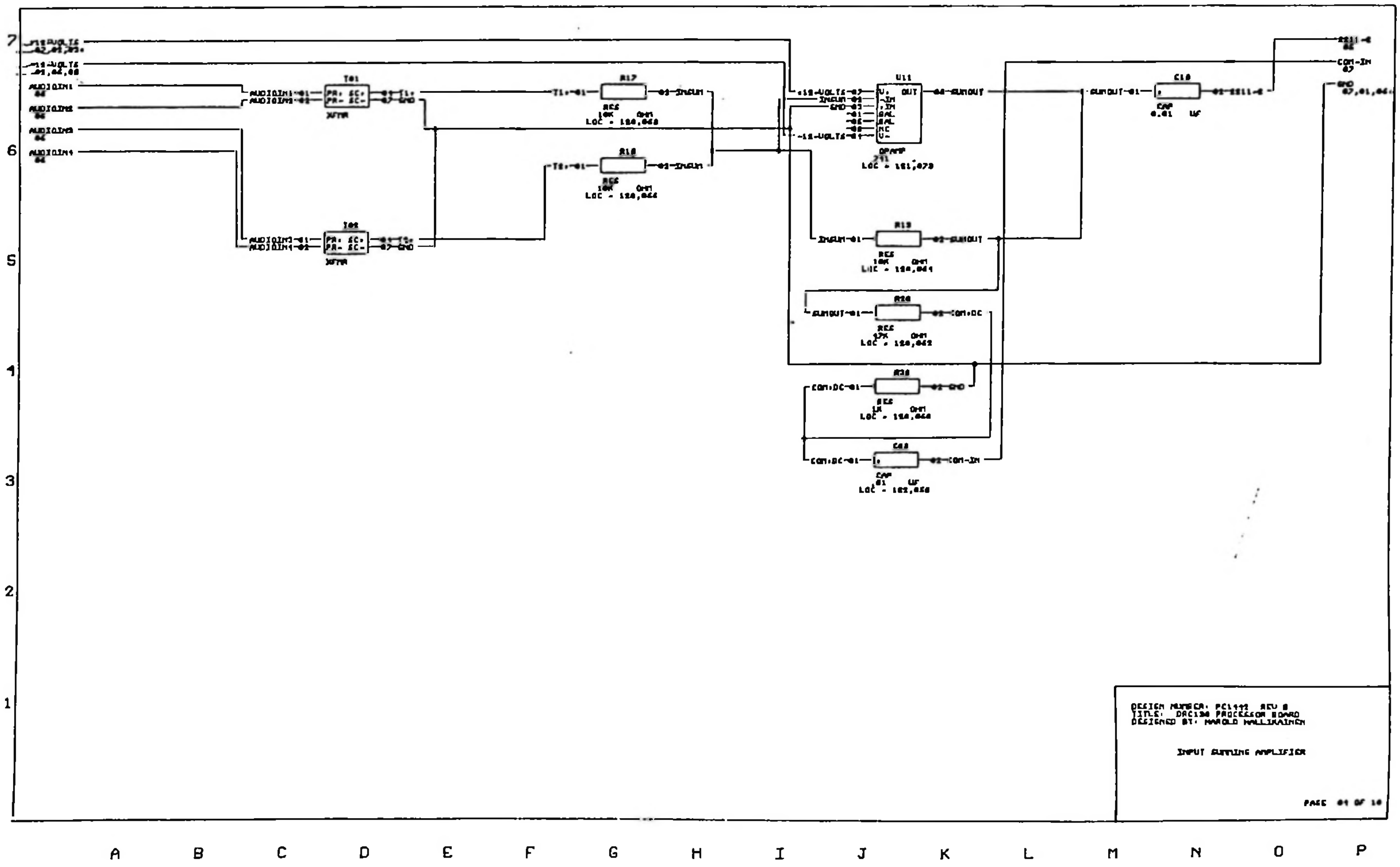
PAGE 05 OF 10



DESIGN NUMBER: PC1112 REV 0
TITLE: DRC150 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

FSK GENERATOR

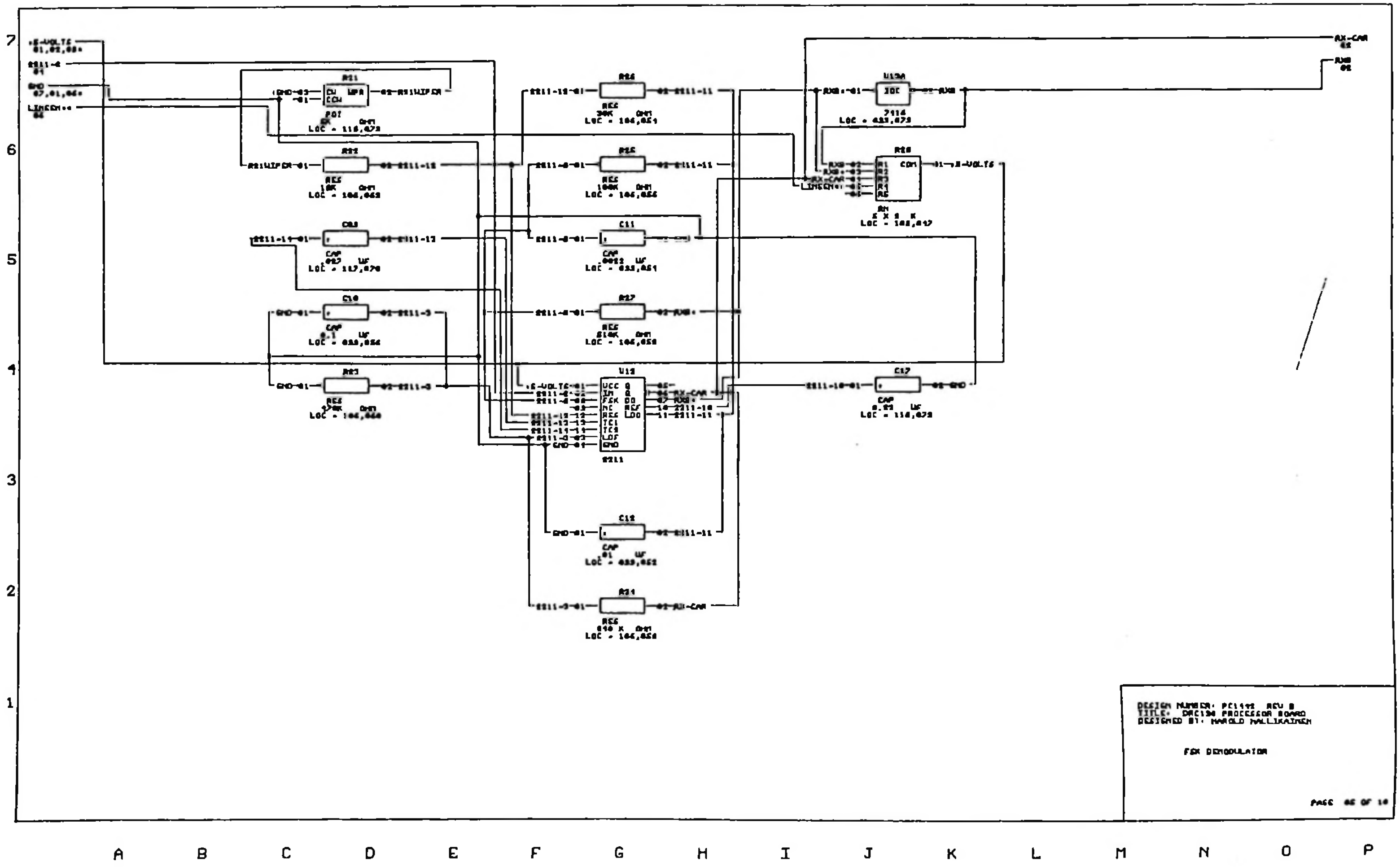
PAGE 03 OF 10



DESIGN NUMBER: PC1442 REV 0
TITLE: DAC136 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

INPUT SUMMING AMPLIFIER

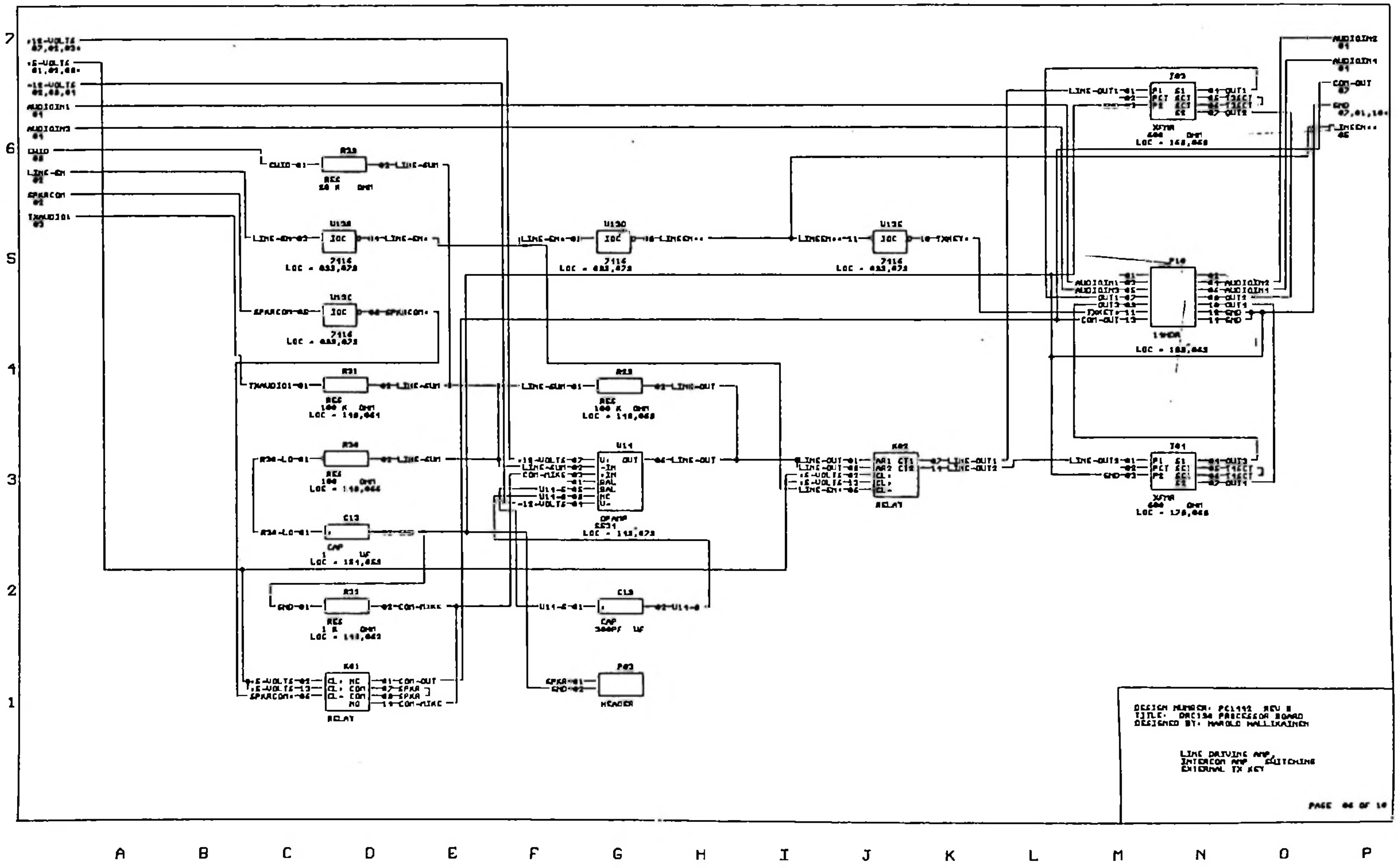
PAGE 04 OF 10



DESIGN NUMBER: PC1492 REV B
TITLE: DRC190 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

FSK DEMODULATOR

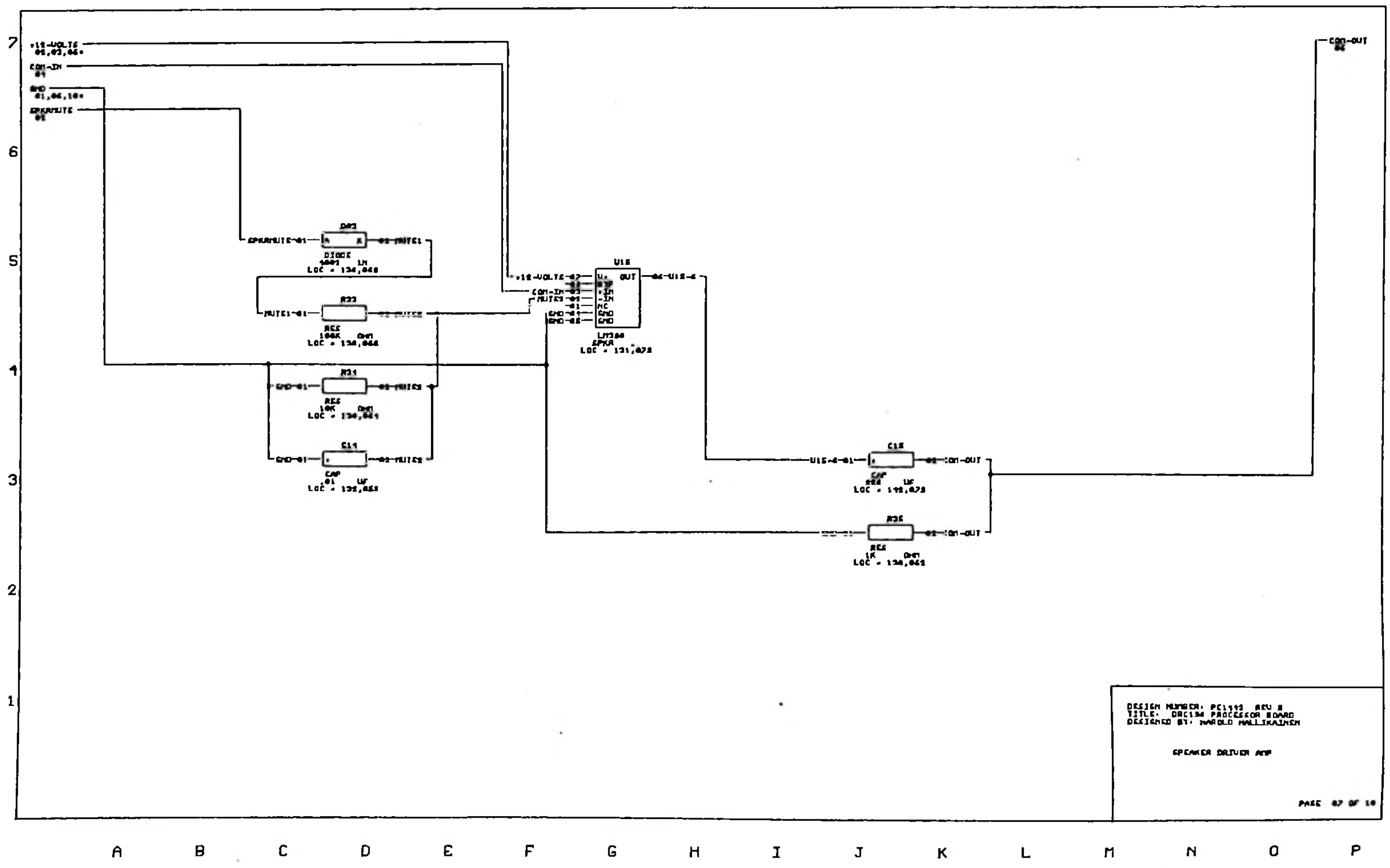
PAGE 05 OF 10



DESIGN NUMBER: PC1492 REV B
TITLE: DRC190 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

LINE DRIVING AMP,
INTERCOM AMP SWITCHING
EXTERNAL TX KEY

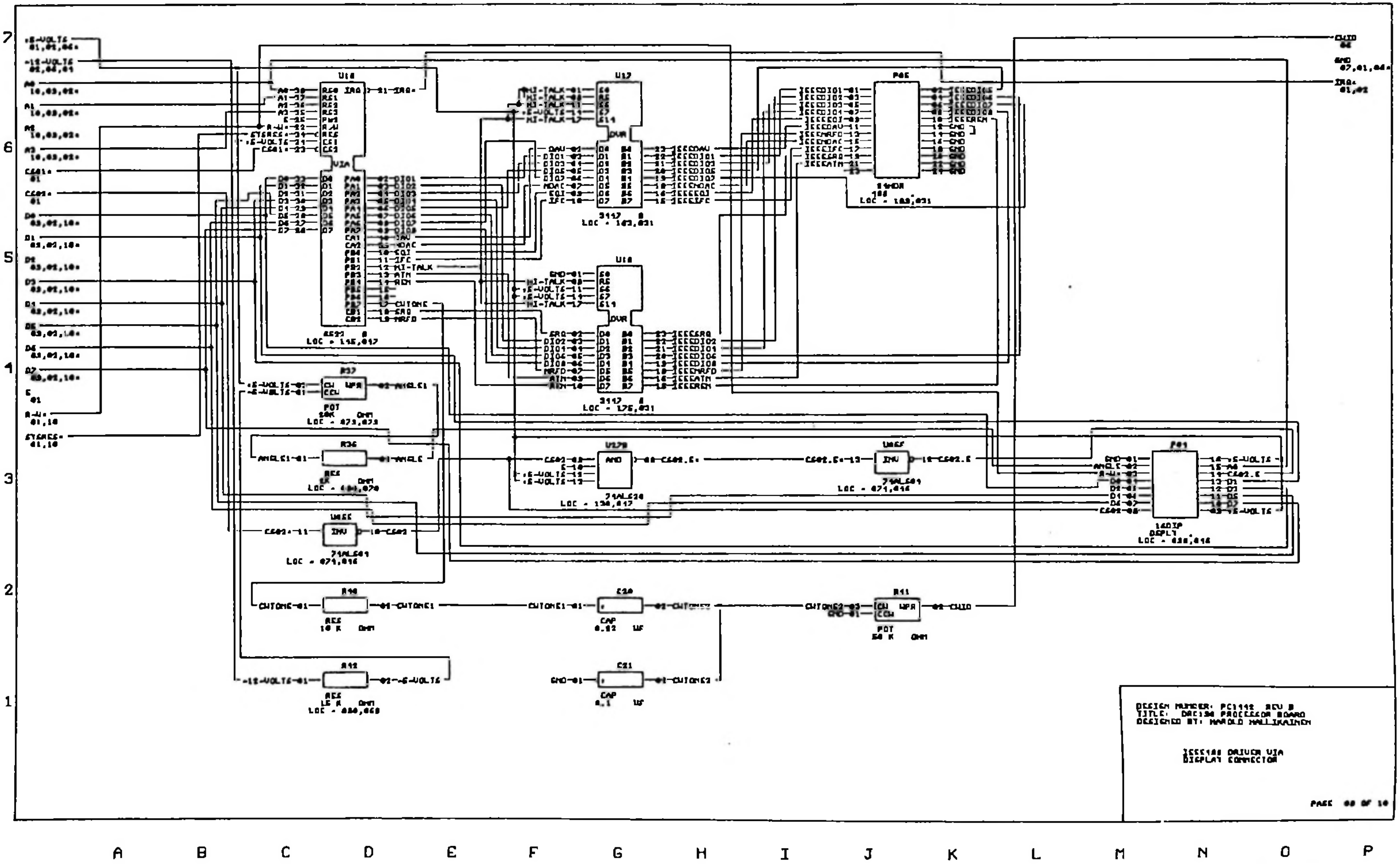
PAGE 06 OF 10



DESIGN NUMBER: PC1392 REV B
TITLE: DRC194 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

SPEAKER DRIVER AMP

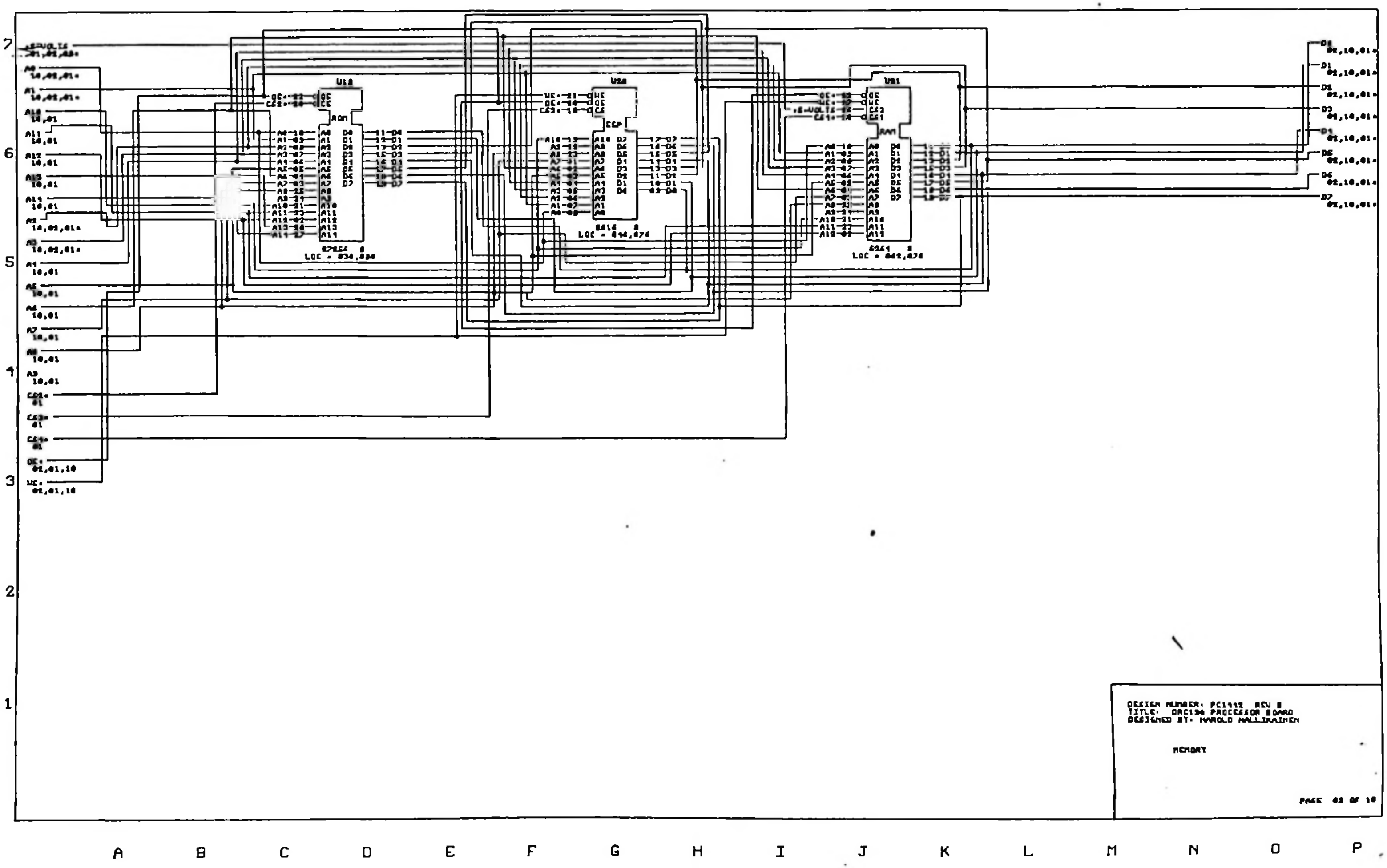
PAGE 07 OF 10



DESIGN NUMBER: PC1192 REV B
TITLE: DAC150 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

IEEE488 DRIVER VIA
DISPLAY CONNECTOR

PAGE 08 OF 10

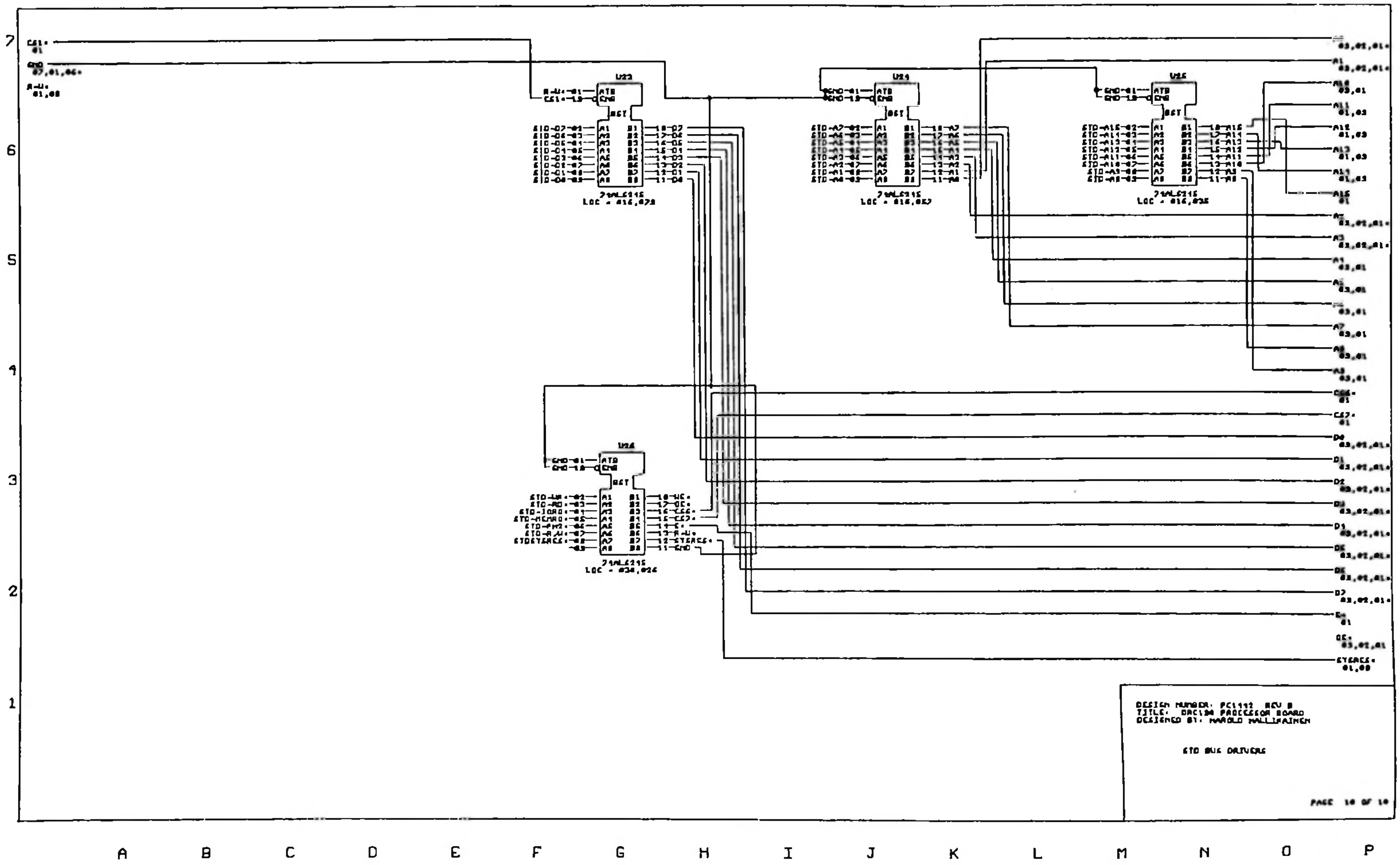


DESIGN NUMBER: PC1412 REV B
TITLE: ORC134 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

MEMORY

PAGE 03 OF 10

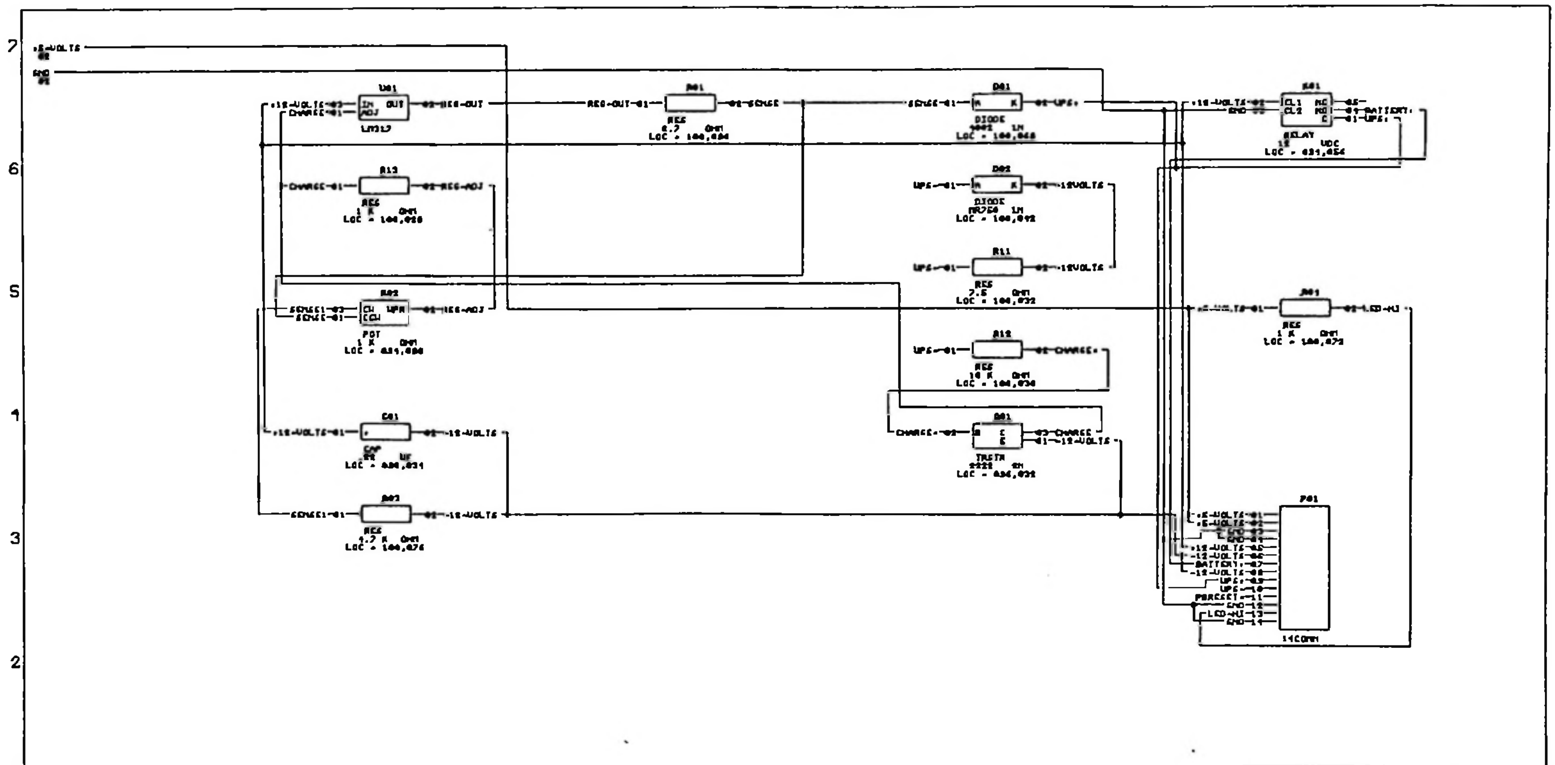
31-9



DESIGN NUMBER: PC1112 REV B
TITLE: ORC180 PROCESSOR BOARD
DESIGNED BY: HAROLD HALLIKAINEN

STD BUS DRIVERS

PAGE 10 OF 10



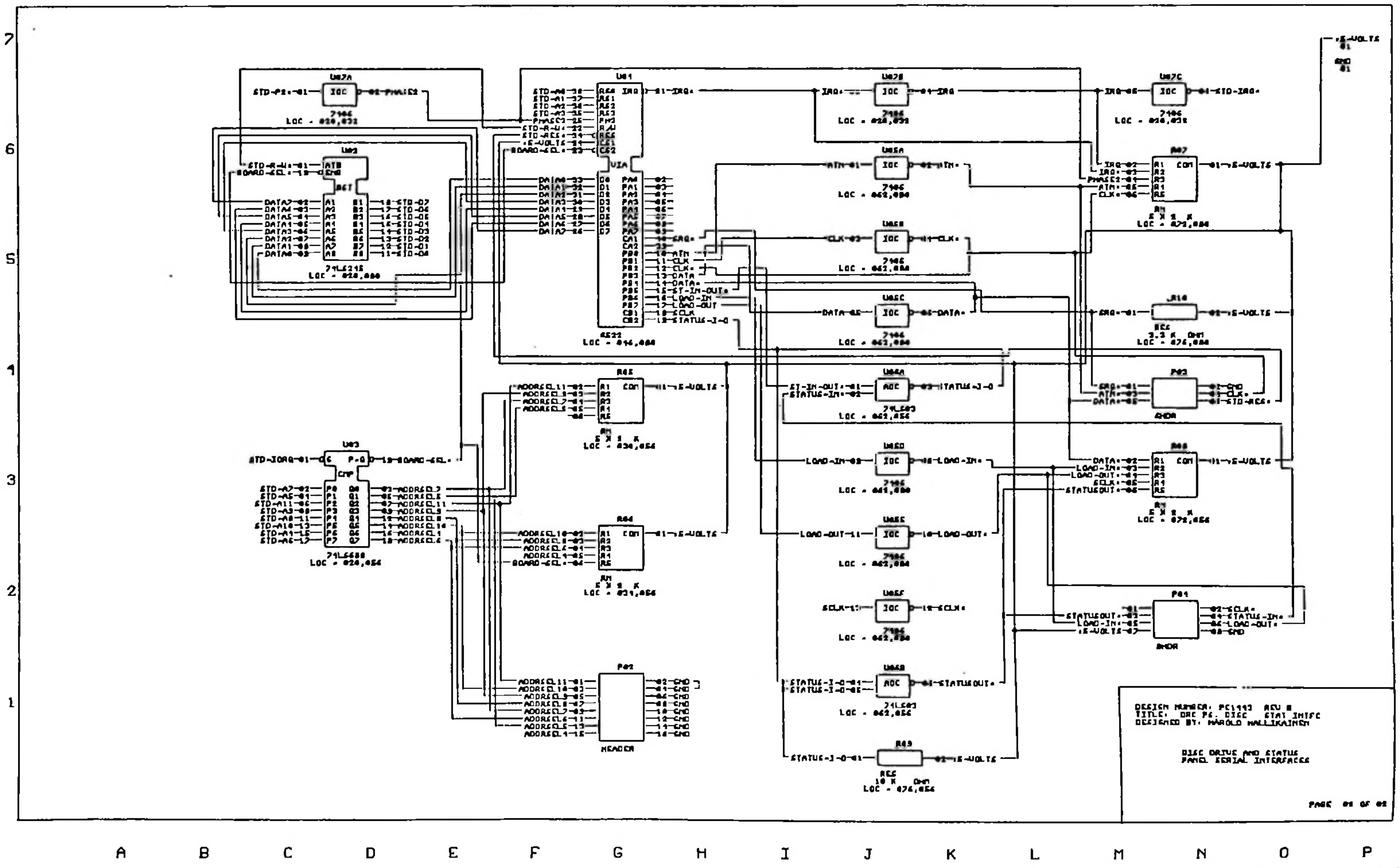
DESIGN NUMBER: PC1443 REV B
TITLE: DRC PS. DISC STAT INTFC
DESIGNED BY: HAROLD HALLIKAINEN

POWER SUPPLY INTERFACE
UPS INTERFACE
BATTERY CHARGER
UNDERVOLTAGE SHUTDOWN

PAGE 01 OF 01

A B C D E F G H I J K L M N O P

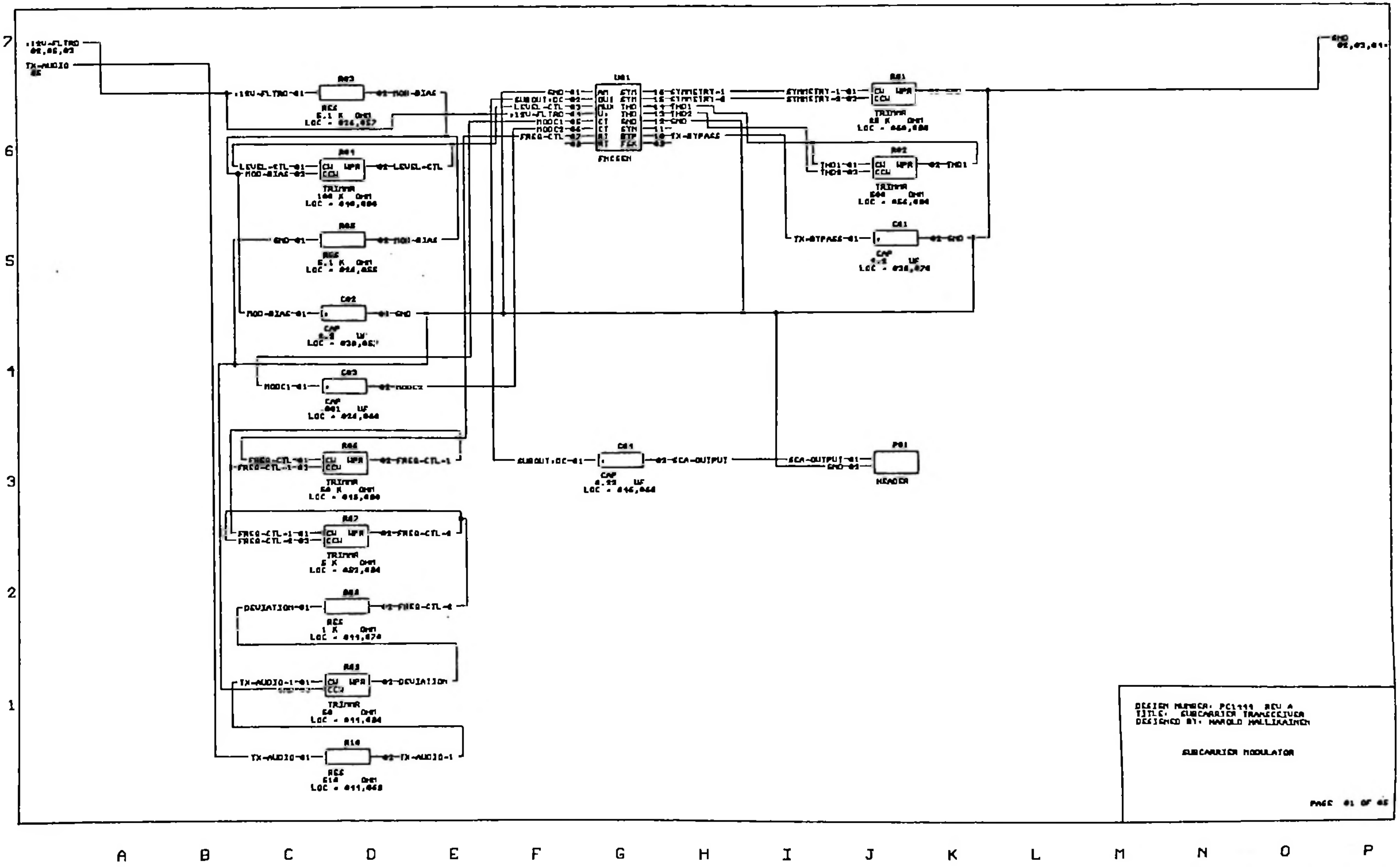
32-1



DESIGN NUMBER: PC1443 REV B
TITLE: DRC PS. DISC STAT INTFC
DESIGNED BY: HAROLD HALLIKAINEN

DISC DRIVE AND STATUS
PANEL SERIAL INTERFACES

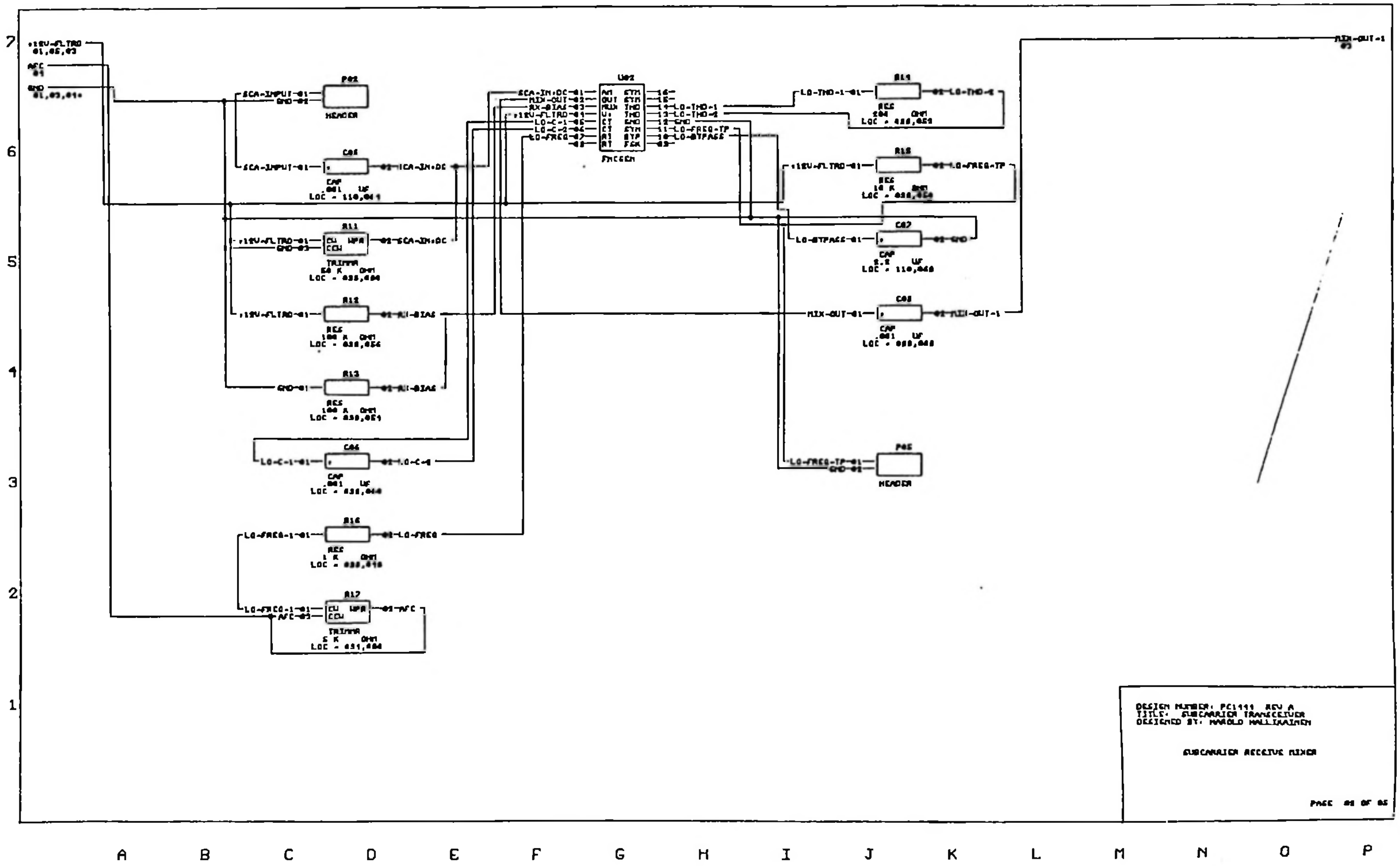
PAGE 01 OF 01



DESIGN NUMBER: PC1111 REV A
TITLE: SUBCARRIER TRANSCEIVER
DESIGNED BY: HAROLD HALLIKAINEN

SUBCARRIER MODULATOR

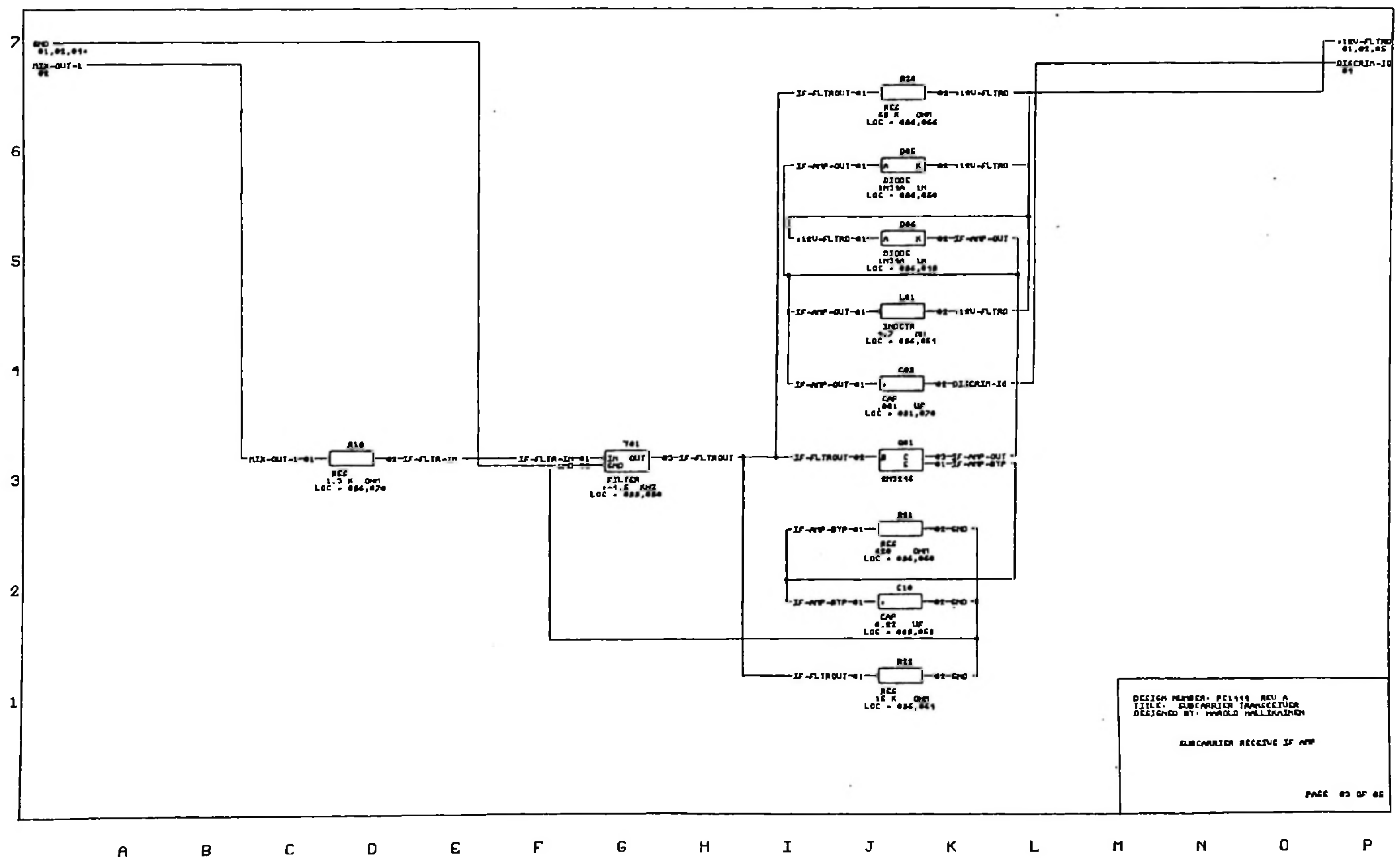
PAGE 01 OF 05



DESIGN NUMBER: PC1111 REV A
TITLE: SUBCARRIER TRANSCEIVER
DESIGNED BY: HAROLD HALLIKAINEN

SUBCARRIER RECEIVE MIXER

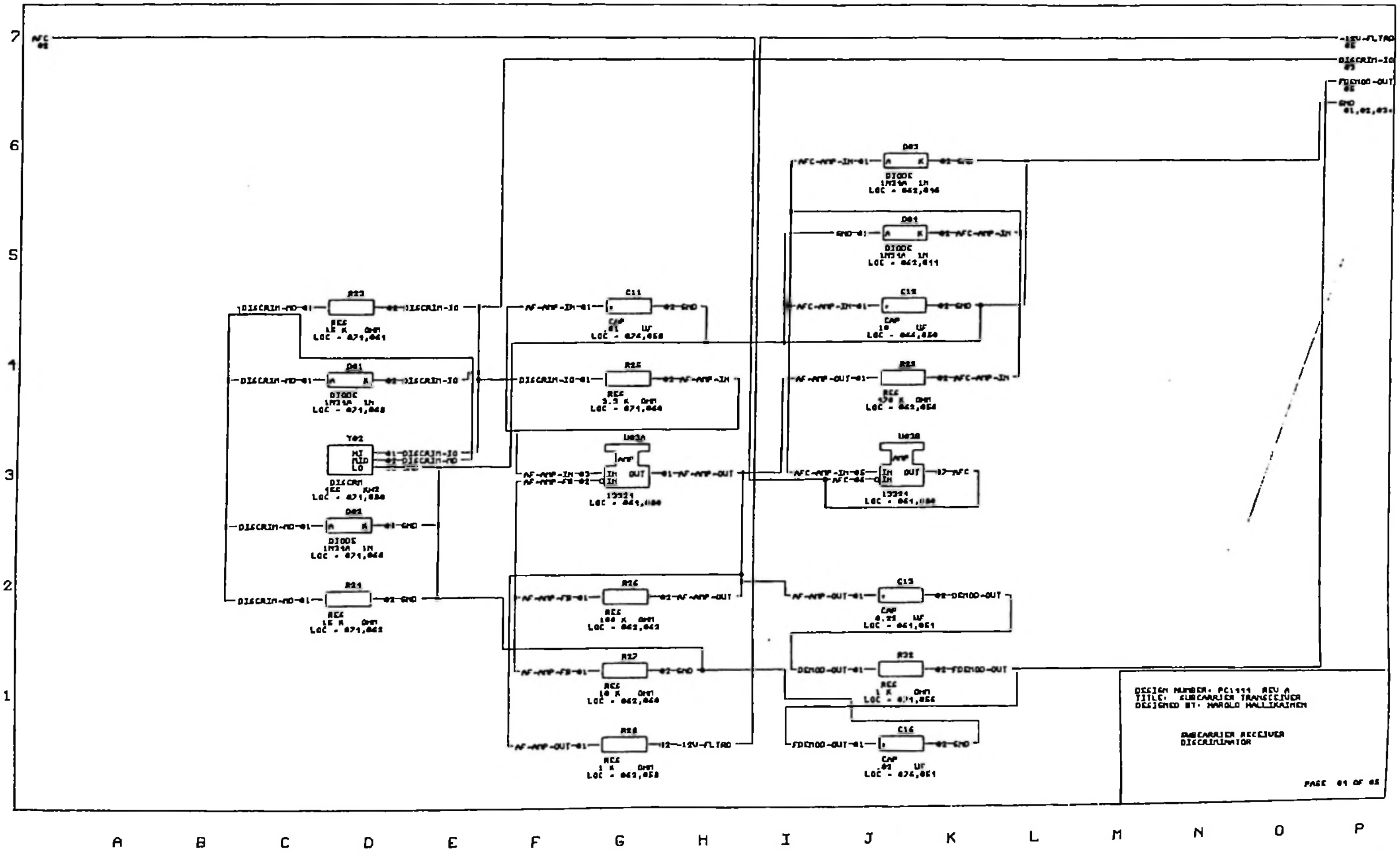
PAGE 05 OF 05



DESIGN NUMBER: PC1111 REV A
TITLE: SUBCARRIER TRANSCEIVER
DESIGNED BY: HAROLD HALLIKAINEN

SUBCARRIER RECEIVE IF AMP

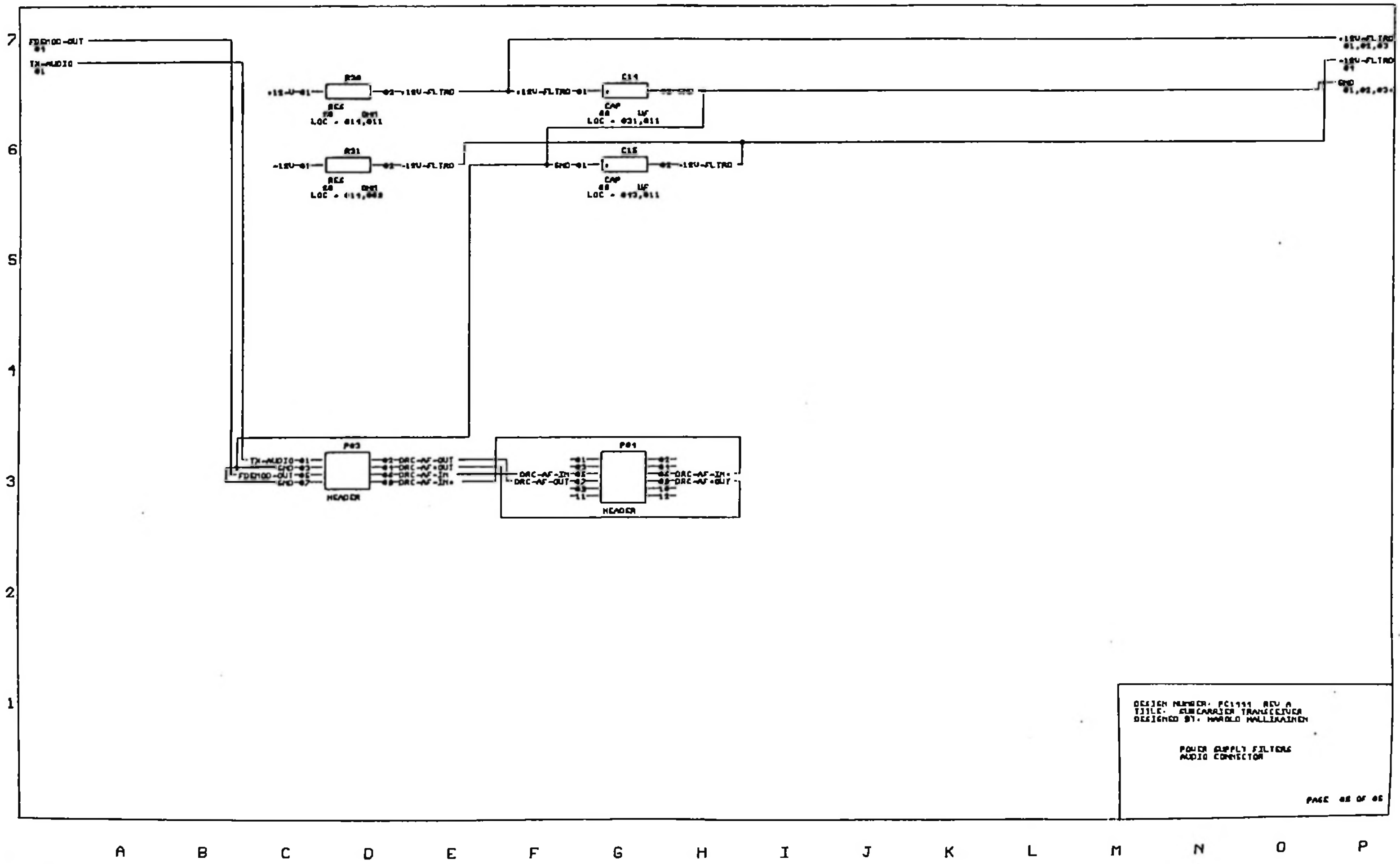
PAGE 03 OF 05



DESIGN NUMBER: PC1111 REV A
TITLE: SUBCARRIER TRANSCEIVER
DESIGNED BY: HAROLD HALLIKAINEN

SUBCARRIER RECEIVER
DISCRIMINATOR

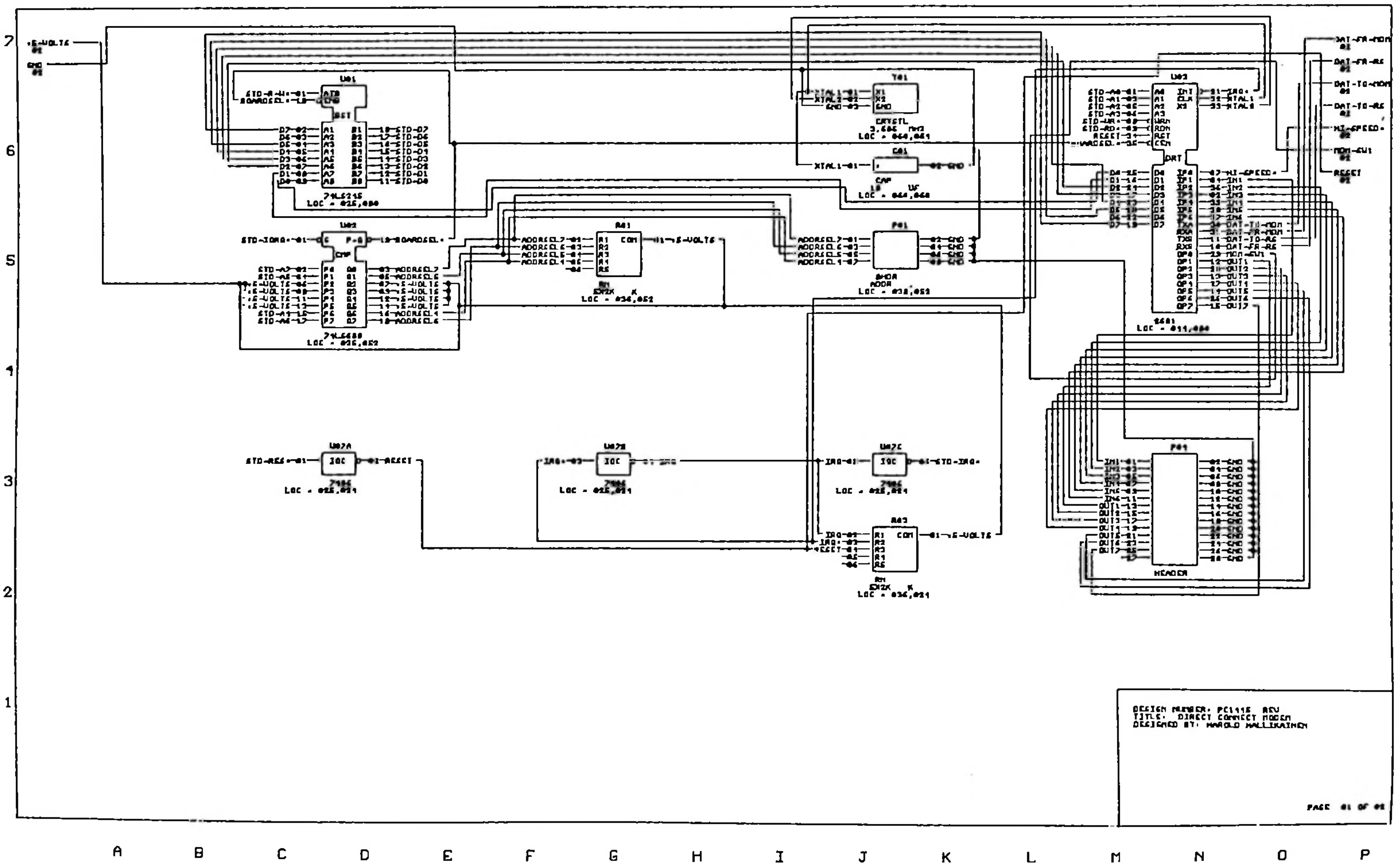
PAGE 01 OF 02



DESIGN NUMBER: PC1111 REV A
TITLE: SUBCARRIER TRANSCEIVER
DESIGNED BY: HAROLD HALLIKAINEN

POWER SUPPLY FILTERS
AUDIO CONNECTOR

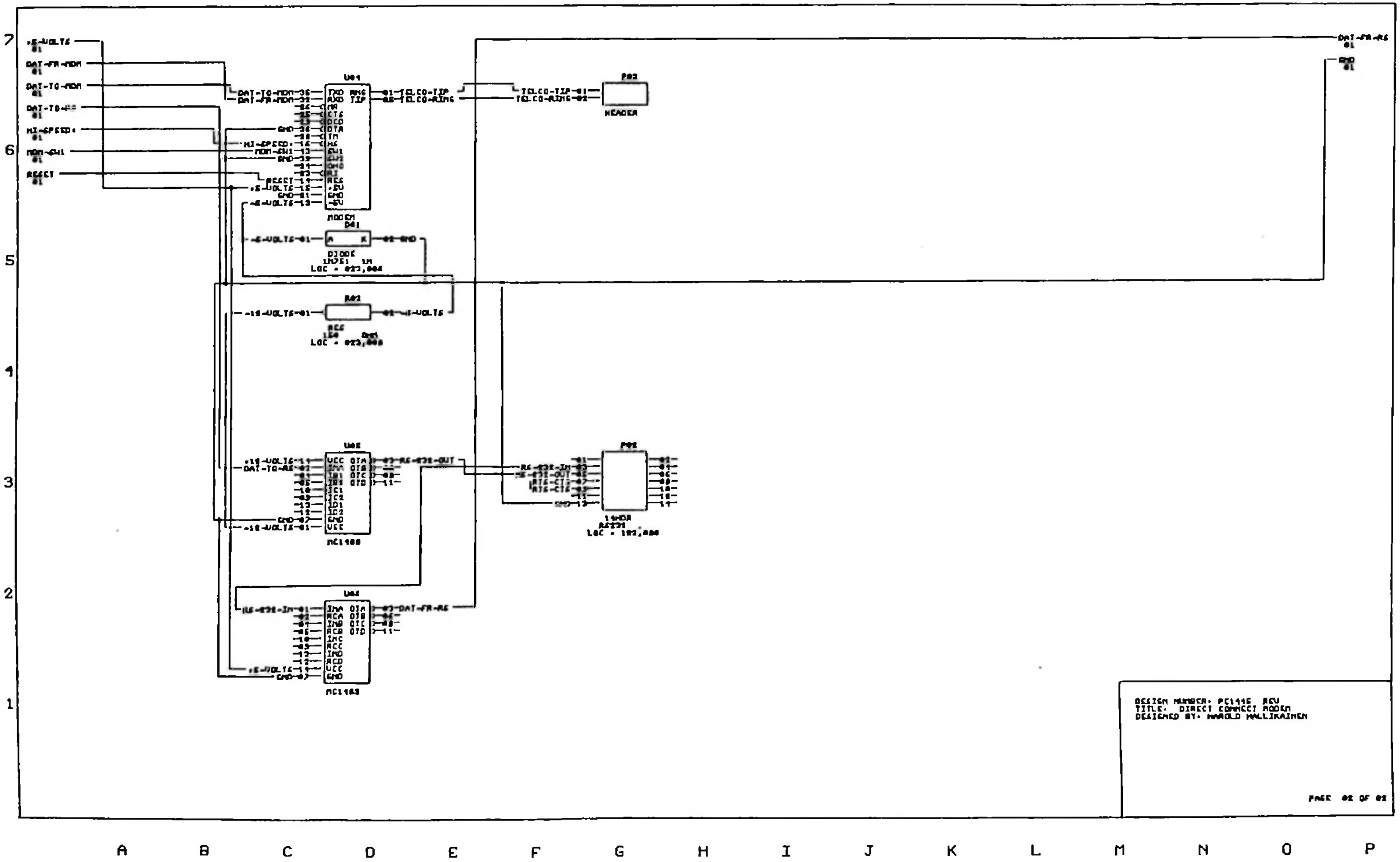
PAGE 05 OF 05



DESIGN NUMBER: PC1415 REV
TITLE: DIRECT CONNECT MODEM
DESIGNED BY: HAROLD HALLIKAINEN

PAGE 01 OF 01

34-1



DESIGN NUMBER: PC1445 REV
TITLE: DIRECT CONNECT MODEM
DESIGNED BY: HAROLD HALLIKAINEN

PAGE 02 OF 02

Bibliography

The following books, articles, and data sheets can be used as reference material with the DRC190.

Commodore Disk Interface

The Anatomy Of The 1541, Authors: Lothar Englisch, Norbert Szczepanowski, Edited by: Greg Dykema, Arnie Lee. Abacus Software, P.O. Box 7211, Grand Rapids, MI 49510 (612) 241-5510

Inside Commodore DOS, Richard Immers, Ph.D., Gerald G. Neufeld, Ph.D.. Datamost, 20660 Nordhoff Street, Chatsworth, CA 91311-6152, (818) 709-1202

VIC Revealed, Nick Hampshire, Hayden Book Company, Inc., Rochelle Park, NJ

Personal Computing on the VIC-20, Commodore Electronics, Inc.

Commodore 1541 Disk Drive User's Guide, Commodore Business Machines, Inc., 1200 Wilson Drive, West Chester, PA 19380

How The VIC/64 Serial Bus Works, Jim Butterfield, Compute magazine, July 1983

IEEE488 Interface

IEEE Standard Digital Interface for Programmable Instrumentation, The Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017

ZT 7488 IEEE488 Interface for STD Bus Operating Manual, Ziatech Corporation, 3433 Roberto Court, San Luis Obispo, CA 93401 (805) 541-0488

FCC Rules & Regulations

Code of Federal Regulations, 47CFR parts 15 (radiation by computing devices), 68 (connection of registed devices to the public switched telephone network), 73 (rules regarding operation of broadcast stations), part 74 (rules regarding auxiliary broadcast services), available from Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402

Semiconductor Devices

A large number of semiconductors are used in the DRC190. Most of these are standard semiconductos available from several manufacturers. The newer or somewhat unusual devices are listed below.

Exar Corporation, 750 Palomar Avenue, P.O. Box 62229, Sunnyvale, CA 94088, (408) 732-7970. XR2206 FSK Generator, XR2211 FSK Demodulator, Exar Modem Handbook.

Hitachi America, Ltd., 2210 O'Toole Ave., San Jose, CA 95131, (408) 942-1500.

HM6264LP-15 8 Kbyte static RAM

Intel Corporation, 3065 Bowers Avenue, Santa Clara, CA 95051, (408) 987-8080. 27256 32 Kbyte EPROM

Intersil, Inc., 10710 N. Tantau Avenue, Cupertino, CA 95014, (408) 996-5000. ICL7135CPI Analog to digital converter

Monolithic Memories, 2175 Mission College Blvd., Santa Clara, CA 95054, (408) 970-9700. MM6331-1 32x8 Tri-State Bipolar PROM (used as memory map decoder).

Motorola Semiconductor Products, 5005 East McDowell Road, Phoenix, AZ 85008. MC6802 Processor Data Sheet, M6800 Programming Reference Manual

RCA Solid State Division, Box 3200, Somerville, NJ 08876, (201) 685-6000. CD74HC166 Parallel In, Serial Out Shift Register, used in status panel

Rockwell International, Semiconductor Products Division, 4311 Jamboree Rd., P.O. Box C, Newport Beach, CA 92658-8902, (714) 833-4700. R6522 VIA used on processor and A/D boards.

Signetics Corporation, 811 East Arques Avenue, P.O. Box 409, Sunnyvale, CA 94086, (408) 739-7700. SCN2681AC1N40 DUART used on processor and direct connect modem boards.

Sprague Semiconductor Division, 115 Northeast Cutoff, Worcester, MA 01606. UCN-4821A serial input latched peripheral drivers, used in status panel

Texas Instruments, Semiconductor Group, P.O. Box 401560, Dallas, TX 75240, (214) 995-2011. SN7538E quad NAND peripheral driver, used on A/D board

Xicor, Inc., 851 Buckeye Court, Milpitas, CA 95035, (408) 946-6920. X2816A EEPROM used on processor board.

DRC190 Remote Control System
General Specifications

The DRC190 Digital Remote Control can be operated manually, or, with the addition of standard computer peripherals, will monitor, display, and print all relevant readings of a broadcast system. These readings include measured transmitter parameters such as plate voltage, plate current, antenna base current, common point current, FM forward and reflected power, and directional array loop currents and phases. When operating in the automatic mode, additional calculated parameters are available. These include output power calculated by the direct or indirect method, transmitter efficiency, directional array loop current ratios, deviation of loop current ratio from licensed values, and deviation of phase angles from licensed values. The instructions for the automatic operation of the DRC190 are written in Basic, allowing field modification of the system operation. Generally, automatic operation includes the display and logging of system parameters, alarming any that are out of set limits, and making adjustments that are available (power trim, day/night power/pattern select, etc.).

The DRC190 is based on the 6802 microprocessor operating in an STD bus system. Where available, standard STD bus cards are used to implement standard functions (such as system RAM). The hardware and firmware located at the different sites in the system are the same. Any "control" unit can be changed to a "transmitter" unit by plugging in analog to digital converter boards. Any unit can run a Basic controlling program, and all units have an RS232 port where a printer or CRT can be connected.

The DRC190 allows up to 100 sites to be used in the system. Such a system might consist of 100 "remote" sites and 1 control point. All units are on a "party line", with each unit transmitting a request for data when it needs it, or responding to a request for data when it has data available. All units communicate on the same frequency using Bell 202 standard FSK 1200 bit per second coding. The firmware includes a multiple access anti-contention system that assigns each unit a time slot when it is allowed to start transmitting data, if it has any data. If no data needs to be transmitted, no carrier is brought up, and no data transmitted. This absence of data is detected by other units in the system. Permission to transmit is then passed to the next unit in the system. On a system where it is not necessary to wait for external key-up and squelch delays (such as wire line or dedicated subcarrier), the time slot allocated to each unit typically runs 50 mS. If a site starts transmitting data in its allocated 50 mS time slot, the advancing of the site counter in all units is inhibited until the data transmission is completed. If no data is transmitted in the 50 mS time slot, the site counter at each site is advanced, granting permission to the next site to transmit, if it has any data. This forms a modified token passing system. The absence of transmitted data acts as a token passing permission to transmit to the next unit in the system. The use of absence of data as a token yields several improvements over typical token passing systems. Since no data is to be transmitted, the carrier bring up and shut down delays can be eliminated from the system when no data is to be transmitted. This results in higher speed token passing. Since the absence of data is the token, there is no danger of the token being "smashed" or lost due to a data error. This improves system reliability and reduces software complexity. When a site fails, it transmits no data, which is equivalent to always passing the token. This avoids system failures and system reconfiguration software requirements.

A system is specified using a "site specification form". This form refers to the various components that can be used in building a DRC190 unit. These components are discussed below:

Main Frame

The main frame includes the system cabinet, a switching power supply, a 32 character backlit alphanumeric liquid crystal display, a front panel speaker, and a 16 key membrane keyboard. All system setup and manual operation are performed using the keyboard and display. The speaker provides acoustic feedback for keyboard operation and serves as an intercom between sites.

Standard Processor

The standard processor includes the processor itself, 32 Kbytes of EPROM, 8 Kbytes of RAM, 2 Kbytes of EEPROM, a real time clock, front panel keyboard and display interface, a 1200 bit per second half duplex modem with intercom and Morse Code identification circuitry, and an RS232 port. The EPROM holds the system firmware (the software that tells the system how to be a DRC190). This firmware includes interface software for all the hardware (keyboards, displays, modems, A/D converters, etc.) and the Basic interpreter. The Basic interpreter is an extended Microsoft 8 Kbyte Basic for the 6800 series of processors.

The EEPROM holds the site specification data (site number, communications speeds, etc.) and calibration data. Each metering channel is assigned ten bytes of EEPROM. These ten bytes hold a three character label for the parameter (such as ICP for common point current) and a two character units symbol (such as KV for kilovolts). Also stored in these ten bytes are a conversion curve (linear or square law), a sample delay, and a scaling factor. The sample delay is programmable in thirds of seconds up to five seconds to allow for settling time of measuring equipment, such as antenna monitors. The scaling factor is determined by the DRC190 firmware during calibration. The desired indication is divided by the A/D conversion (or the squared A/D conversion if in square law mode) to yield the scaling factor. During normal operation, this scaling factor is multiplied by the A/D conversion to yield an appropriate indication based on the sample. This results in a two point linear calibration. It is assumed that zero volts of sample represents a parameter of zero. The calibration point establishes the slope of the line representing the relationship between the parameter and the sample voltage. The calibration uses the linear equation $Y= mX + B$ where $B=0$ (y intercept is zero).

About 1 Kbyte of EEPROM is free to store a user Basic program. This program is automatically loaded and run on system reset. This program could be a simple logging program, or could be a "boot" program that loads another program from disk or from another bank of memory (see the RAM board description).

The processor board includes various features to make it fault tolerant. All unused memory is filled with software interrupt instructions. Should the processor try to execute the contents of this unused memory, the system will reset and begin execution from the reset state. The processor board includes a watchdog timer. At various places in the Basic interpreter, the watchdog timer is reset. Should the program lose control of the processor, the timer will not be reset. When the timer times out (about 40 seconds), the system resets and starts execution from the reset state. Finally, should there be a processor crash, it would be possible for the out of control processor to damage calibration data in the EEPROM prior to the watchdog timer timing out.

For this reason, the EEPROM is write protected. The processor must go through a several step procedure before it can write to the EEPROM. It is unlikely that such a procedure would be successfully completed by an out of control processor. These features have been included to maximize the reliability of the DRC190.

The processor board includes a Morse code indentifier. This allows the DRC190 to identify a radio transmitter (TRL) that it is operating using International Morse Code. This code is sent at 20 words per minute using a 750 Hz tone. The frequency of the ID can be varied between an ID message every minute to an ID every 99 minutes. The ID frequency and the ID message can be programmed through the front panel keyboard and display. The ID is stored in EEPROM, preserving it during power failure.

The DRC190 processor includes the hardware to drive IEEE488 controlled test equipment. The firmware to drive the IEEE488 port has not been completed at this writing. Registered system users will receive the firmware update at no charge when it is available.

Dual Audio Option

The dual audio option provides two separate inputs and outputs for the audio data. The audio outputs containing the FSK data leaving this site. The audio inputs are summed and presented to the FSK demodulator. Dual audio is normally used at intermediate two way microwave sites. In a North/South system, one audio output would drive the North bound transmitter, the other would drive the South bound transmitter. One audio input would receive data from the North bound receiver. The other would receive data from the South bound receiver. In this manner, all units in the system can hear all other units, yet we do not introduce crosstalk between North and South paths.

Battery Backup

The battery backup option adds an Uninterruptable Power Supply, a 12 volt rechargable battery, and battery charging circuitry to the system. The UPS module converts the 12 volts DC from the battery to +/- 100 volts DC, which replaces the output of the off-line rectifier of the switching power supply during a power failure. During a power failure, the battery charging circuit is shut down. On power recovery, the battery charger returns to operation. The Battery Backup runs the entire DRC unit for about 30 minutes, instead of merely backing up the system memory.

Disk/Status I/O

These additional serial I/O lines on the power supply interface board allow the DRC190 to communicate with a disk drive and a status transceiver. The disk drive allows program storage on a 5.25 inch floppy diskette. The status transceiver transmits and receives digital status information between sites.

A/D Converter

The analog to digital converter board takes analog samples from the external equipment. This analog sample can be floating with a maximum common mode voltage of +/- 100 volts peak and a maximum differential voltage of +/- 2 volts DC. If the differential voltage exceeds 2 volts, resistors can be plugged into the A/D board to form a voltage divider that is applied to all channels on that board. The sample voltage must be DC. Reed relays are used to select which sample voltage is presented to the analog to digital converter.

The analog to digital converter is a 4.5 digit (maximum count of 19999) modified dual slope converter. The converter completes three conversions per second. It is driven by a temperature controlled precision voltage reference and a precision voltage divider. Dual slope A/D converters are noted for their extremely high linearity and resolution. The A/D used in the DRC190 has a resolution of 100 uV of sample voltage.

The A/D converter board also presents 10 Raise and 10 Lower control outputs to the rear panel. These outputs are enabled by selecting the appropriate A/D channel and pressing the R or L key on the front panel keyboard. They can also be reached using the RAISE (S,C) or LOWER (S,C) commands from Basic, where S is the site number and C is the channel number. Raise or Lower commands from specific sites can be locked out by programming the system during initial setup through the front panel.

The A/D converter also provides 10 "channel selected" outputs. These outputs are enabled as each channel is selected. These outputs are generally used to drive tower select relays on antenna monitors.

The A/D converter also provides a "fail safe" output. This output is active as along as all required sites have been "heard from" in the past minutes. The "required sites" are those that the user has programmed into the system during initial setup through the front panel keyboard. The fail safe firmware keeps a record of all sites heard from in the past 30 seconds. If the fail safe timer reaches 30 seconds and all required sites have been heard from, the timer resets. If not all required sites have been heard from, the unit sends an "all call" status request. If all required sites respond, the fail safe timer is reset. If the required sites do not respond, the unit keeps sending the all call request every five seconds. It the fail safe timer reaches 60 seconds, the fail safe output is disabled. This can be used to discontinue the operation of a transmitter, if desired.

All the A/D board outputs (Raise, Lower, Channel Selected, and Fail Safe) are open collector outputs that are pulled to ground when active. The open circuit voltage must not exceed +30 volts and the short circuit current must not exceed 500 mA.

Each A/D board provides ten channels of metering. Up to ten boards (for 100 channels of metering) may be plugged into each DRC190 main frame.

32 Channel Status Transceiver

The status transceiver accepts 32 status inputs and provides 32 status outputs. The status transceiver appears on the right half of the front panel of the DRC190. 32 LEDs show the current received status. These LEDs are in four columns of eight. The vertical spacing is 0.400 inches, allowing the use of 0.375 inch embossed tape to label the LEDs.

The front panel LEDs and the rear panel status output lines reflect the received status. The received status is received from a site selected by programming jumpers on the status transceiver board. This allows the LEDs to continuously display the status of one of the sites in the system (00 to 99). If the site select jumpers are set to 100, the displayed status follows the site selected by the front panel keyboard and LCD. If there are several sites that have the same meaning to the various status indications, a common label can be attached to each LED. For example, if LED 0 is lit and the LCD is showing a parameter from site 1, this may indicate that the main transmitter is on at site 1.

Status of all sites is available to Basic at all sites using the function STATUS (S,C), where S is site and C is channel. This returns a -1 if status is true, and a 0 if it is false.

The condition of the status LEDs is also available on the rear panel of the DRC190. These 32 outputs (0 to 31) are open collector outputs with a maximum open circuit voltage of 40 volts and a maximum sink current of 100 mA.

Bell 212 Modem Board

This board includes an FCC approved direct connect, auto dial, auto answer 1200/300 bit per second modem and an extra RS232 port. The modem and RS232 port are accessible to the local Basic program, allowing dial up operation of the system. Typical dial up applications include: complete dial up control where dedicated circuits are not available, paging an engineer on system malfunction and giving the engineer access to system parameters, providing routine readings and alarm reporting to a central location. The modem supports both tone and pulse dialing. The modem speed is selectable as 1200 or 300 bits per second under software control.

64 Kbyte RAM Board

This board expands system RAM beyond the 8 Kbytes on the processor board. The 64 Kbytes are broken into two banks of 32 Kbytes, called banks 0 and 2. Bank 0 is used for holding an expanded Basic program, variable storage and temporary storage (system stack, etc.). Bank 2 is battery backed. This can provide non-volatile storage of the Basic program that is down-loaded to bank 0 on system reset.

Subcarrier Transceiver

The subcarrier transceiver board generates and demodulates FM subcarriers between 20 and 200 KHz. The generator portion uses a single chip function generator. The generator portion includes controls for coarse frequency adjust, fine frequency adjust, deviation, waveform symmetry, and sine shaper distortion.

The demodulator portion of the subcarrier transceiver board uses the superhetrodyne principle. The baseband is up-converted to 455 KHz. The upconverted subcarrier is filtered from the remainder of the baseband using a ceramic filter. The Intermediate Frequency is then amplified and presented to a ceramic discriminator. The audio output of the discriminator is amplified and presented to the DRC190 processor board for FSK demodulation. The DC component of the discriminator output is applied to the local oscillator to form an Automatic Frequency Control.

Note that the minimum acceptable subcarrier level is about 150 mV P-P. This is generally available from the multiplex output of STL systems, but is not available from the baseband output of most FM modulation monitors. To recover SCA data from FM stations, an SCA monitor or SCA receiver is suggested.

CRT Terminals

Most standard CRT terminals can be used with the DRC190. The following terminals are available from Hallikainen & Friends.

Fluke 1020

This terminal has a touch screen overlay allowing operator interaction with the DRC190 without the use of a keyboard. This considerably simplifies the operator interface, as only the options available to the operator at a specific time are presented. This compares favorably with dedicated function keys or the use of alphanumeric keys for making menu selections. With the touch-screen, the operator touches the screen to make the choice. This is the ultimate "soft key". Further information on this terminal is attached.

Informer 203-100

This full featured terminal has a 9 inch display that keeps the terminal from taking over the desk. The terminal includes a printer port that can be used in driving a logging printer. The terminal is available with an optional Bell 212A built-in modem that includes auto-dial, auto-answer, and automatic password exchange. This makes the terminal ideal for dial-up applications.

ITT Qume QVT101

This low cost terminal satisfies many DRC190 applications. It includes a printer port for driving a logging printer. It is available with green or amber screens. Further data on this terminal is attached.

Printer

Most RS232 printers can be used with the DRC190. Hallikainen & Friends is presently supplying the Brother M1509 dot matrix printer. This 180 character per second dot matrix printer includes an RS232 port for connection to the DRC190 or to the printer port on a CRT terminal. The printer accepts paper up to 16 inches wide. Further data on this printer is attached.

Disk Drive

The DRC190 uses the Commodore 1541 disk drive for program storage. This low cost disk drives include its own processor and operating system firmware. Up to five disk drives can be connected to each DRC190 equipped with a disk drive interface. Each disk holds up to 167,000 bytes of program. At this writing, the disk drive can be used only for program storage.

UHF Telemetry/Control Radio Transceiver

Hallikainen & Friends currently supplies the Standard Communications FX-60U 450 MHz transceiver for TRL service. This transceiver is modified to comply with FCC part 74 group P specifications (1.5 KHz deviation, 10 KHz channel spacing). The transmitter is enabled only when a particular site has data or a command to send. This allows all sites in the system to share the same radio frequency, resulting in less spectrum usage. In addition, any site can communicate with any other site, allowing operators at the AM site to read meters at the FM site.

Instruction Manual

A detailed instruction manual is included with each DRC190 main frame. This manual can also be purchased separately for $25 per copy.

For further information, or to place an order, call

Hallikainen & Friends, Inc.
141 Suburban Road, Building E4
San Luis Obispo, CA 93401-7590
(805) 541-0200

DRC190 Site Specification Form

Specified by: ____________

Company: ____________

Phone: ____________

PO#: ____________ H&F SC#: ____________

Site Number: ________ Serial Number: ________

Qty	[Qty Range]	Description	Cost ea	Total Cost
1	[1]	DRC190 Main Frame (0000-1900)	1,600	1,600

Processor Options

Choose one only

Qty	[Qty Range]	Description	Cost ea	Total Cost
___	[0..1]	Standard Processor (0010-1442)	0	0
___	[0..1]	Dual Audio (0012-1442)	91	_____

Power, Special I/O Interface Options

Qty	[Qty Range]	Description	Cost ea	Total Cost
___	[0..1]	Standard (0010-1443)	0	0
___	[0..1]	Battery Backup (0011-1443, 4000-0004, 4000-0005)	207	_____
___	[0..1]	Disc/Status I/O (0012-1443)	100	_____
___	[0..1]	Battery Backup & Disc/Status I/O (0013-1443, 4000-0004, 4000-0005)	307	_____

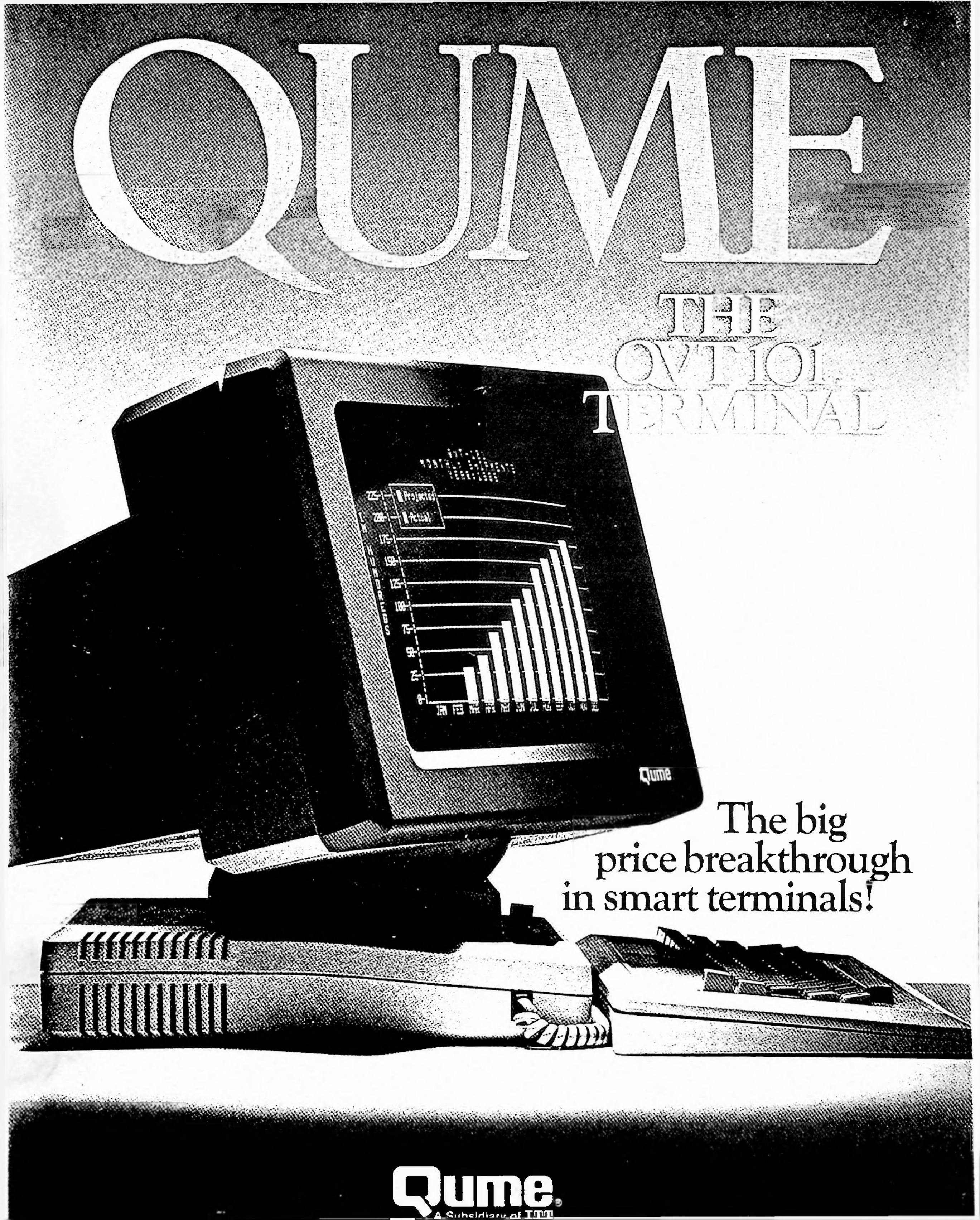
A/D, Memory, Subcarrier Boards

___	[0..9]	Ten Channel A/D Boards (0010-1441)	600	___
___	[0..1]	Bell 212 Modem Board (0010-1445)	600	___
___	[0..1]	32 Channel Status Transceiver (0010-1449)	500	___
___	[0..1]	64K RAM Board (2000-6440)	500	___
___	[0..1]	Subcarrier Transceiver (0010-1444) RX Freq: _____ KHz TX Freq: _____ KHz	300	___

External I/O Equipment

___	[0..1]	Fluke 1020 Touch Screen CRT (2000-1020)	1,600	___
___	[0..1]	Informer 203-100 CRT Terminal (2000-2031)	895	___
___	[0..1]	Informer 203-100-212 CRT Terminal with Modem (2000-2032)	1,495	___
___	[0..1]	Qume QVT101 CRT Terminal (2000-0101)	395	___
___	[0..1]	Brother M1509 Printer (2000-1509)	549	___
___	[0..2]	Commodore 1541 Disk Drive (2000-1541)	300	___
___	[0..1]	Standard FX60-U Control/ Metering Radio Transceiver Freq: __________ MHz Call Sign:__________	1,600	___

TOTAL COST THIS SITE>>>>>>>>>>> $______


QUME
THE
QVT 101
TERMINAL
Projected
Actual
IN HUNDREDS
225
200
175
150
125
100
75
50
25
0
JAN
FEB
Qume
The big price breakthrough in smart terminals!
Qume
A Subsidiary of ITT

QVT 101™

The best buy in VDT history.

Introducing Qume's new QVT 101™, the lowest-priced smart terminal in history. All these features are standard: Full emulation of the Hazeltine 1500, Lear-Siegler ADM 3A/5 and TeleVideo 910 • Block mode • 14-inch non-glare screen • Bidirectional printer port • 12 host or user-programmable functions • 1 to 128 characters dynamically allocatable to a single function key • Cursor keys can double as an additional 4 programmable function keys • Double-injection key caps • Full compatibility with Qume's popular QVT 102™.

Only Qume can offer such unheard-of value, with smarter design, smarter sourcing of components, and greater resources behind us. Only Qume provides a one-year warranty from the date of end-user purchase — double the warranty of most other terminals.* And only Qume has a reliability record you can't beat at any price: a warranty return rate of under 2 percent.

Display Format
24 lines x 80 characters
25th status/set-up line (user selectable)

Character Formation
7x 11 matrix in a 9 x 12 cell, with 2 dot decenders

Displayed Character Set
96 ASCII characters, 15 graphic symbols, and 32 control character symbols

Editing
Cursor: up, down, left, right, and home. Character/line insert and delete, erase to end of line/field/page, tab, back tab, field tab, field back tab

Communications Interface
EIA RS232-C. Options: 20 mA current loop, EIA RS422

Communication Protocols
DTR, X-ON/X-OFF, and none

Communication Modes
Full or half duplex, block/line, block/page; 7 or 8 data bits

Baud Rates
16 selections from 50 to 19.2K

Auxiliary Port
Bidirectional EIA RS232-C, partial/full screen/line copy, transparent

Screen
Tilt/swivel 14-inch diagonal standard non-glare green (optional non-glare amber)

Character Attributes
Blink, blank, normal video, reverse video, underline, half intensity

Keyboard
Detached, adjustable-tilt, low-profile (DIN standard), alphanumeric keys, 14-key numeric pad, 12 host or user-programmable functions, 4 cursor keys double as additional programmable function keys, selectable auto-repeat and key click. Print, setup, and no-scroll keys.

Fields
Protected and Unprotected fields

Parity
Odd, even, mark, space, none

Screen-savers
Screen shuts off after 15 minutes of inactivity without data loss (selectable)

Emulations
Hazeltine 1500, Lear Siegler ADM-3A/5, TeleVideo 910

Set-Up Mode
Menu style, preserved in non-volatile memory

Power Requirements
120 VAC, 0.5 A, 45 W, 60 Hz

Dimensions
Keyboard 1.5" (H) X 18" (W) X 8" (D)
Display 14¾" (H) X 13½" (W) X 12½" (D)

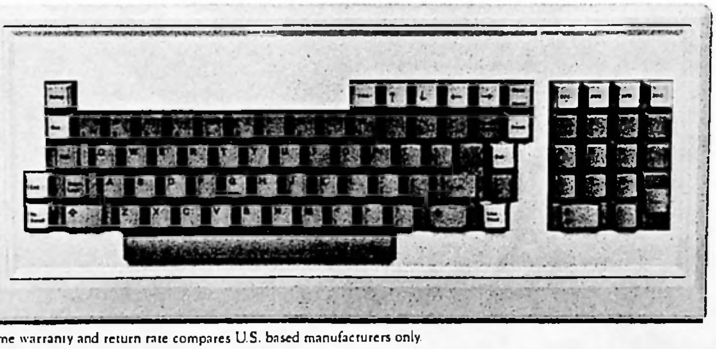
Weight
Display 19 lbs. 2 oz., Keyboard 3 lbs.

Options
Amber phosphor screen
Foreign language kits
EIA RS422 main port interface
20 mA current loop (passive and active)

Hazeltine 1500 is a registered trademark of Hazeltine Corp. (Esprit Systems Inc.). Lear Siegler ADM 3A/5 is a registered trademark of Lear Siegler, Inc. TeleVideo and TeleVideo 910 are registered trademarks of TeleVideo Systems, Inc.

Command Codes (partial list)

Cursor home	Ctrl
Cursor right	CtrlL
Cursor left	CtrlH
Cursor up	CtrlK
Cursor down	CtrlJ
Clear screen	CtrlZ
Clear from cursor to end of line	Esc T
Clear from cursor to end of screen with spaces	Esc Y
Clear all foreground	Esc ;
Line insert	Esc E
Line delete	Esc R
Keyboard lock	Esc #
Keyboard unlock	Esc "
Bell ring	CtrlG
Column tab	CtrlI
Address cursor (row/column)	Esc=(nn)
Read cursor (row/column)	Esc?(nn)
Status line display	Esc }
Status line suppress	Esc {
Monitor mode on	Esc U
Monitor mode off	Esc X or Esc u
Print transparent on	CtrlR
Print transparent off	CtrlT
Half intensity	Esc)
Full intensity	Esc (
Graphic mode on	Esc $
Graphic mode off	Esc %
Auxiliary port on	Esc @
Auxiliary port off	Esc A
New line	Ctrl__
Back tab	Esc I
Clear screen to nulls	Esc *
Display control character equivalent	Esc F n
Load cursor line	Esc [n
Load cursor column	Esc] n
Print entire screen	Esc P
Print from top to cursor	Esc N
Print from cursor to end	Esc O
Tab set	Esc 1
Tab clear	Esc 2
Tab all clear	Esc 3
Autoscroll on/off	Esc H
Self test	Esc V
Protect mode on	Esc &
Protect mode off	Esc '
Send line full intensity only	Esc 4
Send page full intensity only	Esc 5
Send line all	Esc 6
Send page all	Esc 7
Clear foreground to nulls	Esc :
Block/page on	Esc B
Conversational mode on	Esc C
Block/line on	Esc D
Character insert	Esc Q
Character delete	Esc W
Termination character select	Esc x N
Print line	Esc p
Program cursor keys	Esc Mnabcd
Program function keys	Esc|P1P2Data EM
Write saved values into temp RAM	Esc !0
Write default into temp RAM	Esc !1
Write temp RAM into non-volatile RAM	Esc !2



Authorized Qume Distributor/Dealer

Keyboard adjusts to three height levels. Available in French, German and U.K. configurations.

*Qume warranty and return rate compares U.S. based manufacturers only.

Qume Corporation
2350 Qume Drive
San Jose, CA 95131
(408) 920-2500

Qume Corporation
7601 North Federal Hwy.
Suite B215
Boca Raton, FL 33431
(305) 997-8667

Qume Corporation
2354 Hassel Road
Hoffman Estates, IL 60196
(312) 490-9320

Qume Corporation
6133 Bristol Parkway
Suite 150
Culver City, CA 90230
(213) 410-1458

Qume Corporation
19 Crosby Drive
Bedford, MA 01730
(617) 275-3200

Qume Corporation
1425 Greenway Drive
Suite 300
Irving, TX 75038
(214) 659-0745

Qume Corporation
5335 Far Hills Avenue
Suite 107
Dayton, OH 45429
(513) 439-0469

Qume Corporation
20 Mayfield Avenue
Edison, NJ 08837
(201) 225-5005

Qume Corporation
6287 Memorial Drive
Suite F
Stone Mountain, GA 30083
(404) 469-3900

Qume®
A Subsidiary of ITT
Qume Corporation, 2350 Qume Drive
San Jose, California 95131
(800) 223-2479

Qume Corporation
Carriage Executive
2659 W. Guadalupe Road
Suite C220
Mesa, AZ 85202
(602) 820-6226

Qume Corporation
Westpark Office Building
Building #1, Suite 235
8140 Mopac Expressway
Austin, TX 78759
(512) 346-9764

CANADA
QUME Canadian Office
207 Place Frontenac
Pte. Claire
Quebec, Canada H9R4Z7
(514) 695-3837

ENGLAND
QUME (UK) Ltd
Qume House
Parkway, Newbury
Berkshire, RG 13 1EE
PH: 635-31-400

FRANCE
QUME France
20 rue Thiers
92100 Boulogne, France
PH: (1) 6082334

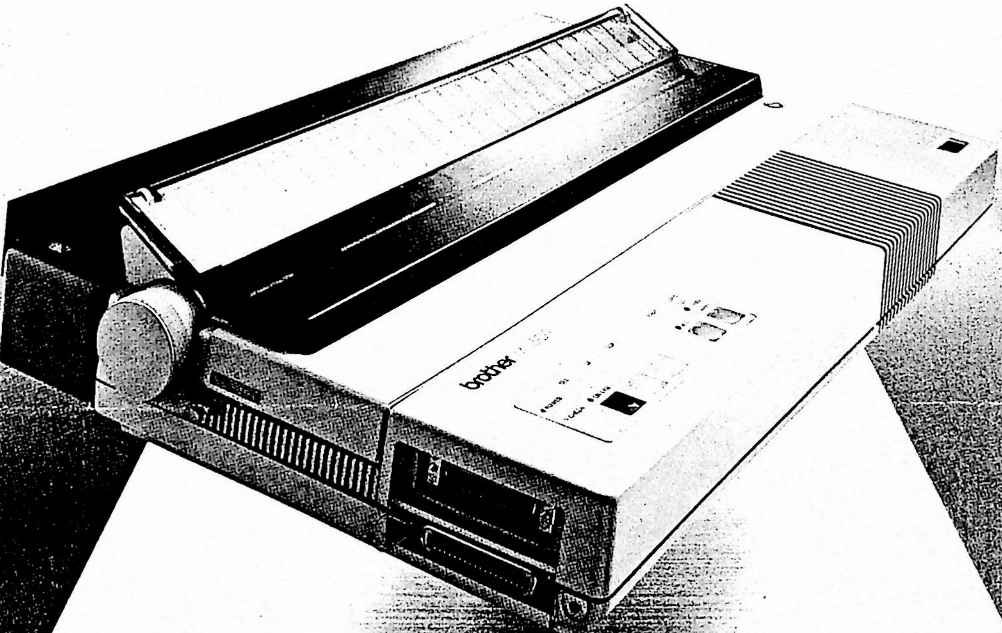
GERMANY
QUME GmbH
Schiess-Strasse 55
4000 Dusseldorf 11
West Germany
PH: (211) 59798-0

brother®
We put our reputation on paper.

M-1509

SERIAL IMPACT DOT MATRIX PRINTER

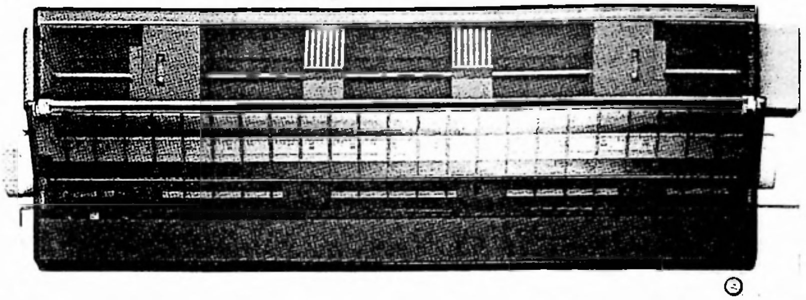
Streamlined Printing Efficiency

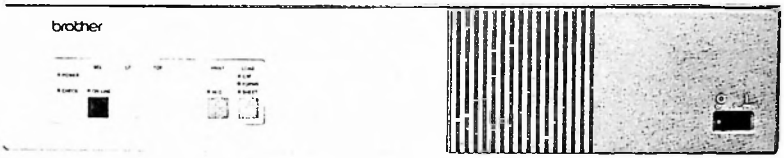


- High speed output (180 CPS)
- Near Letter Quality printing (45 CPS)
- Variety of printing styles and functions
- Large buffer memory
- Optional cut sheet feeder
- Compatible with IBM PC® and Epson FX-100+®
- One-inch paper cut function
- Slim, compact, and attractively styled
- Quiet operation

brother® M-1509

Streamlined Printing Efficiency





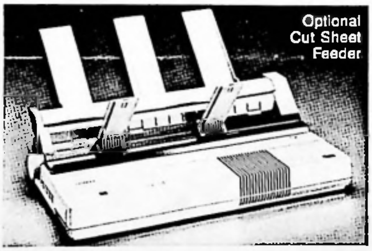
The Brother M-1509 is a serial impact dot matrix printer designed to offer reliability, economy, and easy operability to users of personal or small business computers. Normal print speed is a fast 180 cps, but the M-1509 also provides handsome Near Letter Quality printing at 45 cps. A variety of character sets are available, and the high-density bit image capability will produce convincing graphics. The M-1509 incorporates a number of cost- and labor-saving features, and is exceptionally slim, light, and quiet. A printer that will quickly prove its worth in either home or office.

High Speed Printing, Large Capacity Memory

In many offices, work slows down when the printer goes into operation, but not when it's an M-1509. With logic-seeking bi-directional printing capability, printing speed is a rapid 180 cps. And thanks to a 3 Kbyte input data buffer, the operator will nearly always be able to go straight to the next task.

Near Letter Quality and Graphics

The M-1509 has eliminated the need for a second printer for those special jobs where quality is the prime requirement. A handsome Near Letter Quality font (Prestige) will upgrade the business image with clear and clean correspondence. One touch of the NLQ button is all it takes. Three other NLQ fonts can be optionally selected. And seven different levels of bit image mode produce attractive charts and graphs.



Auto Paper Loading

The M-1509 does not require manual paper insertion. Simply place the paper behind the platen and push the TOF button—the paper will automatically be inserted and forwarded to the correct position. And despite the M-1509's compact size, you can use 16-1/2" width cut sheets or 16" width fanfold paper.

One-Inch Paper Cut

Another convenient function is provided by the push-up pin tractor, which lets you separate copy immediately, one inch after printing stops, saving both time and paper.

IBM/Epson Compatibility

The M-1509 is equipped with an IBM PC® printer compatible character set and control codes. A character set and control codes compatible with the Epson FX-100+® are also incorporated for easy interfacing with the vast range of personal computers that can be used with either of these two printers.

Quiet and Unassuming, but a Hard Worker

The M-1509 is designed with people in mind. It purrs along quietly, and is light and slim—easily moved and positioned. Controls are simple and conveniently grouped. In terms of both performance and overall operation, the M-1509 is a printer every operator will love.

Options for Greater Versatility

A cut sheet feeder (SF-40) is available for smooth, automatic paper loading (without having to remove the pin feed tractor). And two NLQ font boards provide an extra choice of type styles and memory. The LQ-100 offers Anelia PS, Gothic and Quadro, while the LQ-200 has Anelia PS and Gothic, plus 16 kilobytes of RAM memory.

SPECIFICATIONS

Print Method:	Serial impact dot matrix
Print Speed:	180 cps 45 cps (NLQ)
Print Head:	9-pin
Print Direction:	Bi-directional, logic-seeking
Character Set:	ASCII 96 characters International 43 characters (16 versions) IBM PC® 64 characters + Scandinavian characters
Fonts:	Normal (Pica size), Prestige (NLQ) Each with Elite, Enlarged, Condensed, Emphasized, Double-Strike, Italic, Underline, Super/Subscript available. Options: Gothic, Anelia PS, Quadro
Bit Image Printing:	8 × 816, 978, 1,088, 1,224, 1,632, 1,956, 3,264 dots/line
Columns:	136 (Pica size), 162 (Elite size) 68 (Enlarged Pica), 81 (Enlarged Elite), 232 (Condensed), 116 (Enlarged Condensed)
Line Spacing:	1/6, 1/8, 7/72, n/72, n/216 inch
Paper Feed:	Friction feed Pin feed tractor Cut sheet feeder (optional)
Paper Width:	Cut sheet: 127—420 mm (5"—16.5") Fanfold: 127—406.4 mm (5"—16")
Copies:	1 original plus 2 copies
Interfaces:	Centronics parallel RS-232C serial
Dimensions (W × H × D): (inc. PF-40 pin tractor)	490 × 76 × 304.5 mm (19.3" × 3" × 12")
Weight: (inc. PF-40 pin tractor)	6.7 kg (14.7 lbs.)

Specifications are subject to change without notice.
IBM PC is a registered trademark of International Business Machines Corp.
FX-100+ is a registered trademark of EPSON CORPORATION.

brother®
We put our reputation on paper.

Jim Thylin

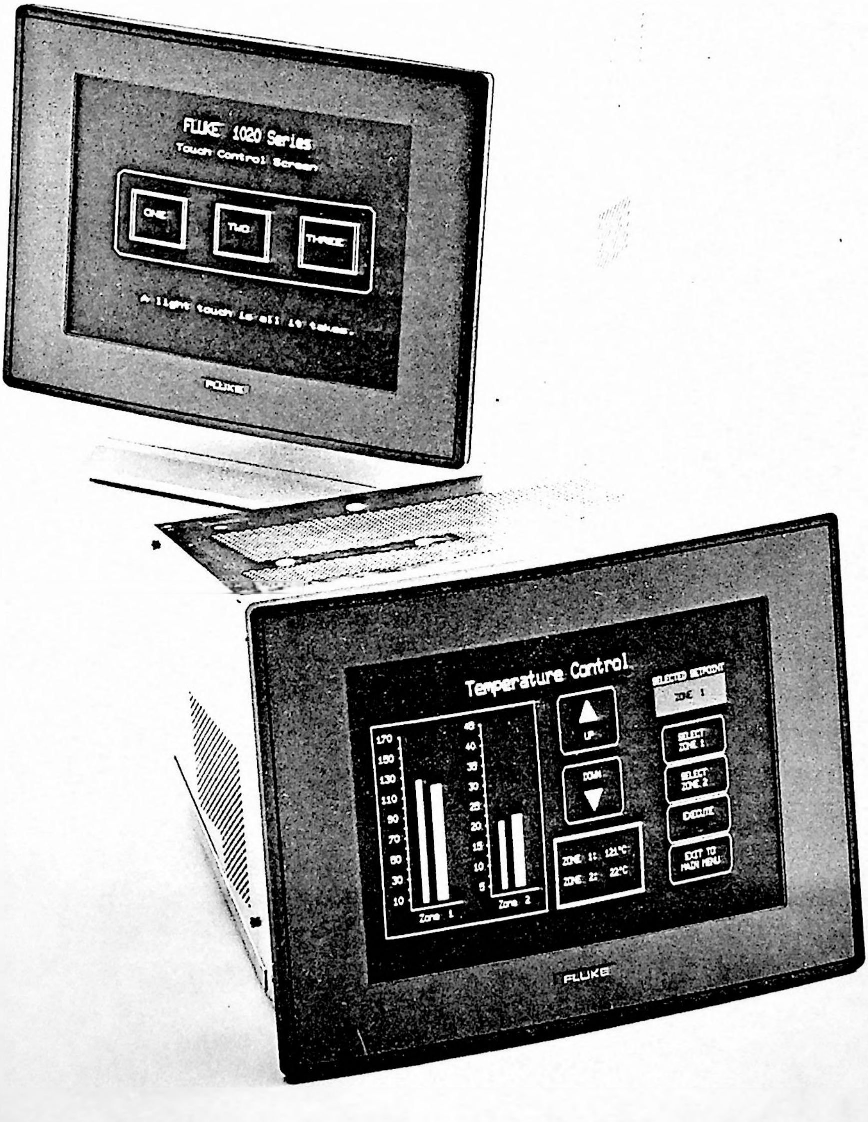
DISTRIBUTED TECHNOLOGY, INC. (DTI)
1201 SAN LUIS OBISPO AVE.
HAYWARD, CA 94544
(415) 487-8300

Printed in Japan

FLUKE®

1020 Series

Touch Control Screen



Introduction

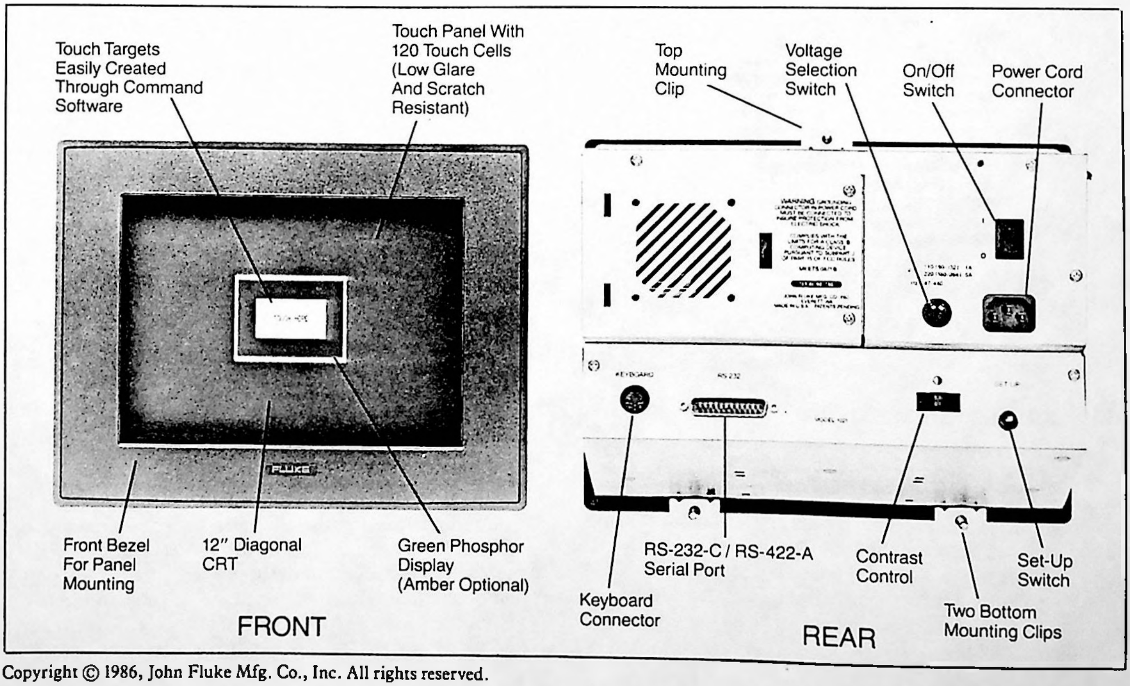
The 1020 Series Touch Control Screen (TCS) is a sophisticated interface between human operators and computer-driven systems. The TCS allows the operator, with a minimum of training, to accurately and efficiently control even the most complex operations, simply by touching the screen.

The Touch Control Screen interprets character strings from the host computer to create displays that provide the operator with information, or ask the operator to select from the choices presented. The operator responds naturally by pointing to the appropriate choice and touching the display lightly. The TCS, acting as a software configured control panel, tells the computer that a particular area was touched.

The 1020 Series is easy to integrate into any computer-based system, and is packaged to be easily mounted into a panel, a rack, or on a table top. High-level software support facilitates the design of "touch targets" using only a few simple commands. The extreme reliability and durability of the TCS makes it appropriate for use in a variety of rugged environments.

The Touch Control Screen has many useful features, including:

- Touch Panel consisting of a matrix of 120 touch cells (12 rows, 10 columns).
- High contrast, green phosphor 12" CRT capable of displaying 1920 characters (24 rows, 80 columns).
- RS-232-C / RS-422-A serial interface with selectable baud rates in standard increments up to 19,200.
- Advanced programming commands to reduce programming time and increase communication throughput.
- Selectable character attributes. Highlight, blinking, underline, and reverse video.
- Non-volatile memory to store set-up parameters while power is off.
- Easily mounts in a panel, instrument rack, or on a desk top.
- Optional NEMA 12 water and dust resistance when panel mounted.
- Accessories include a keyboard, rack mounting kits, and serial interface cables.
- Options include amber display, internal fan, desk-top enclosure, and tilt-swivel base.

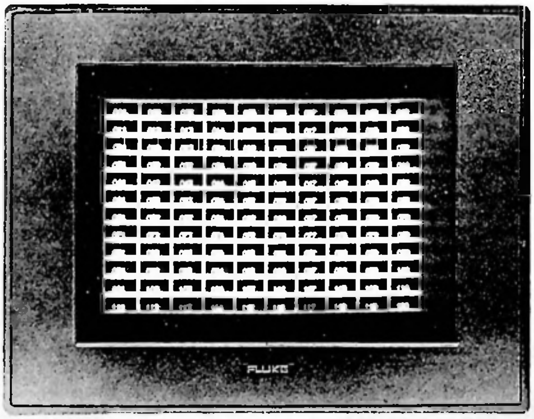

Touch Targets Easily Created Through Command Software
Touch Panel With 120 Touch Cells (Low Glare And Scratch Resistant)
Front Bezel For Panel Mounting
12" Diagonal CRT
Green Phosphor Display (Amber Optional)
FRONT
Top Mounting Clip
Voltage Selection Switch
On/Off Switch
Power Cord Connector
RS-232-C / RS-422-A Serial Port
Contrast Control
Set-Up Switch
Keyboard Connector
Two Bottom Mounting Clips
REAR

The Touch Panel

The Touch Panel consists of transparent, conductive polyester sheets that are affixed to the front of the display over a rigid acrylic backplane, forming a transparent switch matrix. The switch matrix provides 120 separate switch locations that can be labeled with words or symbols on the underlying display. Individual switches are called touch cells, and are arranged in a matrix of 12 rows by 10 columns. On the TCS display, each touch cell overlies an area two lines high by eight characters wide. Each touch cell is identified by a number that is reported to the host computer when the touch cell is pressed.

For increased resistance to chemical attack and abrasion, the Touch Panel is coated with a hardened anti-glare surface that adapts it to a variety of rugged environments. The durable Touch Panel has been successfully tested for over 1 million touch operations.

The 1020 Series display provides precise geometry and linearity, and is designed to remain in proper alignment with the Touch Panel. If adjustments are required, an alignment routine is provided to easily perform display-to-Touch Panel alignment.

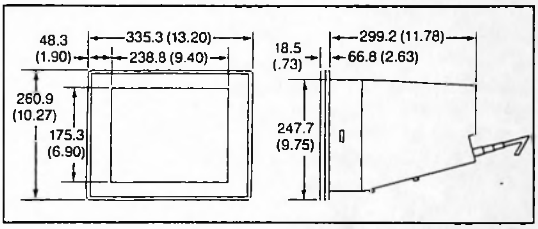


Mechanical Integration

The Touch Control Screen is designed to be easily mounted into a simple cutout in a metal, fiberglass, or wood panel and is supplied with complete hardware for panel mounting. To properly mount the 1020 Series, a hole size of 308 mm (12.13 in) high by 232 mm (9.13 in) wide should be cut in the mounting material. 299 mm (11.78 in) should be allowed for the depth of the TCS. An additional 82 mm (3.22 in) should be allowed for the power cord and a standard RS-232-C cable, for a total depth of 381 mm (15 in).

Mounting is as simple as cutting a hole, inserting the Touch Control Screen, tightening three clips, and attaching the cables. For applications that require a drip-proof and dust-tight seal, a NEMA 12 Panel Mounting Kit is available from Fluke. For easy installation in a standard 19-inch instrument rack, a Rack Mount Kit is available.

The 1020 Series is available with an optional sturdy plastic enclosure for applications that require desk-top or stand-alone mountings. An optional tilt-swivel base provides a forward tilt of up to 5 degrees and a backward tilt of up to 15 degrees.



Electrical Integration

Power

The Touch Control Screen is powered from standard AC line voltage. The voltage setting can be switch-selected between 110 VAC and 220 VAC with any line frequency between 47 and 440 Hz. The TCS consumes less than 50 watts of power.

Communications Interface

The Touch Control Screen has an RS-232-C serial interface that allows it to be connected to a host computer directly, or through a modem. The TCS supports full-duplex communication, providing two-way serial data exchange for simultaneous transmit and receive by both the Touch Control Screen and the host computer. The 1020 Series also supports RS-422-A for greater noise immunity when communicating over long distances.

Communication between the host computer and the TCS conforms to the ANSI standard (ANSI X3.64-1979) supported by most common computer terminals. All communication between the host and the 1020 Series is accomplished using standard ASCII characters.

Communication parameters are easily set by pressing the SET UP switch on the rear of the Touch Control Screen. A menu allows selection of the appropriate baud rate, parity and other communication parameters. The parameter values can be stored in the TCS in non-volatile memory and recalled on the next power-up or reset.

The Custom Connector

Internal logic board signals may be accessed externally by use of a 26-pin Custom Connector. Available signals include composite video for an external monitor, off-board contrast control, reset, external SET UP switch, beeper, and +5V, +12V, −12V and Ground. There is also an alternate set of pins for custom keyboard applications. An optional Custom Connector Extender Cable is available to provide access to these signals from the rear of the Touch Control Screen.

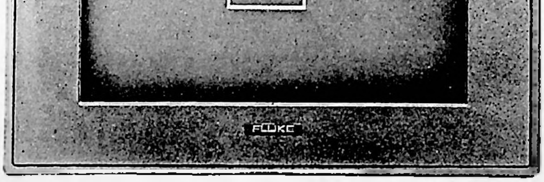
Software Integration

Enhanced command software allows the programmer to create touch targets with just a few simple commands. These commands reduce programming time and increase communication throughput. The firmware resident in the 1020 Series provides unique features that allow for complete flexibility and programming ease.

Region Commands

Region commands allow the programmer to create touch targets by drawing rectangular boxes on the display. The programmer uses a single command to define the size of a rectangle and select one of four box types. The current cursor position determines the location of the upper left corner of the rectangle. Other Region commands allow the host computer to erase a region and alter the attributes in a region.

An example Region Command from the host would be: "⟨ESC⟩ [6; 10; 20; 4t" which would create the following display:



Touch Panel Commands

These commands determine how the Touch Panel responds when a touch cell is pressed. Some commands allow the host computer to set an automatic repeat rate for touches and generate a beep to acknowledge when a touch has been made. Other Touch Panel commands allow the host to place the Touch Control Screen in a polled mode, where touches will be queued and reported to the host only on request. The default touch report from the TCS is: "⟨ESC⟩ [>2;⟨Pn⟩n" where "⟨Pn⟩" is a 3-digit ASCII numeric string representing the number of the touch cell. In addition, the introducers can be changed and/or a terminator can be appended to the touch report string.

Attributes

Character Attributes allow the programmer to alter the visual appearance of characters on the display. Characters can be displayed in reverse video, highlighted, blinking, underlined, or any combination of these.

Line Attributes can be used to provide double-size or double-width lines in selected areas of the display.

The Scrolling Region Command allows the programmer to specify a section of the screen that will scroll. Areas of the display outside the Scrolling Region will remain stationary.

Lockout

To provide for complete host control of TCS operations, the 1020 Series can be programmed to lock the user out of the keyboard, Touch Panel, and Set Up mode.

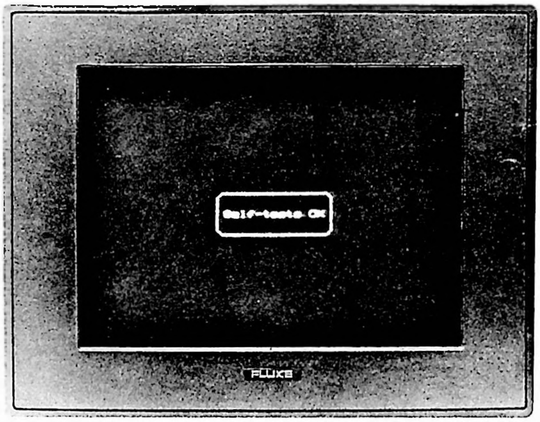
Other Commands

Many other commands are provided to simplify programming. Some of these commands move the cursor, read back the attributes or characters displayed under the cursor, obtain status information from the Touch Control Screen, and affect the appearance of the screen.

International Character Sets

The 1020 Series supports National Replacement Code characters, allowing the display of characters used in many countries. When a national character set is selected, characters unique to that language can be easily displayed by the host program. Character sets supported by this are: Danish, Dutch, French, Canadian, Finnish, German, Italian, Spanish, Swedish, Swiss, and English (UK).

Diagnostics



The 1020 Series' extensive diagnostic and self-test routines ensure the reliability of TCS operations. Whenever the Touch Control Screen is powered-up or reset, an automatic self-test of the ROM, RAM, non-volatile memory and Touch Panel ensures the integrity of TCS components. The tests used to automatically verify TCS operation can also be initiated by the host computer.

Six user-selected tests are available from the test screen to test touch panel integrity, display geometry and intensity, the standard character and extended character sets, and the keyboard. Additional tests accessible from the test screen, the Raw Keyboard Test and the Continuous Integrity Test, are provided to assist service personnel in maintaining the TCS.

Service

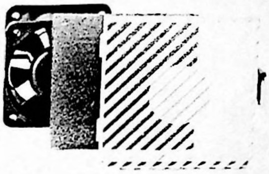
The high quality of the 1020 Series Touch Control Screen ensures extended use in rugged environments. The modular design of the TCS simplifies periodic maintenance and necessary servicing. The 1020 Series is backed by a one year warranty against defects in materials and workmanship. This warranty applies throughout the world at any of sixty worldwide Fluke Technical Service Centers. The Fluke service network also offers extended warranties and customer support services.

Options

Amber Phosphor Display 1021-10

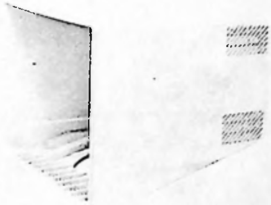
The amber monochrome P134 phosphor display is available as a factory configured option. A green P31 phosphor display is standard.

Fan 1021-20



An optional fan, including filter and filter cover, is available to increase the operating temperature range of the 1020 Series. The fan is located inside the enclosure and is not visible from the exterior. The fan will increase the reliable operating temperature range of the Touch Control Screen by 10°C. (See Specifications.)

Enclosure 1021-30



An attractive plastic enclosure is available for operating the Touch Control Screen on a desk-top. The enclosure is standard with the Model 1020 Touch Control Screen Package.

Tilt/Swivel Base 1021-31

The Tilt/Swivel Base can be attached to the Touch Control Screen for easy up-down and left-right movement of the unit. The Tilt/Swivel Base is designed to be used with the 1021-30.

Custom Connector Extender Cable 1021-60

This cable extends the Custom Connector signals to a connector at the rear of the TCS. A male 37-pin D-type connector can be used to interface with this option.

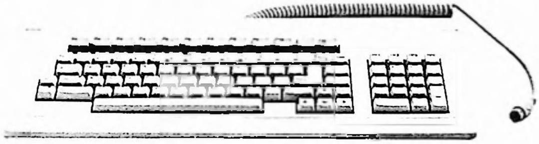
1020 Series Manual Set 1021-90

Manuals for the 1020 Series include the Inspection and Installation Guide, User's Guide, Programmer's Guide, and Designer's Guide. The manual set is standard with the Model 1020 Touch Control Screen Package.

1020 Series Service Manual pn 778803

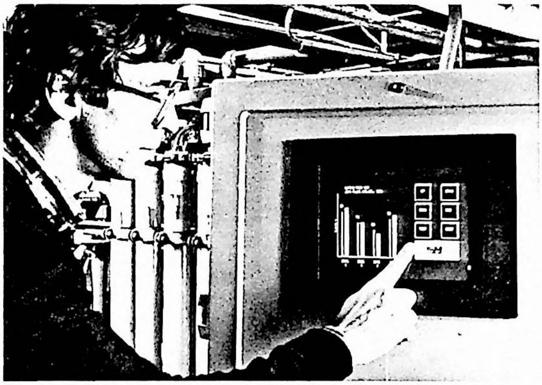
This manual is available through Customer Support Services. It covers all the information necessary to perform maintenance and service of the Touch Control Screen. Complete set of schematics and test procedures are included.

Accessories



Alphanumeric Keyboard Y1000

The Alphanumeric Keyboard is provided for programming or data entry applications with the 1020 Series. The keyboard also has a numeric keypad and 10 function keys for custom programming support.



NEMA 12 Panel Mounting Kit Y1070

This kit provides the gasketing and mounting hardware necessary to give the Touch Control Screen a NEMA Type 12 rating (drip-proof, dust-tight) when panel mounted in a suitable enclosure. The Type 12 standard is published by the National Electrical Manufacturers Association.

19″ Rack Mount Kit Y1080

The Y1080 is the front panel to use when mounting the Touch Control Screen into a 482.6 mm (19 in) wide instrument rack. The panel is 6 rack units high (265.9 mm, 10.5 in). Hardware necessary for mounting the Touch Control Screen into the Rack Mount Kit is supplied with the 1020 Series.

19″ Rack Mount Kit with 18″ Slides Y1081

The Y1081 is used for complete slide-mounting of the Touch Control Screen in a 482.6 mm (19 in) instrument rack. 457 mm (18 in) rack slides, mounting hardware, and the Rack Mount Kit (Y1080) are all included in this accessory.

19″ Rack Mount Kit with 24″ Slides Y1082

The Y1082 is identical to the Y1081 except that is provides 610 mm (24 in) rack slides instead of 457 mm (18 in) rack slides.

Keyboard Extender Cable Y1085

The Y1085 is used to extend the keyboard interface to the front panel of the Y1080, Y1081, and Y1082 rack mounting accessories. The Keyboard Extender Cable can also be used in most custom panel mounting applications.

Contrast Enhancement Overlay Y1090

The Y1090 is a gray polyester overlay that enhances visual contrast between the display background and characters on the Touch Control Screen. The Y1090 has a scratch-resistant, anti-glare surface.

Serial Communication Cables

Standard RS-232-C and Null Modem cables are available in various lengths. The Standard RS-232-C cables may be ordered in lengths of 2 and 10 meters. The Null Modem cable may be ordered in lengths of 0.3, 2, and 4 meters.

Specifications for the 1020 Series Touch Control Screen

Environmental

Temperature:
Operating: 0° to 50°C without plastic enclosure.
0° to 60°C without plastic enclosure, with optional fan.
0° to 40°C with plastic enclosure.
0° to 50°C with plastic enclosure, with optional fan.
Non Operating: −40° to 75°C.

Relative Humidity:
0° to 30°C: 95% (Noncondensing).
30° to 40°C: 75% (Noncondensing).
40° to 55°C: 45% (Noncondensing).
55° to 75°C: 40% (Noncondensing).

Altitude:
Operating: 3,050 meters (10,000 ft).
Non Operating: 12,000 meters (40,000 ft).

Power:
Voltage: Switch selectable: 90 to 132V ac, 47 to 440 Hz; 180 to 264 V ac, 47 to 440 Hz.
Safety Ground Leakage Current: Less than 0.5 mA at 47 to 63 Hz. Less than 3.5 mA at 63 to 440 Hz.
Power Consumption: Less than 50 watts.

Vibration: Meets the requirements of MIL-T-28800C, class 3.

Frequency	Force	Double Amplitude
5–15 Hz	0.7G at 15 Hz	1.52 mm (0.06 inch)
15–25 Hz	1.3G at 25 Hz	1.02 mm (0.04 inch)
25–55 Hz	3G at 55 Hz	0.51 mm (0.02 inch)

Shock: ½ sine shock of 30G for 11 ms ±1 ms.

Dust-tight and Drip-tight: Suitable for use in NEMA Type 12 enclosure with optional NEMA 12 Panel Mounting Kit.

Packaging

Dimensions: 335.3 mm (13.2 in) wide; 260.9 mm (10.27 in) high; 318 mm (12.5 in) deep without enclosure; 330 mm (13.0 in) deep with enclosure.

Minimum Mounting Depth: 300 mm (11.8 in) from back of bezel to back of TCS, without cables. 381 mm (15.0 in) with Fluke cables.

Weight: 8.44 kg (18.6 lbs) without plastic enclosure.
Enclosure: 1.59 kg (3.5 lbs).
Tilt/Swivel Base: 0.68 kg (1.5 lbs).

Panel Mount: Three point mounting clips provided.

Rack Mount (optional): 482.6 mm (19.0 in) wide. 265.9 mm (10.5 in) high, six rack units.

Tilt/Swivel Base (optional): Forward tilt of up to 5 degrees. Backward tilt of up to 15 degrees.

Touch Panel

Type: Membrane switch matrix.

Number of Touch Cells: 12 rows, 10 columns, extending to the edge of the display.

Touch Cell Size: 2 rows, 8 columns of single size characters.

Resistance to Damage: Hardened scratch resistant surface.

Optical Characteristics: Antiglare standard.

CRT

Tube Size: 305 mm (12 in), measured diagonally.

Tube Type: Monochrome P31 Phosphor (green) standard. Monochrome P134 phosphor (amber) optional.

Display Characteristics

Scanning Method: Non-interlaced raster.

Refresh: 50/60 Hz, programmable, stored in non-volatile memory.

Active Display Size: 203 mm (8 in) wide, 137 mm (5.4 in) high.

Character Cells on Screen: 24 lines of 80 characters.

Pixels per Character Cell: 8 dots wide, 10 dots high.

Line Attributes: Normal, double width, double size.

Character Attributes: Normal, highlight, underline, blink, reverse video.

Character Set Size: 256 characters (includes ASCII characters).

International Characters: Provided using National Replacement Code (NRC) method.

Display Alignment: Programmable; stored in non-volatile memory. Horizontal adjustment range: 12 characters, 1 character resolution. Vertical adjustment range: 2.4 characters (24 scan lines), 0.1 character (1 scan line) resolution.

Beeper

Frequency: 2.1 kHz.

Loudness: Two levels, jumper selectable.

Keyboard (Optional)

Length: 470.2 mm (18.51 in).

Width: 211.5 mm (8.33 in).

Maximum Total Height: 50.8 mm (2.0 in), includes keycaps.

Weight: 1.59 kg (3.5 lb).

Standards

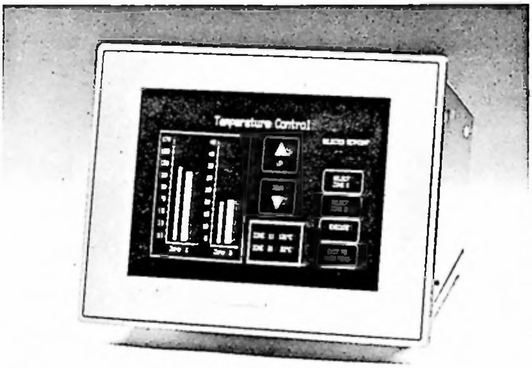
The Touch Control Screen is designed to comply with the following safety standards:

ANSI/UL	478	Standard for Electronic Data Processing Units and Systems.
IEC	348	Safety Requirement for Electronic Measuring Apparatus.
IEC	435	Safety of Data Processing Equipment.
CSA	556B	Certification Requirements for Electronic Instruments and Scientific Apparatus.

The TCS has been verified to comply with the following EMI (electromagnetic interference) standards:

FCC	Part 15, subpart J	Computing Devices (class B).
VDE	0871	Radio Frequency Interference Suppression of Radio Frequency Equipment for Industrial, Scientific and Medical and Similar Purposes (class B).

Ordering Information



Model

1021	Touch Control Screen

Options

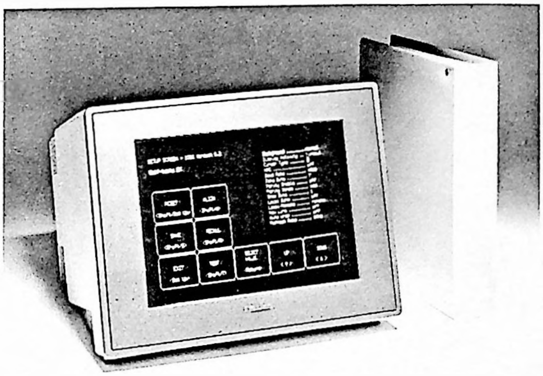
1021-10	Amber Phosphor Display (factory configured)
1021-20	Fan (with cover, filter)
1021-30	Enclosure
1021-31	Tilt/Swivel Base (used on 1021-30)
1021-60	Custom Connector Extender Cable (factory configured)
1021-90	1020 Series Manual Set

Accessories

Y1000	Alphanumeric Keyboard
Y1070	NEMA 12 Panel Mounting Kit
Y1080	19-inch Rack Mount Kit
Y1081	19-inch Rack Mount Kit with 18-inch slides
Y1082	19-inch Rack Mount Kit with 24-inch slides
Y1085	Keyboard Extender Cable (used on Y1080, Y1081, Y1082)
Y1090	Contrast Enhancement Overlay
Y1702	2m RS-232-C Null Modem Cable
Y1703	4m RS-232-C Null Modem Cable
Y1705	0.3m RS-232-C Null Modem Cable
Y1707	2m RS-232-C Cable
Y1708	10m RS-232-C Cable

Service/Warranty

SC1 1020	One-year Extended Warranty for 1020
SC1 1021	One-year Extended Warranty for 1021
PN 778803	1020 Series Service Manual



Model

1020	Touch Control Screen Package Includes: Touch Control Screen Enclosure Manual Set

Options

1020-10	Amber Phosphor Display (factory configured)
1020-31	Tilt/Swivel Base

John Fluke Mfg. Co., Inc.
P.O. Box C9090, Everett, WA 98206
Tel. (206) 347-6100
For more information — or where to buy Fluke products call:
(800) 426-0361 (toll free) in most of U.S.A.
(206) 356-5400 from AK, WA
(206) 356-5500 from other countries

Fluke (Holland) B.V.
P.O. Box 2269, 5600 CG Eindhoven, The Netherlands
(040) 458045, TLX: 51846
Phone or write for the name of your local Fluke Representative.

Printed in U.S.A.

A0245A-02U8604/SE EN