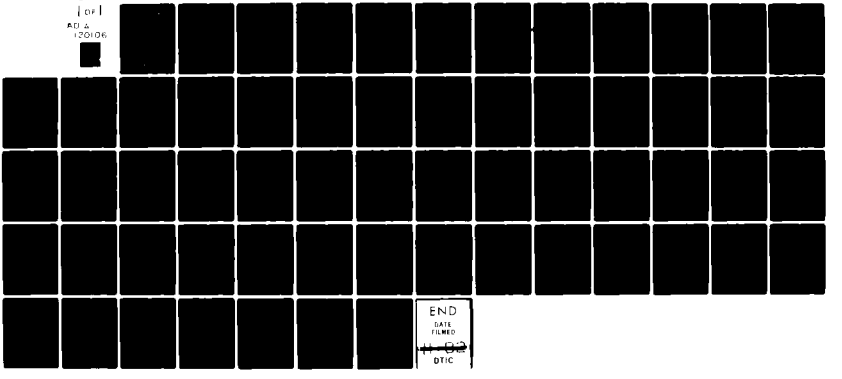
AD-A120 106 SYSTEMS CONTROL TECHNOLOGY INC WEST PALM BEACH FL F/G 17/7
3D LORAN-C NAVIGATOR DOCUMENTATION.(U)
JAN 82 E H BOLZ, L D KING DTFA01-80-C-10080
UNCLASSIFIED DOT/FAA/RD-82/16 NL



DOT/FAA/RD-82/16

Systems Research &
Development Service
Washington, D.C. 20591

3D Loran-C Navigator Documentation

Eric H. Bolz
Larry D. King

January 1982

Final Report

U.S. Department of Transportation
Federal Aviation Administration

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT/FAA/RD-82/16	AD-A120106	

4. Title and Subtitle	5. Report Date
3D Loran-C Navigator Documentation	January 1982
	6. Performing Organization Code
	8. Performing Organization Report No.
7. Author(s)	
Eric H. Bolz, Larry D. King	
9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Systems Control Technology, Inc. 2326 S. Congress Ave., Suite 2A West Palm Beach, Florida 33406	11. Contract or Grant No. DTFA01-80-C-10080
	13. Type of Report and Period Covered Final Report May 1981 - January 1982
12. Sponsoring Agency Name and Address	14. Sponsoring Agency Code
Department of Transportation Federal Aviation Administration Systems Research and Development Service Washington, D.C. 20590	ARD-320

15. Supplementary Notes

16. Abstract

The purpose of this task was to develop a 3D Loran-C Navigator by configuring an interface unit between an airborne Loran-C navigator (Teledyne TDL-711) and an Altitude Alerter/VNAV Guidance system (Intercontinental Dynamics model 541). The digital computer-based interface unit was designed to allow the flight crew to specify the approach slope (3.0 to 9.9 degrees). This report documents the hardware and software in the interface unit, and interconnection with the other involved systems.

The availability of accurate, three-dimensional approach guidance information at airports where no ILS is available provides significant operational advantages, to helicopter operators in particular. The 3D Loran-C navigator system was bench tested and flight demonstrated. Smooth, accurate (within the limitations of Loran-C) descent guidance information was obtained.

17. Key Words	18. Distribution Statement
INSTRUMENT APPROACH, LORAN-C, PRECISION APPROACH, 3D NAVIGATION, VNAV, MICROCOMPUTER APPLICATIONS	This document is available to the U.S. public through the National Technical Information Service, Springfield, Virginia 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	60	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

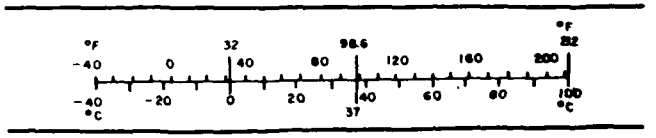


TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 3D LORAN-C NAVIGATOR FUNCTIONS AND SOFTWARE

1.1 INTRODUCTION

The concept which motivated the development of this device was to be able to conduct instrument approach procedures with let-down guidance at airports or landing sites which have no ILS glideslope equipment. This capability requires the availability of an area coverage navigation sensor with good accuracy and stability, such as Loran-C, and other instrumentation required to process altitude and position data in order to calculate command descent guidance. At present there is no commercially available Loran-C navigator which provides 3D (VNAV) guidance. In order to demonstrate the 3D Loran-C capability, hardware was developed to interface a popular Loran-C navigator (the Teledyne TDL-711) to a commercially-available approach guidance computer and altimeter (the model 541 VNAV/ALERTER made by Intercontinental Dynamics). This hardware is discussed in detail in Section 2.

1.2 SYSTEM INTERFACE REQUIREMENTS

The major problem addressed by this system was to resolve the interface incompatibility between the TDL-711 navigator and IDC VNAV system. The normal input provided to the IDC VNAV in an ordinary installation, where it is connected to a DME or an RNAV computer, is a D.C. voltage analog of distance to station/or waypoint. The TDL-711 is designed to drive a standard localizer needle, localizer flag and TO/FROM flag, and has no outputs for other aircraft instrumentation, in particular, no output intended to drive a distance to waypoint (DTW) display. The TDL-711 does have two digital data ports. The first is dedicated to communications with the associated control/display unit (CDU). The second is intended for driving a remote display unit (RDU), which repeats the data displayed on the CDU. The data content of the RDU data stream is, however, modifiable through parameters stored on an EPROM. This fact has been used to advantage in flight test programs conducted by SCT, FAATC and DOT/TSC. A version of the EPROM was developed several years ago for NASA LRC which has been used in most of these tests. It causes the RDU data port to generate a long (157 bytes) stream of data which is then recorded by the

airborne data acquisition system. One of the elements in that data stream is distance to waypoint (DTW) in BCD format in units of nautical miles and tenths. The task of the interface unit provided under this contract was to accept the RDU data format which is periodically transmitted, extract the DTW data (and other related information), and generate a smooth, accurate D.C. analog representation of aircraft distance to waypoint to drive the VNAV computer. A further function was to provide approach guidance at a selectable glideslope angle (up to 9.9°), rather than just the fixed 3° built into the VNAV.

1.3 LORAN-C OUTPUT DATA CHARACTERISTICS

Since the TDL-711 was not conceived with the intention of driving a remote DTW indicator (let alone a VNAV descent guidance computer), the output data which is available requires extensive processing to result in suitable guidance data. The form of the output data is binary coded decimal in units of one-tenth nautical mile. This resolution is not sufficiently fine to directly control descent guidance. At a 6° approach angle, a 0.1 nm resolution yields a 64 foot vertical guidance resolution. The scale factor of the IDC VNAV is 20 ft per dot deflection, so the discontinuity would be in excess of 3 dots. The nominal data update rate of the TDL-711 is roughly once per second. However, it has been found in earlier flight evaluations that this rate can be variable, and interruptions to the data stream of several seconds can result. Additional parameters in the output data stream can be used to determine when waypoints have been sequenced, or if the navigator is unable to track the Loran-C signal. These factors should control the presence of the glideslope flag.

It was apparent from the outset that some form of digital filtering scheme should be applied within the interface unit in order to track and smooth the data. The use of True Airspeed data as an aid to the filtering process was considered. However, Loran-C receivers typically have no provision for airspeed input (it is not necessary for tracking the time difference signals, even at high velocities). Furthermore, many of the types of aircraft which would find a Loran-C 3D approach system useful (in particular helicopters) do not have an electrical

airspeed signal on board. Therefore, the only data available is the Loran-C DTW information itself. In order to avoid tracker standoff errors, a second order alpha-beta tracker algorithm was implemented. This set of equations estimates velocity as well as position, and requires an initial guess of velocity to start tracking. This is provided through a set of jumper wires which may be changed as required for a given aircraft performance.

1.4 TRACKER ALGORITHM

The alpha-beta (α-β) tracker technique is tailored to tasks involving real-time tracking of data where knowledge of probable future behavior is limited, and where no redundant data sources are available. Its design is based on assumptions regarding (1) the accuracy of the data representing the physical entity being tracked (in this case the Loran-C DTW is the data and the physical entity is flight progress along the path), and (2) probable future behavior of the entity being tracked. The representation of accuracy is the standard deviation of the measurement error. The representation of probable future behavior is the standard deviation of the longitudinal acceleration expected of the aircraft as it conducts an instrument approach.

The philosophy behind the α-β tracker is discussed in further detail in Reference 5. The equations for computing the optimal gains (α and β values) are presented there. The gains are functions of three values, the standard deviation of measurement error, the standard deviation of acceleration, and the time between updates. Based on earlier flight test experience with Loran-C, it was assumed that the Loran-C random error component (repeatability) would be 300 ft (0.05 nm). In addition, the granularity of the data (0.1 nm) introduces an error whose S.D. = 0.03 nm. These values were RSS added to yield an overall measurement error estimate of 0.06 nm. Based on earlier experimental tracker development work[5], it was assumed that the standard deviation of longitudinal acceleration was 0.1 g. As previously mentioned, the time between updates is a variable. The

α and β gain equations were solved for several values of update interval, with results shown in Table 1.1.

Table 1.1 α and β Gains Versus Update Interval

Δt	1	3	5	10
α	.121	.321	.474	.719
β	.0078	.0206	.0302	.0442

Linear approximations of gain values versus update interval have been estimated, as follows:

$$\alpha = 0.05447 + 0.0664\Delta t$$

$$\beta = 0.0039 + 0.0039\Delta t$$

As implemented in the interface unit, the update interval is calculated as the difference in arrival times of two sucessive Loran-C data streams. Each time a Loran-C data block is received, the gains are calculated based on the update rate. The gains are used to estimate new distance and velocity values using the following equations:

$$\left.\begin{aligned} \hat{\hat{x}}(n+1) &= \hat{x}(n) + \Delta t\, \hat{\dot{x}}(n) \\ \hat{\dot{\hat{x}}}(n+1) &= \hat{\dot{x}}(n) \end{aligned}\right\} \text{ Prediction Equations}$$

$$\left.\begin{aligned} \hat{x}(n) &= \hat{\hat{x}}(n) + \alpha(D(n) - \hat{\hat{x}}(n)) \\ \hat{\dot{x}}(n) &= \hat{\dot{\hat{x}}}(n) + \beta(D(n) - \hat{\hat{x}}(n)) \end{aligned}\right\} \text{ Smoothing Equations}$$

Given, at any update n, a value for distance estimate $\hat{x}(n)$ and velocity estimate ($\hat{\dot{x}}(n)$), the predicted distance $\hat{\hat{x}}(n+1)$ and velocity $\hat{\dot{\hat{x}}}(n+1)$ at the next update n+1 are given by the prediction equations. Once predicted values ($\hat{\hat{x}}$, $\hat{\dot{\hat{x}}}$) are known for an update n, the measured distance D(n) is used to smooth and update the tracker yielding estimated

distance $\hat{x}(n)$ and velocity $\hat{\dot{x}}(n)$, using the smoothing equations and the calculated values for α and β. To start the process:

$$\hat{x}(1) = D(1)$$

$$\hat{\dot{x}}(1) = \text{Value programmed using jumpers}$$

This is done at any point that the software chooses to reset (discussed in the next section). In between updates, the distance to waypoint value sent to the D/A output device is calculated using the prediction equations. The value for Δt used is time since the last update. This value is updated sixteen times per second. A second D/A channel is used to generate the DME valid flag signal (+16 VDC is valid, 0 is invalid).

1.5 OPERATIONAL SOFTWARE

The software for the 3D Loran-C Navigator interface unit is written in Intel 8080 Assembly Language. The object code has been programmed into a 2716 EPROM located at board position U102. The Monolithic Systems Uniform Monitor EPROM is located at position U101. Upon power-up or system reset, the program in the Uniform Monitor executes, which transfers control to the interface program in U102. The first function of this program is to move the entire program into RAM (random access memory) at location 8040H (hexidecimal) where it executes. The Assembly Listing in Figure 1.1 shows the code as assembled to load at location 8040. Once loaded, no further use of the EPROM at U102 is made. Several subroutines in the monitor at U101 are used in many places in the interface program.

The interface unit program consists of the following sections:

1) Data Area Initialization 8040H
   - Initializes all data areas, addresses and pointers used by the program (this section is not executed).

2) Monitor Set-Ups 80A4H
   - Completes initializations normally performed by the Monitor (copied from the MSC Uniform Monitor listing)

```
                        ;PROGRAM LORANV -- DERIVE SMOOTHED DTW FROM TDL-711 RDU DATA
                        ;12/07/81

8040 =          BASE    EQU     8040H
9000 =          ARRAY   EQU     9000H           ;LORAN-C DATA BUFFER
                                                ;PROGRAM LOAD POINT
                                                ;ADDRESS OF LORAN DATA BUFFER AREA

8040                    ORG     BASE
8040 C3A480             JMP     START                                   ;JUMP TO EXECUTABLE CODE
8043 C33582             JMP     TIME            ;FOR TEST               ;) THESE MAY BE EXECUTED BY HITTING
8046 C34182             JMP     HDISP           ;FOR TEST               ;( NMI SWITCH, THEN TYPING GO 8043,
8049 CDF080             CALL    CSETUP                                  ;( ETC. WITH TERMINAL ATTACHED
804C C36581             JMP     PSETUP          ;(CSETUP&PSETUP) FOR TEST ;)

8060                    ORG     BASE+20H                                ;ORIGIN OF DATA AREAS

8060 161E       INTRVL  DW      1E16H           ; =1/256 OF SECOND      ;FORCES CLOCK TO 1/256TH SEC INCREMENTS
8062            CLKADR  DS      3               ;INTERRUPT-DRIVEN CLOCK ;3 BYTE TIME FIELD (LSB FIRST)
8065            ARRTIM  DS      9               ;SPACE FOR ARRIVAL TIME, LAST ARRIVAL   ;DATA AREA
                                                ;TIME, AND DIFFERENCE
806E 00         T16     DB      00H             ;1/16TH SECOND ELAPSED TIME FLAG 00H=NO ;SET ON EVEN 1/16TH SEC MARKS
806F 1000       PT10H   DW      0010H           ;FOR DIVIDING BY 16
8071 01         SOSEQ   DB      01              ;'START OF SEQUENCE'    ;SET TO 1 INDICATING START OF LORAN DATA
8072 00         POINTR  DB      00              ;'CURRENT BYTE POINTER' ;POINTS TO BYTE IN LORAN BUFFER
8073 00         DDONE   DB      00              ;'DATA LOAD DONE'       ;SET TO 1 INDICATING FULL LORAN BUFFER
8074 9D         PLIMIT  DB      09DH            ;=157, "BYTE LIMIT" OF LORAN DATA       ;BYTES IN DATA STREAM
8075 00         LASTWP  DB      00H             ;PREVIOUS WP PAIR       ;)
8076 00         WP      DB      00H             ;PRESENT WP PAIR        ;( FROM DATA STREAM
8077 00         DECIML  DB      00H             ;LORAN DECIMAL POINT 08H=VALID  ;)
8078 0000       DTW     DW      0000H           ;LORAN DTW WORD
807A 00         VALID   DB      00H             ;VALID LORAN DATA STREAM 01H=VALID      ;RESULT OF DATA CHECK
807B 00         FLAG    DB      00H             ;OUTPUT DME VALID FLAG  00H=LOW, FFH=HI ;DRIVES D/A CONVERTER FLAG VOLTAGE
807C 0000       DTWC    DW      0000H           ;OUTPUT DME DISTANCE    ;CALCULATED DTW (MODIFIED BY APPR. ANGLE)
807E 00         NEWLEG  DB      00H             ;LEG HAS CHANGED  00H=NO        ;BASED ON CHANGED WP PAIR, ETC.
807F 0000       DELT    DW      0000H           ;TIME SINCE LAST LORAN UPDATE   ;CALCULATED FROM ARRTIM & CLKADR
8081 00         PROGSP  DB      00H             ;TO BE READ FROM JUMPERS        ;SPEED = 20X JUMPER SETTING
8082 00         ANGLE   DB      00H             ;INPUT DIGI-SWITCH ANGLE        ;BCD DEGREES (UNITS & TENTHS)
8083 0000       ANGLEB  DW      0000H           ;BINARY VALUE OF ANGLE          ;16 BIT FIXED POINT BINARY (HH.HH)
8085 AF00       ANGLE3  DW      00AFH           ;(1/3DEG)X16(SCALE)X((.040/4.9988)X4096)        ;SCALING & CONVERSION FACTOR
                                                        ;/(256=1MILE AS 0100H)
8087 0000       LASDTW  DW      0000H           ;SAVE DTW VALUE         ;) DTW AND SPEED VALUES
8089 0000       LASSPD  DW      0000H           ;SAVE SPEED VALUE       ;( FROM PREVIOUS UPDATE
808B 0000       BCD16   DW      0000H           ;ARGUMENT TO BCDCV      ;( ARGUMENTS FOR BCDCV
808D 0000       OUT16   DW      0000H           ;RETURNED SUBROUTINE ARGUMENT   ;) SUBROUTINE CALL
808F 0010       TLIM    DW      1000H           ;16-SECOND TIMEOUT PEROUT       ;RESET AFTER 16 SEC DATA LAPSE
8091 0000       DRIVE   DW      0000H           ;(D(N)-X(N))            ;USED IN TRACKER ESTIMATING EQUATIONS
8093 0000       NXTDTW  DW      0000H           ;PREDICTED DTW          ;) OUTPUT OF DTW AND
8095 0000       NXTSPD  DW      0000H           ;PREDICTED SPEED        ;( SPEED ESTIMATION
8097 0000       A;6     DW      0000H           ;WORKING REGISTERS
```

Figure 1.1 3D Loran-C Navigator Assembly Listing

```
8099 0000      B16     DW 0000H                ;
809B 0E00      ALPHA0  DW 000EH                ;ALPHA GAIN OFFSET                  ;
809D 1100      ALPHA1  DW 0011H                ;ALPHA GAIN TIME SENSITIVITY        ; } ALPHA-BETA
809F 0001      BETA0   DW 0100H                ;BETA GAIN OFFSET X256              ; } TRACKER GAINS
80A1 0001      BETA1   DW 0100H                ;BETA GAIN TIME SENSITIVITY X256    ;
80A3 00        SIGN    DB 00H                  ;SIGN SAVER FOR MULTIPLY (M1616)    ;USED IN SUBROUTINE M1616

80A4 21B400    START   LXI     H,00B4H         ;MON1: PRESET "GO" ADDR             ;COMPUTER HARDWARE AND SOFTWARE
80A7 2227FF            SHLD    0FF27H          ;STAD:                              ;INITIALIZATIONS, COPIED FROM
80AA CD1881            CALL    BAUDA           ;INITIALIZE MAIN PORT               ;MSC UNIVERSAL MONITOR PROGRAM
80AD CD3106            CALL    0631H           ;CINT: INIT I/O
80B0 2100FF            LXI     H,0FF00H        ;RMORG:
80B3 2231FF            SHLD    0FF31H          ;RSP: PRESET USER STACK
80B6 CD0607            CALL    0706H           ;MSGI:
80B9 0D0A4C4F52        DB      0DH,0AH,'LORAN-V',00H

                       ; INIT RST VECS TO POINT AT BKPT

80C3 21E0FF            LXI     H,0FFE0H        ;VECS:
80C6 11A701            LXI     D,01A7H         ;BKPT:
80C9 0EC3              MVI     C,0C3H          ;JMPI:
80CB 0608              MVI     B,08H
80CD 71        MONP    MOV     M,C             ;STORE JUMP OP CODE
80CE 23                INX     H
80CF 73                MOV     M,E             ;STORE BKPT PTR
80D0 23                INX     H
80D1 72                MOV     M,D
80D2 23                INX     H
80D3 05                DCR     B               ;DJNZ:
80D4 C2CD80            JNZ     MONP            ;LOOP TILL VECS ALL FIXED

80D7 31E0FF            LXI     SP,0FFE0H       ;MONSR: PRESET STACK
80DA AF                XRA     A
80DB 3200FF            STA     0FF00H          ;EF:

                       ;MONITOR SETUPS COMPLETE                                    ;END OF SECTION COPIED FROM MONITOR

80DE CD5281            CALL BAUDB              ;SET PORT B BAUD RATE               ;SOFTWARE INITIALIZATIONS
80E1 CD3981            CALL PARSET             ;SETUP PARALLEL PORTS & READ SPEED
80E4 CD6581            CALL PSETUP             ;PATCH INTERRUPT ADDRESSES
80E7 CD9281            CALL CRESET             ;RESET CLOCK
80EA CDF080            CALL CSETUP             ;SETUP CLOCK INTERRUPT, START CLOCK
80ED C36482            JMP  MAIN               ;START DTW TRACKING                 ;JUMP TO MAIN ROUTINE
                                                                                   ;INITIALIZATION ROUTINES FOLLOW

80F0 3E01      CSETUP  MVI A,01H                                                   ;SET UP SYSTEM REAL TIME CLOCK
80F2 327180            STA SOSEQ                                                   ;USES INTERRUPT 7. INTERRUPT IS
80F5 AF                XRA A                                                       ;GENERATED EVERY1/256TH SECOND
80F6 327280            STA POINTR
```

Figure 1.1 - Continued

```
80F9 327380          STA DDONE
80FC 21F3FF          LXI H,0FFF3H        ;INTERRUPT 7 VECTOR ADDR
80FF 119C81          LXI D,I7ADR         ;NEW INTERRUPT 7 JUMP ADDR
8102 73              MOV M,E
8103 23              INX H
8104 72              MOV M,D             ;JMP TO I7ADR FOR INTERRUPT

8105 3E70            MVI A,70H           ; 01   11   000   0    TIMER BYTE
                                         ;CTR1 2BYT MODE0 BNR
8107 D3DF            OUT 0DFH            ;TIMER CHIP CONTROL PORT

8109 2A6080          LHLD INTRVL         ; =1/256 OF SECOND
810C 7D              MOV A,L
810D D3DD            OUT 0DDH            ;TIMER 1 PORT
810F 7C              MOV A,H
8110 D3DD            OUT 0DDH

8112 3E08            MVI A,08H           ;SET Z80 MODE 0, NO PRIORITY.
8114 D3D7            OUT 0D7H            ;INTERRUPT CONTROLLER REGISTER

8116 FB              EI                  ;START CLOCK
8117 C9              RET

8118 3E96    BAUDA   MVI   A,096H        ; 10   01   001   0                ;SETS MODE, BAUD RATE, CONTROL FOR
                                         ; CTR2  LSB MODE3 BNR              ;PORT A (MAY BE USED TO MONITOR SYSTEM
811A D3DF            OUT   0DFH          ;SET COUNTER 2 MODE                ;ACTIVITY ON CRT TERMINAL)
811C 3E68            MVI   A,068H        ;1200 BAUD
811E D3DE            OUT   0DEH          ;COUNTER 2 REGISTER

8120 3E80    PORTA   MVI   A,80H         ;RESET PORT
8122 D3ED            OUT   0EDH          ;PORT A CONTROL/STATUS BYTE
8124 3E80            MVI   A,80H
8126 D3ED            OUT   0EDH
8128 3E40            MVI   A,40H
812A D3ED            OUT   0EDH
812C 3ECE            MVI   A,0CEH
812E D3ED            OUT   0EDH
8130 3E37            MVI   A,37H
8132 D3ED            OUT   0EDH
8134 DBED            IN    0EDH
8136 DBEC            IN    0ECH
8138 C9              RET

8139 3E9B    PARSET  MVI   A,09BH        ;SET ALL PARALLEL PORTS TO INPUT   ;INITIALIZE PARALLEL PORTS FOR INPUT
813B D3E7            OUT   0E7H          ;8255 CONTROL PORT
813D DBE5            IN    0E5H          ;PARALLEL PORT B (INITIAL
                                         ;A/C SPEED ESTIMATE)
813F 2F              CMA                 ;COMPLEMENT
8140 E60F            ANI   0FH           ;LOW ORDER FOUR BITS
8142 328180          STA   PROGSP        ;SAVE IT
```

Figure 1.1 - Continued

```
8145 210000          LXI H,0000H      ;ZERO DISTANCE                  ;INITIALIZE ANALOG BOARD
8148 220477          SHLD 0770AH
8148 213303          LXI H,0333H      ;'FLAG IN VIEW' VALUE
814E 220677          SHLD 07706H      ;RESET ANALOG BOARD
8151 C9              RET

                                      ;SET PORT B BAUD RATE           ;PORT B RECEIVES TDL-711 DATA STREAM
8152 3E16    BAUDB   MVI A,016H       ; 00   01   011    0
                                      ;CTR0  LSB MODE3  BNR
8154 D30F            OUT 0DFH         ;SET COUNTER 0 MODE
8156 0E06            MVI C,006H       ;= 20833 BAUD                   ;TDL-711 BAUD RATE
8158 DBE5            IN 0E5H          ;PARALLEL PORT B (BAUD SELECT)
815A E610            ANI 10H          ;BIT 4                          ;BIT 4 NOT SET SIGNALS
815C C26181          JNZ BAUDB1                                       ;TEST SET-UP (4800 BAUD)
815F 0E1A            MVI C,01AH       ;= 4800 BAUD
8161 79      BAUDB1  MOV A,C
8162 D30C            OUT 0DCH         ;COUNTER 0 REGISTER
8164 C9              RET

                                      ;INTERRUPT 6 VECTOR ADDRESS     ;SET UP INTERRUPT SCHEME TO
8165 21F0FF  PSETUP  LXI H,0FFF0H                                     ;RECEIVE LORAN-C DATA STREAM
8168 11D381          LXI D,I6ADR
816B 73              MOV M,E
816C 23              INX H
816D 72              MOV M,D          ;PATCH ADDR IN TABLE
816E AF              XRA A
816F 327280          STA POINTR       ;INITIALIZATIONS
8172 327380          STA DDONE
8175 3C              INR A
8176 327180          STA SOSEQ
8179 3E80            MVI A,80H        ;RESET PORT B                   ;BAUD RATE SET UP BY BAUDB ROUTINE
817B D3CF            OUT 0CFH
817D 3E80            MVI A,80H
817F D3CF            OUT 0CFH
8181 3E40            MVI A,40H
8183 D3CF            OUT 0CFH
8185 3ECE            MVI A,0CEH
8187 D3CF            OUT 0CFH
8189 3E37            MVI A,037H
818B D3CF            OUT 0CFH
818D DBCF            IN 0CFH
818F DBCE            IN 0CEH
8191 C9              RET

8192 216280  CRESET  LXI H,CLKADR     ;3-BYTE CLOCK ADDRESS           ;INITIALIZE CLOCK
8195 AF              XRA A            ;RESET CLOCK
8196 77              MOV M,A
8197 23              INX H
```

Figure 1.1 - Continued

```
8198 77              MOV M,A
8199 23              INX H
819A 77              MOV M,A
819B C9              RET

                                    ;TRANSFER POINT FOR CLOCK INTERRUPT          ;COUNTS TIME
819C F5        I7ADR PUSH PSW                                                    ;THIS POINT IS REACHED ON
819D E5              PUSH H                                                       ;INTERRUPT ONLY
819E 2A6280          LHLD CLKADR             ;LOW ORDER 2 BYTES
81A1 23              INX H                   ;TICK-TOCK
81A2 226280          SHLD CLKADR
81A5 7C              MOV A,H                 ;TEST FOR CARRY
81A6 B5              ORA L
81A7 C2B181          JNZ CLKNXT              ;NOT YET

81AA 3A6480          LDA CLKADR+2            ;HIGH ORDER BYTE
81AD 3C              INR A                   ;=256 SECONDS
81AE 326480          STA CLKADR+2

81B1 3E70    CLKNXT  MVI A,70H               ;RE-INITIALIZE TIMER  01 11 000 0    ;COUNTER-TIMER CHIP MUST BE
81B3 D3DF            OUT 0DFH                                                     ;RESET EACH TIME
81B5 2A6080          LHLD INTRVL             ;=1/256 SEC.
81B8 7D              MOV A,L
81B9 D3DD            OUT 0DDH
81BB 7C              MOV A,H
81BC D3DD            OUT 0DDH
81BE 3E..            MVI A,08H
81C0 D3D7            OUT 0D7H                ;INTERRUPT CONTROLLER

81C2 3A6280          LDA CLKADR              ;CALCULATE 1/16TH SEC INTERVAL       ;SET T16 TO FFH ON EVEN 16TH SEC.
81C5 E60F            ANI 0FH
81C7 C2CF81          JNZ CLKNI
81CA 3EFF            MVI A,0FFH
81CC 326E80          STA T16

81CF E1      CLKNI   POP H
81D0 F1              POP PSW
81D1 FB              EI                      ;START CLOCK AGAIN
81D2 C9              RET


                                    ;RECEIVE LORAN DATA BURST INTERRUPT           ;STORES LORAN DATA IN BUFFER
81D3 F5        I6ADR PUSH PSW                ;SAVE REGISTERS                     ;THIS POINT IS REACHED ON
81D4 C5              PUSH B                                                       ;INTERRUPT ONLY
81D5 D5              PUSH D
81D6 E5              PUSH H
81D7 DBCE            IN 0CEH                 ;GET DATA BYTE
81D9 47              MOV B,A
81DA 3E08            MVI A,08H               ;RESET INTERRUPT CONTROLLER
```


Figure 1.1 - Continued

```
81DC D307           OUT 0D7H
81DE FB             EI                  ;ENABLE INTERRUPTS
81DF 3A7180         LDA SOSEQ           ;START OF SEQ ?
81E2 B7             ORA A
81E3 CA0982         JZ REGSEQ
81E6 78             MOV A,B             ;YES
81E7 FEAA           CPI 0AAH            ;BYTE 'AA' ?            ;FIRST BYTE OF VALID DATA
81E9 C23082         JNZ RDONE                                   ;STREAM IS AAH
81EC 216280         LXI H,CLKADR        ;TIME OF DAY
81EF EB             XCHG
81F0 216580         LXI H,ARRTIM        ;SAVE AS ARRIVAL TIME   ;ARRIVAL TIME IS SAVED
81F3 F3             DI                  ;DISABLE INT (FREEZE CLOCK)
81F4 1A             LDAX D
81F5 77             MOV M,A
81F6 13             INX D
81F7 23             INX H
81F8 1A             LDAX D
81F9 77             MOV M,A
81FA 13             INX D
81FB 23             INX H
81FC 1A             LDAX D
81FD 77             MOV M,A             ;MOVE 3 BYTES
81FE FB             EI                  ;ENABLE INTERRUPTS

81FF AF             XRA A
8200 327180         STA SOSEQ
8203 327280         STA POINTR
8206 327380         STA DDONE

8209 210090 REGSEQ  LXI H,ARRAY         ;BUFFER                 ;STORE BYTE IN BUFFER
820C 3A7280         LDA POINTR          ;LOC IN BUFFER
820F 5F             MOV E,A
8210 AF             XRA A
8211 57             MOV D,A
8212 19             DAD D               ;COMPUTE ADDR
8213 70             MOV M,B             ;SAVE CHAR IN ARRAY
8214 3A7280         LDA POINTR
8217 3C             INR A               ;INCREMENT POINTER
8218 327280         STA POINTR
821B 47             MOV B,A
821C 3A7480         LDA PLIMIT
821F 90             SUB B               ;TEST AGAINST LIMIT
8220 C23082         JNZ RDONE
8223 3E01           MVI A,01            ;COMPLETE
8225 327380         STA DDONE           ;SET 'DONE'             ;TELLS MAIN ROUTINE THAT ANOTHER
8228 327180         STA SOSEQ           ;SET 'START OF SEQUENCE' ;UPDATE IS COMPLETE

822B 3E00           MVI A,00H
822D CD3B00         CALL 003BH          ;RING BELL              ;(DISABLED)
```

Figure 1.1 - Continued

```
8230 E1        RDONE   POP H                ;RESTORE REGISTERS
8231 D1                POP D
8232 C1                POP B
8233 F1                POP PSW
8234 C9                RET


                                            ;WHAT TIME IS IT?
8235 2A6280    TIME    LHLD CLKADR          ;LOW ORDER BYTES          ;USED ONLY AS A DIAGNOSTIC
8238 AF                XRA A                ;ZERO                     ;FROM CRT TYPE  GO 8043
8239 57                MOV D,A
823A 3A6480            LDA CLKADR+2         ;HIGH ORDER BYTE
823D 5F                MOV E,A
823E C36600            JMP 0066H            ;NON-MASK INTERRUPT LOC.
                                            ;TIME DISPLAYED IN DE,HL


8241 210090    HDISP   LXI H,ARRAY          ;DISPLAY TDL-711 BUFFER IN HEX  ;USED ONLY AS A DIAGNOSTIC
8244 0E9D              MVI C,9DH            ;157 BYTES                      ;FROM CRT TYPE  GO 8046
8246 CD0006    HLOOP   CALL 06D0H           ;CR LF
8249 0610              MVI B,10H            ;16 BYTES PER LINE
824B 7E        HLOOP1  MOV A,M
824C CD1D03            CALL 031DH           ;HEX OUTPUT ROUTINE
824F 3E20              MVI A,' '            ;SPACE
8251 CD4300            CALL 0043H           ;OUT 1
8254 23                INX H
8255 0D                DCR C
8256 CA6082            JZ HLOOP2            ;ALL DONE
8259 05                DCR B
825A CA4682            JZ HLOOP             ;LINE DONE
825D C34B82            JMP HLOOP1
8260 CD0006    HLOOP2  CALL 06D0H           ;CR LF
8263 C9                RET


                                            ;MAIN PROGRAM LOOP              ;BEGIN MAIN PROGRAM
8264 AF        MAIN    XRA A
8265 327580            STA LASTWP           ;INITIALIZATIONS                ;CLEAR LAST WAYPOINT BYTE
8268 CD0006            CALL 06D0H           ;CR LF                          ;CARRIAGE RETURN-LINE FEED TO CRT

826B CD7482    MLOOP   CALL TRACKR          ;TEST FOR NEW LORAN RECORD      ;TRACKR CHECKS FOR NEW LORAN DATA
826E CD2884            CALL DCALC           ;CALCULATE SMOOTHED DTW OUTPUT  ;DCALC UPDATES ANALOG OUTPUT
8271 C36B82            JMP MLOOP                                            ;ITERATE

8274 3A7380    TRACKR  LDA DDONE            ;NEW LORAN RECORD?              ;TRACKR ONLY HAS ACTIVITY WHEN A
8277 B7                ORA A                                                ;NEW LORAN DATA BURST IS RECEIVED
8278 C8                RZ                   ;NOT NEW
```

Figure 1.1 - Continued

```
8279 AF          XRA A
827A 327380      STA DDONE      ;RESET DDONE FLAG          ;PROCESS EACH DATA BURST ONLY ONCE
827D F3          DI             ;FREEZE LORAN DATA         ;DISABLE INTERRUPTS
827E 210090      LXI H,ARRAY    ;LORAN DATA-ADDRESS        ;GET DECIMAL POINTS, DIST TO WP,
8281 AF          XRA A                                     ;TO-FR WAYPOINTS
8282 57          MOV D,A
8283 1E09        MVI E,09H      ;DECIMAL POINTS LOCATION
8285 19          DAD D
8286 7E          MOV A,M        ;GET DECIMAL POINTS BYTE
8287 E608        ANI 08H        ;MASK 'ALL OTHER DEC. PTS.' BIT  ;MASK BIT 3
8289 327780      STA DECIML     ;SAVE IT
828C 23          INX H
828D 46          MOV B,M        ;DTW HIGH ORDER BYTE
828E 23          INX H
828F 4E          MOV C,M        ;DTW LOW ORDER BYTE
8290 60          MOV H,B
8291 69          MOV L,C
8292 227880      SHLD DTW       ;DISTANCE TO WAYPOINT 'DDD.D'  ;NOTE FOUR DIGIT DTW WORD
8295 210090      LXI H,ARRAY
8298 1E11        MVI E,11H      ;WAYPOINTS
829A 19          DAD D
829B 7E          MOV A,M
829C 326780      STA WP         ;TO/FR WAYPOINT NUMBERS 'DD'
829F FB          EI             ;ENABLE INTERRUPTS         ;RE-ENABLE INTERRUPTS

                 ;  PRINT LORAN-C SOURCE DATA                ;IF CRT IS CONNECTED, THIS
82A0 3A7780      LDA DECIML                                  ;DATA WILL APPEAR
82A3 CD1D03      CALL 031DH
82A6 3E20        MVI A,' '
82A8 CD4300      CALL 0043H
82AB 2A7880      LHLD DTW
82AE 7C          MOV A,H
82AF CD1D03      CALL 031DH
82B2 7D          MOV A,L
82B3 CD1D03      CALL 031DH
82B6 3E20        MVI A,' '
82B8 CD4300      CALL 0043H
82BB 3A7680      LDA WP
82BE CD1D03      CALL 031DH
82C1 3E20        MVI A,' '
82C3 CD4300      CALL 0043H
82C6 CD4300      CALL 0043H


82C9 AF          XRA A
82CA 32A780      STA VALID      ;WILL STAY UNSET IF TESTS FAIL  ;VALIDITY TEST
82CD 32E780      STA NEWLEG
82D0 3A7780      LDA DECIML     ;TEST DECIMAL POINTS       ;IF DECIMAL POINTS ARE LIT,OR
82D3 B7          ORA A
```


Figure 1.1 - Continued

```
82D4 CA0383            JZ ENDI
82D7 3A7680            LDA WP              ;TEST BLANK WAYPOINTS          ;IF ONE OR BOTH WAYPOINTS ARE BLANK,
82DA E6F0              ANI 0F0H                                           ;THEN DATA IS INVALID
82DC FEF0              CPI 0F0H
82DE CA0383            JZ ENDI             ;'FN'
82E1 3A7680            LDA WP
82E4 E60F              ANI 0FH
82E6 FE0F              CPI 0FH
82E8 CA0383            JZ ENDI             ;'NF'
82EB 3E01              MVI A,01H
82ED 327A80            STA VALID           ;VALID DATA STREAM
82F0 3A7580            LDA LASTWP
82F3 47                MOV B,A             ;B HAS LAST WP PAIR
82F4 3A7680            LDA WP
82F7 B8                CMP B
82F8 CA0383            JZ ENDI             ;NOT A NEW LEG
82FB 327580            STA LASTWP          ;SAVE NEW WP PAIR AS LAST      ;IF THIS WP PAIR <> LAST WP PAIR,
82FE 3E01              MVI A,01H                                          ;THEN THIS IS A 'NEW LEG'
8300 327E80            STA NEWLEG          ;SET 'NEW LEG' FLAG

8303 3A7A80    ENDI    LDA VALID
8306 B7                ORA A
8307 C8                RZ                  ;NOT VALID -- RETURN           ;NO FURTHER PROCESSING IF 'NOT VALID'
8308 3A7E80            LDA NEWLEG
830B B7                ORA A
830C CA4383            JZ UPDATE           ;NOT A NEW LEG -- NO CALCS NECESS.

830F CDE884            CALL SAVART         ;NEW LEG -- SAVE PARAMETERS    ;RE-INITIALIZE TRACK PARAMETERS
                                                                          ;FOR NEW LEG
8312 210000            LXI H,0000H         ;
8315 227F80            SHLD DELT           ;CLEAR DELTA TIME
8318 2A7880            LHLD DTW
831B 228B80            SHLD BCD16                                         ;DTW IN BCD FORMAT
831E CDEE85            CALL BCDCV          ;CONVERT TO F.P. FORMAT        ;CONVERT TO FIXED POINT BINARY
8321 2A8D80            LHLD OUT16                                         ;RETURNS OUT16
8324 228780            SHLD LASDTW         ;SAVE DTW IN F.P. FORMAT
8327 3A8180            LDA PROGSP          ;4 BITS READ FROM JUMPERS
832A 67                MOV H,A
832B 2E00              MVI L,00H           ;LOAD IN HL AS WHOLE NUMBER    ;FIXED POINT FORMAT
832D 229780            SHLD A16
8330 216C01            LXI H,016CH         ;=20/3600X256                  ;TIMES 20 TO GET KNOTS, DIV BY 3600
8333 229980            SHLD B16                                           ;TO GET MILES PER SEC, SCALE BY 256
8336 CD4A85            CALL M1616
8339 2A8D80            LHLD OUT16          ;SCALED SPEED
833C 228980            SHLD LASSPD         ;2 BYTE F.P. FORMAT   LAST SPEED
833F CD9E86            CALL PRINT          ;PRINT LINE OF DATA            ;IF CRT ATTACHED, DATA IS PRINTED
8342 C9                RET

8343 116580    UPDATE  LXI D,ARRTIM        ;THIS ARRIVAL TIME             ;UPDATE ESTIMATED DISTANCE AND SPEED
```

Figure 1.1 - Continued

```
8346 216880          LXI H,ARRTIM+03H    ;LAST ARRIVAL TIME
8349 016880          LXI B,ARRTIM+06H    ;DELTA ARRIVAL TIME          ;SUB24 CALCULATES DELTA ARR TIME
834C CD2585          CALL SUB24          ;FIND TIME CHANGE
834F 2A6880          LHLD ARRTIM+06H                                  ;RESULT
8352 22F780          SHLD DELT           ;SAVE DEL-TIME               ;SAVE TWO LOW-ORDER BYTES AS DELT

                                         ;ROUTINES TO TEST            ;TEST FOR PASSAGE OF TIME > TLIM
                                         ;DATA CONTINUITY

8355 016D80          LXI B,ARRTIM+08H    ;HIGH ORDER BYTE ADDRESS
8358 0A              LDAX B              ;GET HI-ORDER BYTE
8359 B7              ORA A
835A CA6583          JZ UP1              ;TIME LESS THAN 255 SEC

835D 3E01      TOLONG MVI A,01H
835F 32FE80          STA NEWLEG          ;DATA INVALID -- START OVER
8362 C30383          JMP END1

8365 2A8F80    UP1   LHLD TLIM           ;PRESET MAX TIME LIMIT
8368 EB              XCHG
8369 2AF780          LHLD DELT           ;DEL TIME (LOW ORDER 2 BYTES)
836C CD1D85          CALL SUB16          ;COMPARE
836F FA5D83          JM TOLONG

8372 AF              XRA A               ;CLEAR SIGN                  ;TEST FOR DISTANCE DELTA > 1 NMI
8373 32A380          STA SIGN
8376 2A7880          LHLD DTW
8379 228B80          SHLD BCD16
837C CDEE85          CALL BCDCV          ;CONVERT BCD TO F.P.
837F 2A8780          LHLD LASDTW
8382 EB              XCHG
8383 2A8D80          LHLD OUT16
8386 CD1D85          CALL SUB16          ;CALC DELTA DIST
8389 60              MOV H,B             ;(OUTPUT IN BC)
838A 69              MOV L,C
838B CDC685          CALL NEGTST         ;TAKE ABSOLUTE VALUE
838E 7C              MOV A,H             ;UNITS BYTE
838F B7              ORA A
8390 C25D83          JNZ TOLONG          ;DELTA DIST >1 MILE
8393 1614            MVI D,014H          ;20 MILES -- LIMIT ON DTW CALC  ;TEST FOR DIST TO GO > 20 NMI
8395 1E00            MVI E,00H
8397 2A8D80          LHLD OUT16          ;=DTW
839A CD1D85          CALL SUB16
839D FA5D83          JM TOLONG           ;>20 MILES

83A0 CDE884          CALL SAVART         ;RESET ARRIVAL TIME          ;STORE LORAN DATA ARRIVAL TIME
83A3 CDF884          CALL CDTW           ;ESTIMATE DTW                ;AS 'LAST ARRIVAL TIME'
83A6 2A8980          LHLD LASSPD                                      ;ESTIMATE DTW & SAVE AS NXTDTW
83A9 229580          SHLD NXTSPD         ;ESTIMATE SPEED              ;SAVE LASSPD AS NXTSPD ESTIMATE
```

Figure 1.1 - Continued

```
83AC 2A7880          LHLD DTN
83AF 228B80          SHLD BCD16                          ;CALCULATE 'DRIVE' WHICH IS FILTER
83B2 CDEE85          CALL BCDCV      ;CONVERT TO F.P. FORMAT  ;DRIVING FUNCTION (DIFFERENCE BETWEEN
83B5 2A8D80          LHLD OUT16                          ;MEASURED DTW AND ESTIMATED DTW)
83B8 EB              XCHG            ;DTW IN DE
83B9 2A9380          LHLD NXTDTW     ;ESTIMATED VALUE
83BC CD1D85          CALL SUB16      ;SUBTRACT
83BF 60              MOV H,B
83C0 69              MOV L,C
83C1 229180          SHLD DRIVE      ;FILTER DRIVER FUNCTION

                     ;CALCULATE ALPHA, BETA              ;GAINS ARE ELAPSED-TIME PROGRAMMED
83C4 2A9D80          LHLD ALPHA1     ;TIME-SENSITIVE GAIN ;CALCULATE DISTANCE GAIN
83C7 229780          SHLD A16
83CA 2A7F80          LHLD DELT
83CD 229980          SHLD B16
83D0 CD4A85          CALL M1616      ;GAIN TIMES TIME
83D3 2A8D80          LHLD OUT16
83D6 EB              XCHG
83D7 2A9B80          LHLD ALPHA0     ;GAIN OFFSET
83DA 19              DAD D           ;ALPHA GAIN IN HL
                     ;UPDATE LASDTW
83DB 229780          SHLD A16                            ;UPDATE DISTANCE ESTIMATE
83DE 2A9180          LHLD DRIVE
83E1 229980          SHLD B16
83E4 CD4A85          CALL M1616      ;ALPHA TIMES DRIVER
83E7 2A9380          LHLD NXTDTW
83EA EB              XCHG
83EB 2A8D80          LHLD OUT16
83EE 19              DAD D
83EF 228780          SHLD LASDTW     ;UPDATED FILTER      ;SAVE REVISED ESTIMATE

83F2 2AA180          LHLD BETA1      ;TIME-SENSITIVE GAIN ;CALCULATE VELOCITY GAIN
83F5 229780          SHLD A16
83F8 2A7F80          LHLD DELT
83FB 229980          SHLD B16
83FE CD4A85          CALL M1616      ;GAIN TIMES TIME
8401 2A8D80          LHLD OUT16
8404 EB              XCHG
8405 2A9F80          LHLD BETA0      ;GAIN OFFSET
8408 19              DAD D           ;BETA GAIN IN HL
                     ;UPDATE LASSPD                      ;UPDATE VELOCITY ESTIMATE
8409 229780          SHLD A16
840C 2A9180          LHLD DRIVE
840F 229980          SHLD B16
8412 CD4A85          CALL M1616      ;BETA TIMES DRIVER
8415 2A9580          LHLD NXTSPD
8418 EB              XCHG
8419 2A8D80          LHLD OUT16
841C CD1D85          CALL SUB16
```

Figure 1.1 - Continued

```
841F 60                 MOV H,B
8420 69                 MOV L,C
8421 228980             SHLD LASSPD     ;UPDATE FILTER          ;SAVE REVISED ESTIMATE
8424 CDE886             CALL PRINT                              ;PRINT RESULTS ON CRT
8427 C9                 RET                                     ;SUBROUTINE 'TRACKR' FINISHED

                                                                ;DCALC UPDATES ANALOG OUTPUTS
8428 DBE4       DCALC   IN 0E4H         ;READ PORT A - DIGISWITCH ANGLE ;DIGISWITCH SELECTS APPROACH ANGLE
842A 2F                 CMA             ;COMPLEMENT
842B 328280             STA ANGLE
842E 328B80             STA BCD16       ;SET UP BCD16 FOR CONVERSION
8431 AF                 XRA A
8432 328C80             STA BCD16+1                             ;SAVED AS 00D.D
8435 CDEE85             CALL BCDCV      ;CONVERT TO 16-BIT BINARY F.P.
8438 2A8D80             LHLD OUT16
843B 228380             SHLD ANGLEB     ;BINARY VERSION OF ANGLE ;F.P. BINARY VERSION OF APPROACH ANGLE

843E 2A8380             LHLD ANGLEB
8441 229780             SHLD A16
8444 2A8580             LHLD ANGLE3     ;CONVERSION CONSTANT    ;ANGLE3 CONVERTS RELATIVE TO A NOMINAL
8447 229980             SHLD B16                                ;3 DEG APP. ANGLE. ALSO SCALES X16
844A CD4A85             CALL M1616      ;MULTIPLY
844D 2A8D80             LHLD OUT16
8450 228380             SHLD ANGLEB     ;SAVE CONVERTED ANGLE

8453 3A6E80     WAIT    LDA T16
8456 B7                 ORA A
8457 CA5384             JZ WAIT         ;WAIT FOR T16 TO GIVE 1/16TH SEC INCR
845A AF                 XRA A
845B 326E80             STA T16         ;RESET IT
845E 327B80             STA FLAG        ;LOAD FLAG BYTE 'INVALID'
8461 3A7A80             LDA VALID       ;IF NOT VALID OR IF NEWLEG
8464 B7                 ORA A           ;THEN DROP FLAG
8465 CACE84             JZ SETFLG
8468 3A7E80             LDA NEWLEG
8468 B7                 ORA A
846C C2CE84             JNZ SETFLG      ;VALIDITY TESTS

846F 116280             LXI D,CLKADR    ;CURRENT CLOCK          ;COMPUTE TIME FROM LAST UPDATE
8472 216880             LXI H,ARRTIM+03H ;LASDTM TIME
8475 016B80             LXI B,ARRTIM+06H ;DELTA TIME
8478 CD2585             CALL SUB24
8478 2A6B80             LHLD ARRTIM+06H
847E 227F80             SHLD DELT       ;SAVE AS DELT           ;SAVE TWO LEAST SIGNIFICANT BYTES
8481 7C                 MOV A,H         ;TEST FOR TIME-OUT
8482 FE08               CPI 08H         ;8 SECONDS              ;FORCE A RESET IF >8 SECONDS
8484 FA8F84             JM CALCD                                ;SINCE LORAN-C UPDATE
8487 3E01       RESET1  MVI A,01H
8489 327E80             STA NEWLEG
848C C3CE84             JMP SETFLG
```

Figure 1.1 - Continued

```
848F CDF884    CALCD   CALL CDTW       ;CALCULATE CURRENT DTW AS NXTDTW          ;CALCULATE DTW FOR OUTPUT
8492 2A9380            LHLD NXTDTW     ;TEST FOR DIST < 0.2 NMI                  ;IF DTW <0.2 MILES, STOP GUIDANCE
8495 EB                XCHG
8496 213200            LXI H,0032H
8499 CD1D85            CALL SUB16
849C FAB784            JM RESET1       ;DROP FLAG
849F 2A9380    CALCD1  LHLD NXTDTW
84A2 229780            SHLD A16                                                  ;MULTIPLY DTW BY ANGLE
84A5 2A8380            LHLD ANGLEB     ;CONVERSION FACTOR                        ;CONVERSION FACTOR
84A8 229980            SHLD B16
84AB CD4A85            CALL M1616      ;MULTIPLY
84AE 2A8D80            LHLD OUT16
84B1 227C80            SHLD DTWC
84B4 229780            SHLD A16
84B7 2A6F80            LHLD PTIOH      ;DIVIDE BY 16                             ;SCALE BACK BY 16
84BA 229980            SHLD B16
84BD CD4A85            CALL M1616
84C0 2A8D80            LHLD OUT16
84C3 227C80            SHLD DTWC       ;SAVE AS DTWC
84C6 220477            SHLD 7704H      ;ANALOG BOARD OUTPUT                      ;OUTPUT ON ANALOG BOARD

84C9 3EFF              MVI A,0FFH
84CB 327880            STA FLAG        ;SET FLAG AS 'VALID'                      ;CALCULATE FLAG

84CE 210000    SETFLG  LXI H,0000H     ;FLAG VALID
84D1 3A7880            LDA FLAG
84D4 B7                ORA A
84D5 C2D884            JNZ SET1
84D8 213303            LXI H,0333H     ;FLAG INVALID
84DB 220677    SET1    SHLD 7706H      ;ANALOG BOARD FLAG OUTPUT                 ;OUTPUT ON ANALOG BOARD
84DE C9                RET                                                       ;SUBROUTINE 'DCALC' FINISHED

               ;************** END OF MAIN ROUTINE ****************              ;******* UTILITY ROUTINES FOLLOW *******

84DF 7D        SCA256  MOV A,L         ;ROUTINE TO ROUND AND SCALE               ;SCA256 ROUNDS AND SCALES
84E0 E680              ANI 080H        ;DIVIDED BY 256                           ;A 2-BYTE FIXED POINT WORD BY
84E2 87                ADD A           ;SHIFT TO CARRY BIT                       ;256 INTO A SINGLE BYTE (INT FUNCTION)
84E3 8C                ADC H
84E4 6F                MOV L,A
84E5 2600              MVI H,00H       ;FRACTION IN HL
84E7 C9                RET



84E8 116580    SAVART  LXI D,ARRTIM    ;ARRIVAL TIME ADDRESS                     ;SAVART MOVES ARRTIM INTO
84EB 216880            LXI H,ARRTIM+03H;'LAST ARRIVAL TIME'                      ;'LAST ARRIVAL TIME'
84EE F3                DI              ;FREEZE TIME VALUE
84EF 1A                LDAX D                                                    ;INTERRUPTS DISABLED TO FREEZE TIME
84F0 77                MOV M,A
```

Figure 1.1 - Continued

```
84F1 13              INX D
84F2 23              INX H
84F3 1A              LDAX D
84F4 77              MOV M,A
84F5 13              INX D
84F6 23              INX H
84F7 1A              LDAX D
84F8 77              MOV M,A          ;MOVE ARRTIM TO LAST ARRTIM
84F9 FB              EI               ;ENABLE INTERRUPTS
84FA C9              RET

84FB 2A8780  CDTW    LHLD LASSPD      ;COMPUTE DTW BASED ON TRACKER     ;CDTW ESTIMATES DTW (AS NXTDTW)
84FE 228780          SHLD A16         ;SPEED FOR MULTIPLY               ;BASED ON SPEED (LASSPD) AND TIME
8501 2A7F80          LHLD DELT                                          ;(DELT) AND LASDTW
8504 228980          SHLD B16         ;DEL TIME FOR MULTIPLY
8507 CDAA85          CALL M1616       ;MULTIPLY
850A 2A7780          LHLD LASDTW      ;LAST DTW VALUE
850D EB              XCHG
850E 2A8D80          LHLD OUT16       ;RESULT
8511 CDDE84          CALL SCA256      ;SCALE /256
8514 CD1D85          CALL SUB16       ;SUBTRACT DELTA DIST
8517 60              MOV H,B
8518 69              MOV L,C          ;MOVE TO HL
8519 228380          SHLD NXTDTW      ;ESTIMATED DTW
851C C9              RET


                     ;    SUB16 IS A 16-BIT SUBTRACTION ROUTINE
                     ;    A16-B16=C16
                     ;    C16 IS IN BC, A16 IS IN DE, B16 IS IN HL

851D AF              SUB16   XRA A                ;CLEAR A
851E 7B                      MOV A,E              ;LOAD A-LOW
851F 95                      SUB L                ;SUBTRACT B-LOW FROM A-LOW
8520 4F                      MOV C,A              ;STORE IT IN C-LOW
8521 7A                      MOV A,D              ;LOAD A-HIGH
8522 9C                      SBB H                ;SUBTRACT B-HI W/ BORROW
8523 47                      MOV B,A              ;STORE IT IN C-HI
8524 C9                      RET


                     ;
                     ;    SUB24 IS A 24 BIT SUBTRACTION ROUTINE
                     ;    A24-B24=C24
                     ;    (C24) IS IN BC, (A24) IS IN DE, (B24) IS IN HL
                     ;    () POINTS TO LOW ORDER BYTE
                     ;
8525 AF              SUB24   XRA A                ;CLEAR A
```


Figure 1.1 - Continued

```
8525 1A              LDAX D              ;LOAD A-LOW
8527 96              SUB M               ;SUBTRACT B-LOW FROM A-LOW
8528 02              STAX B              ;STORE IT IN C-LOW
8529 23              INX H               ;INCREMENT THE REGISTERS
852A 13              INX D
852B 03              INX B
852C 1A              LDAX D              ;LOAD A-MID
852D 9E              SBB M               ;SUBTRACT B-MID AND CARRY FROM A-MID
852E 02              STAX B              ;STORE IT IN C-MID
852F 23              INX H               ;INCREMENT REGISTERS
8530 13              INX D
8531 03              INX B
8532 1A              LDAX D              ;LOAD A-HI
8533 9E              SBB M               ;SUBTRACT B-HI AND CARRY FROM A-HI
8534 02              STAX B              ;STORE IT IN C-HI
8535 C9              RET


                     ;
                     ;       MULT8--AN 8-BIT MULTIPLICATION ROUTINE
                     ;
                     ;       THE MULT8 ROUTINE EFFECTS THE FOLLOWING: A X B = C
                     ;       A IS IN REGISTER A
                     ;       B IS IN REGISTER E
                     ;       D IS ASSUMED TO HAVE BEEN MADE ZERO BY THE CALLING ENTITY
                     ;       C IS THE 16-BIT RESULT IN HL
                     ;
8536 0608    MULT8   MVI     B,08H       ;LOAD COUNTER
8538 1600            MVI     D,00H
853A 210000          LXI     H,00H       ;ZERO OUT HL
853D 1F      ROTAT3  RAR                 ;ROTATE A TO THE RIGHT
853E D24285          JNC     MLTI        ;NO ADD IF NON-CARRY
8541 19              DAD     D           ;ADD DE AND HL IN HL
8542 EB      MLTI    XCHG                ;EXCHANGE DE AND HL
8543 29              DAD     H           ;HL=2XHL (SHIFT HL LEFT)
8544 EB              XCHG                ;EXCHANGE DE AND HL
8545 05              DCR     B           ;DECREMENT COUNTER
8546 C23D85          JNZ     ROTAT3      ;IF NOT ZERO YET, DO IT AGAIN
8549 C9              RET                 ;DONE--RESULT IN HL


                     ;  M1616--A SUBROUTINE TO MULTIPLY TWO 16-BIT SIGNED FIXED-POINT NUMBERS
                     ;
                     ;       OUT -- RESULT16 = A16 * B16             OUT16= A16 * B16
                     ;       THIS ROUTINE MULTIPLIES TWO XX.XX NUMBERS TO GET A XX.XX NUMBER
                     ;
854A 210000  M1616:  LXI     H,0000H     ;LEAST SIGNIFICANT BITS ARE LOST
```

Figure 1.1 - Continued

```
8540 228D80          SHLD    OUT16           ;CLEAR OUTPUT -- RESULT TOO LARGE FORCES OVERFLOW CC REGISTER
8543 110000          LXI     D,0000H
8546 0E01            MVI     C,01H           ;C=1 MEANS OVERFLOW
8548 AF              XRA     A
8549 329380          STA     SIGN            ;SIGN SAVER
854C 2A9780          LHLD    A16
854F CDC685          CALL    NEGTST          ;IF NEGATIVE, COMPLEMENT
8552 229780          SHLD    A16
8555 2A9980          LHLD    B16
8558 CDC685          CALL    NEGTST
855B 229980          SHLD    B16
855E 3A9980          LDA     B16             ;LOW B BYTE
8561 5F              MOV     E,A
8562 3A9780          LDA     A16             ;LOW A BYTE
8565 CD3685          CALL    MULT8
8568 7D              MOV     A,L             ;ROUND-UP ROUTINE
8569 E680            ANI     080H            ;HIGH BIT OF LOW BYTE
856B 87              ADD     A               ;SHIFT INTO CARRY BIT
856C 8C              ADC     H               ;ADD WITH CARRY
856D 328D80          STA     OUT16           ;LOW OUT BYTE
8570 3A9980          LDA     B16             ;LOW B BYTE
8573 5F              MOV     E,A
8574 3A9880          LDA     A16+1           ;HIGH A BYTE
8577 CD3685          CALL    MULT8
857A EB              XCHG                    ;SWAP DE,HL
857B 2A8D80          LHLD    OUT16
857E 19              DAD     D               ;ADD TO OUT
857F 228D80          SHLD    OUT16
8582 3A9A80          LDA     B16+1           ;HIGH B BYTE
8585 5F              MOV     E,A
8586 3A9780          LDA     A16             ;LOW A BYTE
8589 CD3685          CALL    MULT8
858C EB              XCHG
858D 2A8D80          LHLD    OUT16
8590 19              DAD     D
8591 228D80          SHLD    OUT16
8594 3A9A80          LDA     B16+1           ;HIGH B BYTE
8597 5F              MOV     E,A
8598 3A9880          LDA     A16+1           ;HIGH A BYTE
859B CD3685          CALL    MULT8
859E 7C              MOV     A,H
859F B7              ORA     A
85A0 C0              RNZ                     ;OVERFLOW
85A1 55              MOV     D,L
85A2 AF              XRA     A
85A3 5F              MOV     E,A
85A4 2A8D80          LHLD    OUT16
85A7 19              DAD     D
85A8 228D80          SHLD    OUT16
85AB AF              XRA     A
```


Figure 1.1 - Continued

```
85B9 4F              MOV     C,A
85BA 3AA380          LDA     SIGN
85BD B7              ORA     A
85BE C8              RZ                      ;POSITIVE RESULT
85BF CDD785          CALL    COMPI6          ;COMPLEMENT RESULT
85C2 228D80          SHLD    OUTI6
85C5 C9              RET                     ;FINISHED !!

                     ;NEGTST--TEST FOR NEGATIVE 16-BIT F.P. VALUE & COMPLEMENT
                     ;INPUT AND OUTPUT THROUGH HL
85C6 7C      NEGTST  MOV     A,H             ;TEST HIGH ORDER BIT
85C7 E680            ANI     080H
85C9 C8              RZ                      ;POSITIVE NUMBER
85CA CDD785          CALL    COMPI6          ;COMPLEMENT HL
85CD 3AA380          LDA     SIGN
85D0 3C              INR     A
85D1 E601            ANI     01H
85D3 32A380          STA     SIGN            ;TRACK SIGN OF PRODUCT
85D6 C9              RET

85D7 7C      COMPI6  MOV     A,H             ;2'S COMPLEMENT OF HL
85D8 2F              CMA
85D9 67              MOV     H,A
85DA 7D              MOV     A,L
85DB 2F              CMA
85DC C601            ADI     01H             ;ADD 1
85DE 6F              MOV     L,A             ;RESAVE
85DF 3E00            MVI     A,00H           ;CLEAR A -- LEAVE CARRY BIT
85E1 8C              ADC     H               ;CARRY INTO H
85E2 67              MOV     H,A
85E3 C9              RET

                     ;  CONVERT BCD DDD.D TO 16 BIT BINARY FLOATING POINT
                     ;
                     ;       INPUT IS BCDI6          ;DDD.D
                     ;       OUTPUT IS OUTI6         ;XX.XX
                     ;                               ;  RESULT TOO LARGE OR NON-BCD DIGIT FORCES OVERFLOW (C REGISTER)
85E4 001A334D66BCDCNV   DB      00H,01AH,033H,04DH,066H
85E9 809AB3CDE6         DB      080H,09AH,0B3H,0CDH,0E6H

85EE 210000  BCDCV:  LXI     H,0000H
85F1 228D80          SHLD    OUTI6           ;CLEAR OUTPUT
85F4 1600            MVI     D,00H
85F6 0E01            MVI     C,01H           ;C=1 MEANS OVERFLOW
85F8 3A8B80          LDA     BCDI6           ;LOW BCD BYTE
85FB E60F            ANI     0FH             ;MASK DECIMAL DIGIT
```

Figure 1.1 - Continued

```
85ED FE0A          CPI     0AH
85EF F0            RP                      ;ILLEGAL CHARACTER
8600 21E485        LXI     H,BCDCNV
8603 5F            MOV     E,A
8604 19            DAD     D
8605 7E            MOV     A,M             ;FRACTIONAL BYTE
8606 32BD80        STA     OUT16           ;LOW ORDER BYTE
8609 3AB880        LDA     BCD16
860C E6F0          ANI     0F0H            ;MASK UNITS DIGIT
860E 0F            RRC                     ;SHIFT
860F 0F            RRC
8610 0F            RRC
8611 0F            RRC
8612 FE0A          CPI     0AH
8614 F0            RP                      ;ILLEGAL CHARACTER
8615 32BE80        STA     OUT16+1         ;HI ORDER BYTE
8618 3ABC80        LDA     BCD16+1         ;HIGH BCD BYTE
861B E60F          ANI     0FH             ;TENS DIGIT
861D FE0A          CPI     0AH
861F F0            RP                      ;ILLEGAL CHARACTER
8620 5F            MOV     E,A
8621 3E0A          MVI     A,0AH           ;MULTIPLY BY 10
8623 CD3685        CALL    MULT8
8626 3ABE80        LDA     OUT16+1         ;HIGH BYTE
8629 85            ADD     L               ;UNITS & TENS
862A D8            RC                      ;OVERFLOW
862B 32BE80        STA     OUT16+1         ;HIGH BYTE
862E 3ABC80        LDA     BCD16+1         ;HIGH BCD
8631 E6F0          ANI     0F0H            ;MASK HUNDREDS
8633 0F            RRC                     ;SHIFT
8634 0F            RRC
8635 0F            RRC
8636 0F            RRC
8637 FE0A          CPI     0AH
8639 F0            RP                      ;ILLEGAL CHARACTER
863A 5F            MOV     E,A
863B 3E64          MVI     A,064H          ;MULTIPLY BY 100
863D CD3685        CALL    MULT8
8640 7C            MOV     A,H
8641 B7            ORA     A
8642 C0            RNZ                     ;OVERFLOW
8643 3ABE80        LDA     OUT16+1         ;HIGH BYTE
8646 85            ADD     L               ;UNITS, TENS, HUNDREDS
8647 D8            RC                      ;OVERFLOW
8648 32BE80        STA     OUT16+1         ;HIGH BYTE
864B AF            XRA     A               ;0
864C 4F            MOV     C,A             ;RESET ERROR FLAG
864D C9            RET                     ;FINISHED !!
```

Figure 1.1 - Continued

```
                        ;ROUTINE TO CONVERT HEX F.P. NUMBERS TO BCD  DD.DD
                        ;INPUT AND OUTPUT IN HL
                        ;INPUT XX.XX
                        ;OUTPUT DDD.D
864E F5         CVBCD   PUSH H                  ;SAVE ARGUEMENT
864F 7C                 MOV A,H
8650 CD8386             CALL BBCD               ;CONVERT UNITS BYTE TO BCD
8653 E1                 POP H                   ;RESTORE ARGUEMENT
8654 F5                 PUSH A                  ;SAVE UNITS BCD
8655 2600               MVI H,00H               ;CLEAR UNITS PART
8657 229780             SHLD A16
865A 210064             LXI H,6400H             ;=100 IN F.P.
865D 229980             SHLD B16
8660 CD4A85             CALL M1616
8663 2A8D80             LHLD OUT16              ;FRACTIONAL PART X100
8666 7D                 MOV A,L
8667 E680               ANI 080H                ;MASK HIGH BIT OF LOW BYTE
8669 87                 ADD A                   ;SHIFT INTO CARRY BIT
866A 8C                 ADC H                   ;ADD FRAC X100
                                                ;ROUNDS-OFF THE VALUE
866B 0600               MVI B,00H
866D FE64               CPI 064H                ;>99 ?
866F FA7386             JM CVBCD1
8672 04                 INR B                   ;CARRY INTO UNITS DIGITS
8673 CD8386     CVBCD1  CALL BBCD               ;CONVERT TO BCD
8676 6F                 MOV L,A
8677 F1                 POP A                   ;RECOVER UNITS DIGITS
8678 80                 ADD B                   ;ADD CARRY BIT
8679 27                 DAA                     ;RESULT IN BCD
867A 67                 MOV H,A                 ;4 BCD DIGITS IN HL
867B C9                 RET

                        ;ROUTINE TO CONVERT BYTE TO BCD  DD
                        ;INPUT AND OUTPUT IN A                  ;INPUT XX
867C 001632486480BCDT   DB 00H,16H,32H,48H,64H,80H,96H          ;OUTPUT DD

8683 F5         BBCD    PUSH A                  ;SAVE ARGUEMENT
8684 E6F0               ANI 0F0H                ;MASK HIGH ORDER NYBBLE
8686 0F                 RRC
8687 0F                 RRC
8688 0F                 RRC
8689 0F                 RRC                     ;SHIFT TO LOW POSITION
868A 217C86             LXI H,BBCDT
868D 5F                 MOV E,A
868E 1600               MVI D,00H               ;LOAD ADDRESS IN TABLE
8690 19                 DAD D
8691 F1                 POP A                   ;RESTORE ARGUEMENT
8692 E60F               ANI 0FH                 ;MASK LOW ORDER NYBBLE
8694 FE0A               CPI 0AH                 ;>=10 ?
```

Figure 1.1 - Continued

```
8695 FA9B86            JM BBCD1
8699 C606              ADI 06H                 ;IF SO, ADD 6 TO CONVERT TO BCD
869B 86        BBCD1   ADD M                   ;ADD HIGH NYBBLE IN BCD
869C 27                DAA                     ;RESULT IN BCD
869D C9                RET


                       ;ROUTINE TO PRINT RESULTS OF TRACKER UPDATE
                       ;IF CRT IS ATTACHED, THIS DATA IS PRINTED
                       ;
869E 2A8780    PRINT   LHLD LASDTW             ;DISTANCE TO WP
86A1 CD4E86            CALL CVBCD
86A4 7D                MOV A,L
86A5 F5                PUSH A
86A6 7C                MOV A,H
86A7 CD1D03            CALL 031DH              ;HEX OUTPUT ROUTINE
86AA 3E2E              MVI A,'.'
86AC CD4300            CALL 0043H
86AF F1                POP A
86B0 CD1D03            CALL 031DH
86B3 3E20              MVI A,' '
86B5 CD4300            CALL 0043H

86B8 2A8980            LHLD LASSPD             ;CURRENT GROUNDSPEED ESTIMATE
86BB 229780            SHLD A16
86BE 210E0E            LXI H,0E10H             ;=3600/256  SCALES TO KNOTS
86C1 229980            SHLD B16
86C4 CD4A85            CALL M1616
86C7 0600              MVI B,00H
86C9 2A8D80            LHLD OUT16
86CC 7C                MOV A,H
86CD FE64              CPI 064H                ;>=100 ?
86CF FAD786            JM PRINT1
86D2 0601              MVI B,01H               ;HUNDRED'S DIGIT
86D4 D664              SUI 064H
86D6 67                MOV H,A                 ;SUB 100 FROM VALUE
86D7 78        PRINT1  MOV A,B
86D8 CD1D03            CALL 031DH              ;PRINT HUNDREDS PLACE
86DB CD4E86            CALL CVBCD              ;CONVERT TO BCD
86DE 7D                MOV A,L
86DF F5                PUSH A
86E0 7C                MOV A,H
86E1 CD1D03            CALL 031DH
86E4 3E2E              MVI A,'.'
86E6 CD4300            CALL 0043H
86E9 F1                POP A
86EA CD1D03            CALL 031DH
86ED 3E20              MVI A,' '
86EF CD4300            CALL 0043H
86F2 3A8280            LDA ANGLE
```

Figure 1.1 - Continued

```
86F5 CD1D03          CALL 031DH          ;DISPLAY DIGI-SWITCH ANGLE
86F8 CD0D06          CALL 06D0H          ;CR LF
86FB C9              RET
```


Figure 1.1 - Continued

3) Software Initializations 80DEH
   - Initialize second serial port (Port B) and its baud rate.
   - Initialize parallel port and read initial groundspeed estimate.
   - Patch appropriate interrupt addresses into the interrupt table.
   - Set up interrupt routines for the clock and the Loran-C data stream.

4) Clock Interrupt Handler Routine 819CH

5) Loran-C Interrupt Handler Routine 81D3H
   - Receive data and load into a buffer

6) Main Routine (Described below) 8264H

7) Utility Routines 82DFH
   - Round and scale
   - Save arrival time
   - Compute DTW
   - 16 bit subtract
   - 24 bit subtract
   - 8 bit multiply
   - 16 bit sign test & complement
   - Convert BCD to binary
   - Convert binary to BCD

The main routine consists of two major subroutines. The first, TRACKR (8274H), is the heart of the system. Its function is to update the tracking algorithm when a new Loran-C data stream has been received. It also tests the validity of the incoming data and drops the DME valid flag when necessary. The other subroutine, DCALC (8428H), calculates DTW sixteen times per second and controls the D/A converter. A flowchart showing the basic functions of the software package appears as Figure 1.2.

The TRACKR subroutine updates the DTW and groundspeed estimates (using the equations in Section 1.4). It also examines the Loran-C

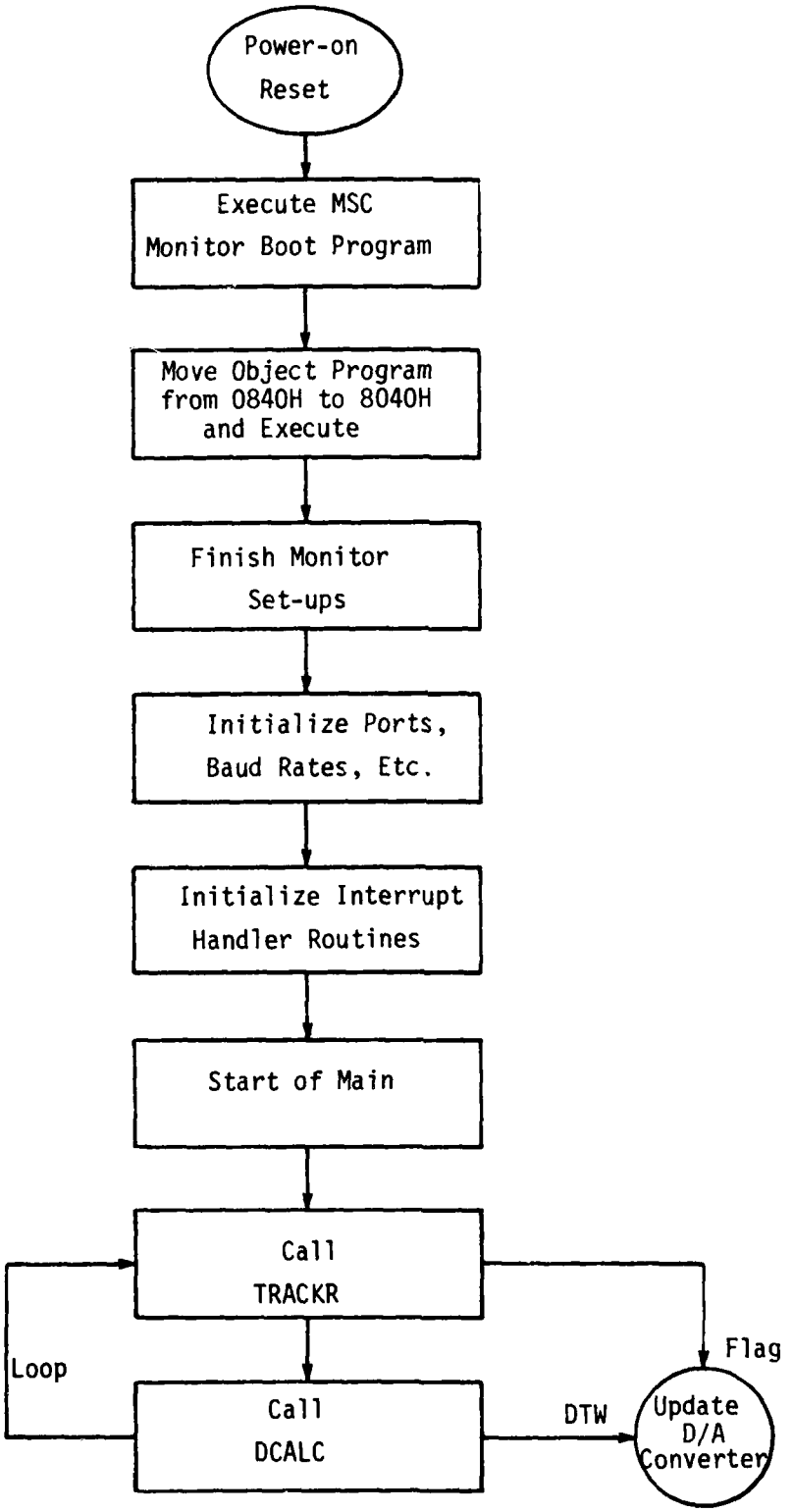


Figure 1.2 Overall Program Flow

data stream for determination of data validity and continuity. When data is invalid or discontinuous the DME valid flag is dropped. This subroutine performs an initialization function if one of several criteria are met: The waypoint pair is changed, the time since last update is greater than 16 sec, the change in Loran-C DTW exceeds one mile, or the DTW value itself exceeds 20 miles (indicating that an instrument approach is not in progress). A flow chart for the TRACKR subroutine appears as Figure 1.3.

The DCALC subroutine calculates current DTW estimate. This estimate is scaled by a factor derived from the pilot's approach angle selector. Any approach angle up to 9.9° may be selected. The flow chart for this subroutine appears as Figure 1.4.

1.6 SYSTEM TEST

Hardware and software checkout was performed on the bench using another computer system as a TDL-711 data stream simulator. System performance was evaluated over a broad range of off-nominal operating conditions including:

1) Velocity estimate wrong by ±30 kts.
2) Data update delays ranging from 1 to 6 seconds.
3) Loran-C DTW random error magnitudes up to 0.1 nm (1σ).
4) Various Loran-C interruptions and failure modes.

Final system checkout was performed by installing the system in the Beech Queen Air operated by this company for navigation system test purposes. The complete system (TDL-711, Interface Unit, IDC VNAV, IDC Altimeter and approach angle selector) was installed in a normal operational configuration. The system was test flown on three different occasions. Approaches using the Loran-C/VNAV combination were conducted at Palm Beach International, Pahokee/Palm Beach Glades, and Sarasota-Bradenton airports. In all tests the guidance information was quite smooth and accurate (given the basic limitations of the Loran-C system). Approaches up to 7° were flown.

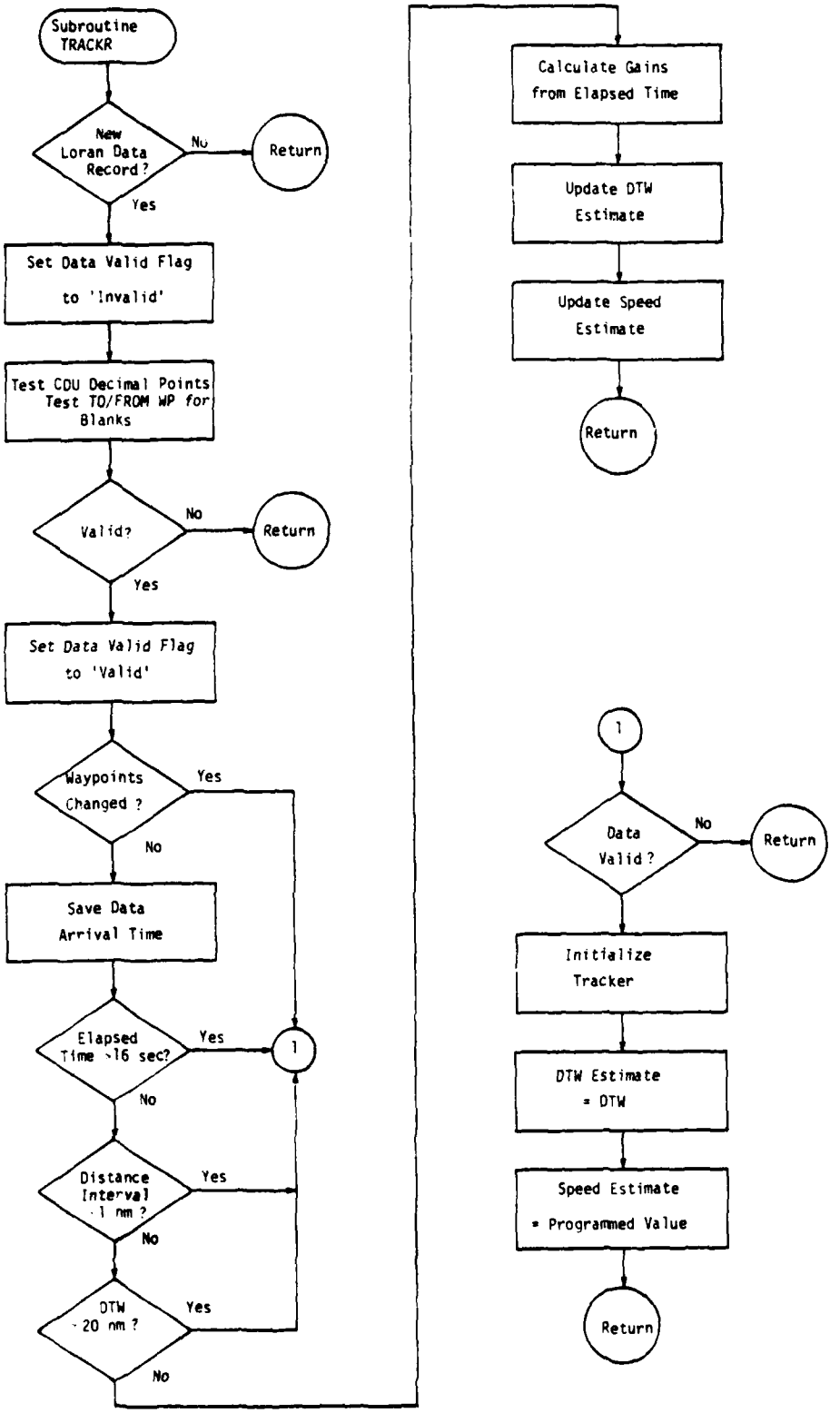


Figure 1.3 Subroutine TRACKR

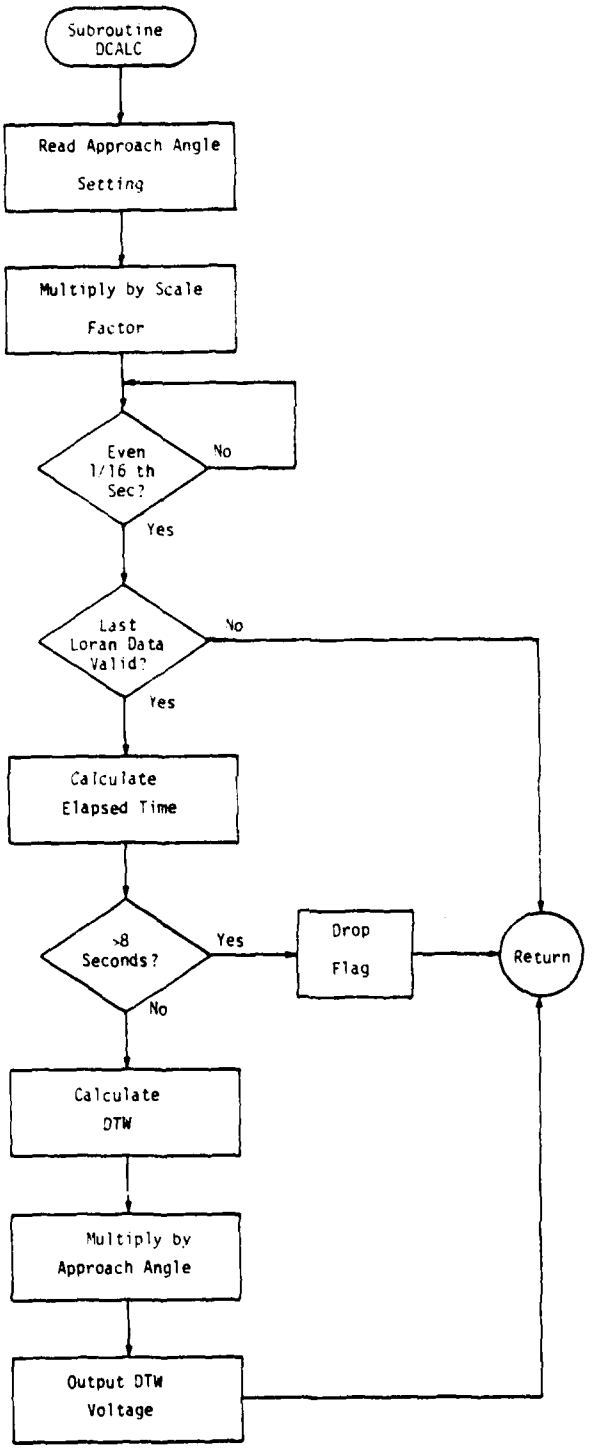


Figure 1.4 Subroutine DCALC

2.0 3D LORAN-C NAVIGATOR HARDWARE

Introduction

The main purpose of the 3D Loran-C navigator is to provide let-down guidance for an equipped helicopter on a Loran-C non-precision approach. This hardware description describes the materials necessary to fabricate a proof-of-concept "bread board" demonstration unit to provide let-down guidance in a Loran-C environment. The unit is interfaced with a TDL-711 Loran-C navigator, an IDC VNAV system and a barometric altimeter. Certain factors that were addressed during this project are as follows:

- TDL-711 Loran-C navigator distance-to-go input
- Barometric altitude input
- Input data rates and filter requirements
- Output displayed on 2 each 1000 Ω CDI's
- Validation/flag for guidance signal
- Helicopter speeds down to 20 kts
- Descent angles of 3 to 9 degrees with 6° nominal
- The expected performance envelope

Figure 2.1 shows a block diagram for the 3D Loran-C navigator. The system has been divided into three major areas: test equipment, the 3D Loran-C navigator and aircraft equipment. The figure shows how each component from the three major areas interacts with the other components in the system. Each component played a key role in either the development/testing or actual operational use of the 3D Loran-C navigator.

Basically, the 3D Loran-C navigator was developed so that the entire system could be tested in a laboratory environment. As shown in Figure 2.1 a computer terminal and its accompanying computer were used to test and develop the navigator's software. The computer terminal was used for display purposes while the computer was configured to simulate the TDL-711 data stream.

Normally the VNAV Alert unit receives DME (distance) data from the aircraft's onboard DME receiver. It in turn, based on altitude information received from the altimeter, field elevation set on the VNAV Alert unit, and a predetermined glide path angle, computes a

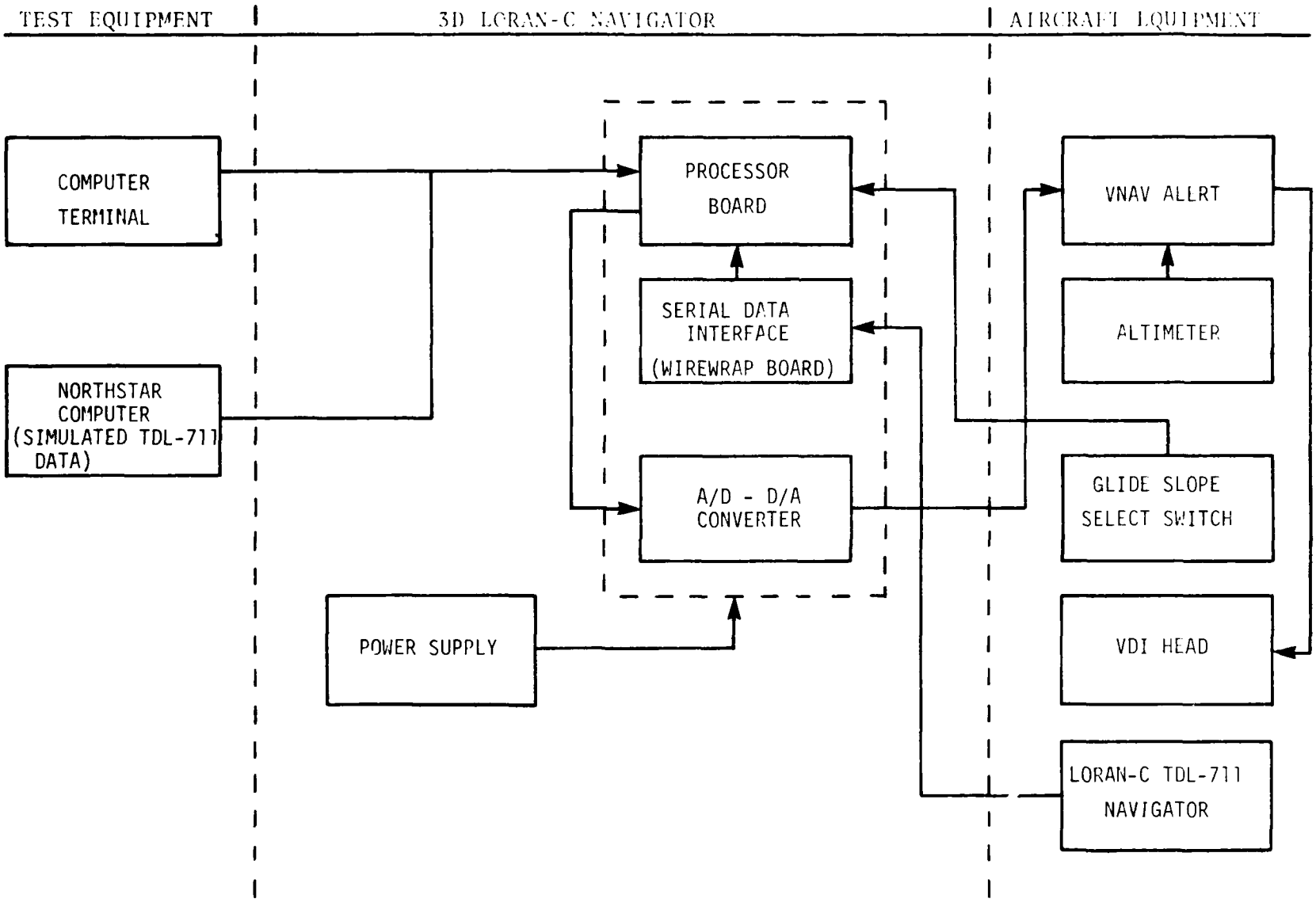


Figure 2.1 3D Loran-C Navigator Block Diagram

vertical deviation from the nominal 3 degree glide path and transmits this information to the VDI (vertical deviation indicator). For this particular operation, the following steps occur. The serial data interface receives DTW (distance to waypoint) information, along with various other necessary items, from the TDL-711 RPU and transmits the information to the processor board. The processor board smooths and filters the TDL-711 data (it is necessary to smooth the data because data is transmitted from the TDL-711 RPU at sporadic intervals, 1-6 sec) and also receives glide path angle data from the glideslope select switch. After certain computations are completed and this information is transmitted to the D/A converter, an analog distance voltage is sent to the VNAV Alert unit. From this point on, the system operates normally as discussed earlier in this paragraph.

The following sections describe the seven major components necessary to fabricate the 3D Loran-C navigator. Each component is described in detail along with the necessary interconnect descriptions and modifications necessary to make the unit function. The seven major components are as follows:

- MSC* 8201 Multibus Chassis
- MSC 8204 Universal Card
- MSC 8007 Z-80 Computer Board
- Burr-Brown D/A Converter Board
- Power Supply (DATEL ±12V & ±5V)
- 3D Loran-C Chassis
- Aircraft Interconnect

2.1 MULTIBUS CHASSIS

The Multibus** Chassis used for the 3D Loran-C system was a Monolithic Systems Corp., MSC 8201. The chassis is multibus compatible and provides seven (7) slots for various computer card combinations. The MSC 8201 chassis is a standard RETMA rack mount, (i.e., 5.25" high, 19.00" wide and 8.50" deep). The chassis also offers a rigid .090" thick printed circuit backplane. Momentary rocker switches are provided for RESET and NMI (Non-Maskable Interrupt), with the RESET LED indicator being red and

/NOTE/ *MSC (Monolithic Systems Corporation)
**Registered Trade Mark of Intel Corporation

the NMI indicator being green [1].

The DC power connections (±12V, ±5V, GND) were soldered directly into the backplane. Five (5) wires were required to make the appropriate connections into the backplane with a six (6) pin male Molex connector (Waldrom Part #1261 PRT) providing the connection from the DATEL PCT-12/2-5/6 power supply. Figure 2.2 shows how the Molex connector was wired as viewed into the male receptacle.

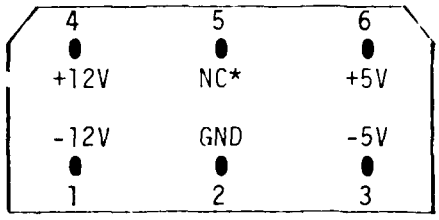


Figure 2.2 Multibus Power Connector

Table 2.1 presents the connector wire and its appropriate pin number (on the backplane) information.

Table 2.1 Multibus Backplane Power Supply Wiring

Backplane Pin#	Voltage
3/4	+5V
7/8	+12V
9/10	-5V
11/12	GND
79/80	-12V

Appendix A shows the "MCS 8201 Maintenance Manual Multibus Chassis". The appendix offers all of the added information, not mentioned above, necessary to operate the system.

/NOTE/ *NC - No Connection

2.2 WIREWRAP BOARD

The wirewrap board utilized for the 3D Loran C navigator was a Monolithic Systems Corporation, MSC 8204. The MSC 8204 Universal Card is for custom prototyping of Multibus* compatible systems. The MSC 8204 will hold a matrix of wirewrap sockets that accommodate any length components on width of 0.3, 0.4 and 0.6 inches. The MSC 8204 offers one, 26-pin I/O connector and two, 50-pin I/O connectors. The MSC 8204 accepts either a standard edge connector or the holes at the bottom of each conductive surface permit the mounting of a right-angle header connector. Power and ground planes terminate on uniformly-spaced pads for ease of access.

The MSC 8204 contains its own programmable, bus-master control system. Full address decoding permits the use of either memory mapped or normally isolated I/O addressing. Also, the MSC 8204 contains Transfer-Acknowledge logic and the option of including inhibits. See Appendix B for more detailed address bus, data bus, and control signals specifications.

The wirewrap board was used solely to interface the main computer board with the TDL-711 RDU data stream. Figure 2.3 shows the circuit diagram required to accomplish this task. The data interface required the use of one chip (DS 8820) (Appendix C) and several wirewrap connections on the MSC 8204 board. To accomplish this task, a 14 pin socket was installed at location D13 on the MSC 8204 board. After the chip (DS 8820) was installed, pins 1 and 3 were wirewraped to pins 10 and 12 on the 26 pin connector, respectively. Pin 13 on the 26 pin connector was connected to ground on the MSC 8204 board. Pin 6, which is the output of chip DS 8820, is wirewraped to pin 32 on the backplane connector of the board. This output (pin 6) is connected to the UART 8251 (RXD 3) chip on the main computer board. See the discussion of modifications made to the computer board in Section 2.3. As shown in Figure 2.3, two capacitors are required - a 0.01 μf capacitor wired across pins 1 and 2, and a 100 pf capacitor wired across pin 5 and ground. Pin 4 on the DS 8820 is wirewrapped for +5 volts.

/NOTE/ *Registered Trade Mark of Intel Corporation

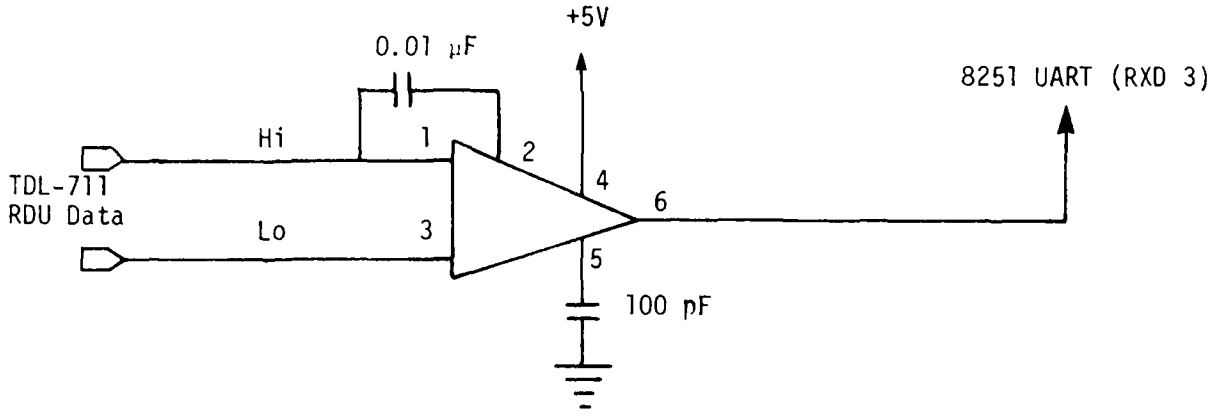


Figure 2.3 TDL-711 RDU Data Receiver [2]

2.3 MICROPROCESSOR BOARD

DESCRIPTION:

The microprocessor board used for the 3D Loran-C navigator was a Monolithic System Corporation MSC 8007. The Monolithic System Corporation MSC 8007 is a single-board OEM computer that is directly compatible with the industry standard MULTIBUS*. The 8080 software compatibility of the MSC 8007 provides an advanced, high-speed, next generation system designed to take full advantage of the Z80A** instruction set[3]. The Z80A microprocessor is fully compatible with the 8080A. The Z80A executes all 8080 instructions without modification.

SERIAL COMMUNICATIONS:

Three serial ports provide programmable interfaces that are synchronous or asynchronous. One port can be configured for either EIA RS-232-C, TTL or opto-isolated 20 mA loop signals. Two ports can be configured for EIA RS-232-C or TTL only. Signal specifications are:

Synchronous:

5- to 8-bit character; one or two programmable SYNC characters.

/NOTE/ *MULTIBUS is a registered trademark of Intel Corporation
**Z80A is a registered trademark of Zilog, Inc.

Asynchronous:

5- to 8-bit character; 1, 1-1/2, or 2 stop bits; choice of parity or error detection.

PARALLEL I/O:

One parallel, peripheral-interface controller (24 lines total) is used. All signals are both buffered and terminated on input and output configurations. Inverting and non-inverting transceivers are supplied.

INTERFACE SIGNAL:

MULTIBUS:

All signals conform with MULTIBUS specifications.

Parallel I/O:

All signals are TTL compatible.

Serial I/O:

Signals fulfill either EIA RS-232-C, TTL or 20 mA current loop convention depending on the MSC 8007 configuration.

Interrupt Requests:

All signals are TTL compatible.

Timer:

All signals are TTL compatible.

POWER REQUIREMENTS:

The MSC 8007 operates with MULTIBUS power supply voltages of +5V, -5V, +12V, and -12V.

PHYSICAL DIMENSIONS:

12 in. (Width) X 6.75 in. (Height) X 0.5 in. (Depth).

System options:

The MSC 8007 transmits and receives data either in parallel or serial form. Three serial I/O interfaces provide the MSC 8007 with serial data communication channels that are programmable and will operate either synchronously or asynchronously. Two of these serial

options were selected for the 3D Loran-C navigator. The serial I/O interfaces are designed around two software programmable devices - an 8251 USART and an 8253 Programmable Timer. The appropriate part numbers for the serial port options selected are as follows:

1) Serial I/O Kit 301-0072-002
2) Aux Serial I/O Kit 301-0072-004

The parallel I/O interface (Parallel I/O Kit 301-0072-003) (three, 8-bit I/O ports or two, 8 bit I/O ports with handshaking capabilities) provides a means of connecting the MSC 8007 to peripheral equipment that is designed to send and receive parallel data; in this particular case, the glide slope angle select switch. The 8255 Interface Controller receives and transmits the information as the 24 parallel lines of J1 (50 pin edge connector). There are three modes of operation available for the 8255. For this particular design, Mode 0 was selected which provides three separate 8-bit ports.

The MSC 8301 Uniform Monitor (Appendix E) was selected for the system and installed at location U101. The Uniform Monitor is a ROM resident stand alone I/O monitor. The program resides in 2K bytes of ROM and provides program loading, memory modification and similar basic functions. A second EPROM was also installed, containing the program developed for this system.

MODIFICATIONS PERFORMED TO MSC 8007 COMPUTER BOARD:

1) Jumper 117-118 is not implemented on this board. Its effect was duplicated by connecting Pin 9 to Pin 7 (ground) of the 74LS04 chip at position U5

2) Backplane pin 32 is connected to pin 8 of the socket for the MC1489 chip at location U9B. This connects the backplane pin 32 directly to the RXD input (pin 3) of the 8251 UART at position U19B. The 9th pin on the 1489 chip is removed from the socket.*

3) The optional Parallel I/O kit, Optional Serial I/O kit, and ontional Auxillary Serial I/O kit have been installed.

/NOTE/ *For test purposes, the chip is installed with the 9th pin in place.

4) A second EPROM, containing the program developed for this system, is installed at position U102.

5) Jumper between pins 1 and 14 on the DIP header at position U6 connects output 0 of the CTC chip to the clock input of the aux. Serial port 1.

Jumper Selected Options:

With reference to Appendices A and B of the "User's Manual for MSC 8007", and the schematics in that manual, the following jumper options have been implemented. Where the option is a normal factory option, a (F) symbol appears. Where a factory option has been disconnected, a (F) symbol appears. The term N.C. means not connected.

Pin 1 to Pin 72 Pin 72 N.C. Pin 73 (F)	Disconnects the Backplane INT7/ from R0 of the interrupt controller, and connects Timer output 1 (implements real time clock)
Pin 4B to Pin 74 Pin 74 N.C. Pin 75 (F)	Disconnects the Backplane INT6/ from R2 of the interrupt controller, and connects RXRDY/ from Aux UART 1 (implements interrupt driven Loran-C data receiver)
Pin 11 to Pin 12	Implements U2 buffer chip to parallel port B as input mode. Allows use of port B to select Nominal Approach Speed.
Pin 18 to Pin 19	Implements U4 buffer chip to parallel port A as input mode. Allows use of port A for digiswitch selection of approach angle.
Pin 24 to Pin 27 Pin 24 to Pin 28	Selects 2 MHz clock line as driver for clocks 1 and 0 (respectively) of 8253 CTC.
Pin 29 to Pin 25 Pin 29 to Pin 26	Grounds the gates for clocks 0 and 1 (respectively) of the 8253 CTC (allows free-running operation)

Pin 40A to Pin 42A	Selects on-board clock for the Optional Serial port
Pin 40B to Pin 42B	On board clock, Aux Serial Port 1.
Pin 41B to Pin 42B	Implemented for test purposes.
Pin 43A to Pin 44A Pin 46A to Pin 47A	Connects ±12 volts to the 75188 driver chip at U10A
Pin 43B to Pin 44B Pin 46B to Pin 47B	Connects ±12 volts to the 75188 driver chip at U10B
Pin 50 to Pin 51(F)	Supplies 2 MHz to CTC
Pin 54 to Pin 55(F)	Not Used
Pin 58 to Pin 59(F)	Supplies 4 MHz to Z80
Pin 60 to Pin 61(F)	Connects Reset to Bus Reset
Pin 62 to Pin 63(F)	Connects 8 MHz to Bus
Pin 64 to Pin 65(F)	Implements Watchdog Timer
Pin 68 to Pin 69(F)	Connects 8 MHz to Bus
Pin 70 to Pin 71(F)	Connects NMI to Bus NMI
Pin 76 to Pin 77(F) thru Pin 86 to Pin 87(F)	These connect backplane lines INT5/ through INT0/ to Interrupt controller lines R2 through R7, respectively
Pins 88 through 95 Factory Set	
Pins 96A to Pin 97A	Implements Serial Port DSR line
Pin 98A to Pin 99A	Makes TXC=RXC
Pin 96B to Pin 97B	As above for optional Aux
Pin 98B to Pin 99B	Serial Port 1
Pins 100 through 116 Factory Set	

2.4 A/D - D/A MICROCOMPUTER SYSTEM

The A/D - D/A Microcomputer-compatible system used was a Burr-Brown model MP8418. This particular analog I/O system is electrically and mechanically compatible with and interfaced directly to Intel's Multibus and other microcomputers of similar configuration.[4]

The analog input portion of the MP8148 includes: an analog multiplexer, resistor-programmed instrumentation amplifier or, a software-programmable amplifier (gain of 1 to 1024); sample/hold amplifier and a 12-bit A/D converter. An optional analog output system is included on the same board. It consists of two 12-bit D/A converters. The MP8418 is a 15-channel, differential or a 31-channel single-ended analog input system.

Table 2.2 presents the A/D - D/A board jumper configuration. Listed are the preset (factory) jumper configurations and the configuration as required to make the A/D - D/A board system compatible with the 3D Loran-C navigation system. Essentially only two areas require a modification from the factory configuration; these are the address jumpers and the analog output jumpers. Concerning the address jumpers, it is only necessary to change the sense of the bit on ADR F. This is accomplished by removing jumper 57 and installing jumper 65.

The Burr-Brown analog board is normally factory set to provide a ±10V output on each digital-analog controller (DAC) with two's complement coding (jumpers 23,25,19,28 and 46). As Table 2.2 indicates, certain modifications were made so that an output of 0V to +5V was available at both the DAC1 and DAC2. In addition, it was necessary to change from two's complement coding to straight binary. All of this was accomplished by removing jumpers 23, 25, 19, 28, 46 and replacing them with jumpers 22, 24, 26, 18, 20, 27 and 47. Also, the gain and the offset had to be adjusted for DAC2 so that the proper voltage tolerance could be obtained.

Appendix F presents the factory manual for the Burr-Brown A/D - D/A Converter. Any additional information or clarification necessary can be obtained from Appendix F.

Table 2.2 A/D - D/A Microcomputer Jumper Configuration

	Factory	Present Configuration	Modification
Address Jumpers	34	34	
	32	32	
	35	35	
	50	50	
	51	51	
	52	52	
	61	61	
	54	54	
	55	55	
	56	56	
	57	-	Remove
ADRF		65	Add
Analog Output ±10V DAC1	23	-	Remove
	25	-	Remove
DAC2	19	-	Remove
	28	-	Remove
2's complement code	46	-	Remove
Output & Feedback Lines DAC1	79	79	
DAC2	21	21	
0 to +5V DAC1		22	Add
		24	Add
		26	Add
DAC2		18	Add
		20	Add
		27	Add
Straight Binary		47	Add
A/D Converter Range Setting ±10V	12	12	
	14	14	
2's complement mode	49	49	
	69	69	

Table 2.2 A/D - D/A Microcomputer Jumper Configuration (continued)

	Factory	Present Configuration	Modification
15 Channel Differential	3 4 7 42	3 4 7 42	
Grounds CH0 Input	80	80	
Halt Mode	17 29 30 81	17 29 30 81	
Interrupt Vector (Int.1)	71	71	
Optional Expander Board	39 43	39 43	

2.5 POWER SUPPLY

The power supply used for the 3D Loran C navigation system was a DATEL INTERSIL, PCT-12/2 - 5/6. The system utilizes an input line voltage of 115 VAC and has output voltages of ±12 VDC and +5 VDC. The specifications for the DATEL power supply are as follows:

Input Voltage*	115/230VAC±10%
Line Frequency	48-440 Hz
Output Voltage Adjustment	±10%
Output Ripple	2mV RMS, max.
Transient Response	50μsec. max.
Output Protection	Current Limiting or Foldback Limiting
Isolation Resistance	100 Meg. min.
Voltage Stability, after warmup	±0.25%, 24 hours
Isolation Capacitance	250pF max.
Breakdown Voltage	1500VAC min.
Operating Temperature Range	0°C to 50°C (No Derating)
Storage Temperature Range**	-25°C to +85°C
Output Voltage	±12V @ 2A/5V @ 6A
Chassis Size (in.)	11.25 x 4.87 x 2.75
Weight (lbs)	5.8

One minor modification had to be made so that a -5V power source could be obtained from the DATEL power supply. By utilizing a -5V regulator and two polarized filter capacitors, the task was

/NOTE/ *Input voltage is selected by transformer connection
**Derated to 40% of output current at 71°C.

accomplished easily. A solder terminal strip was mounted directly to the DATEL chassis so that all of the necessary connections could be made. Figure 2.4 shows the wiring schematic and general assembly for the -5V power source.

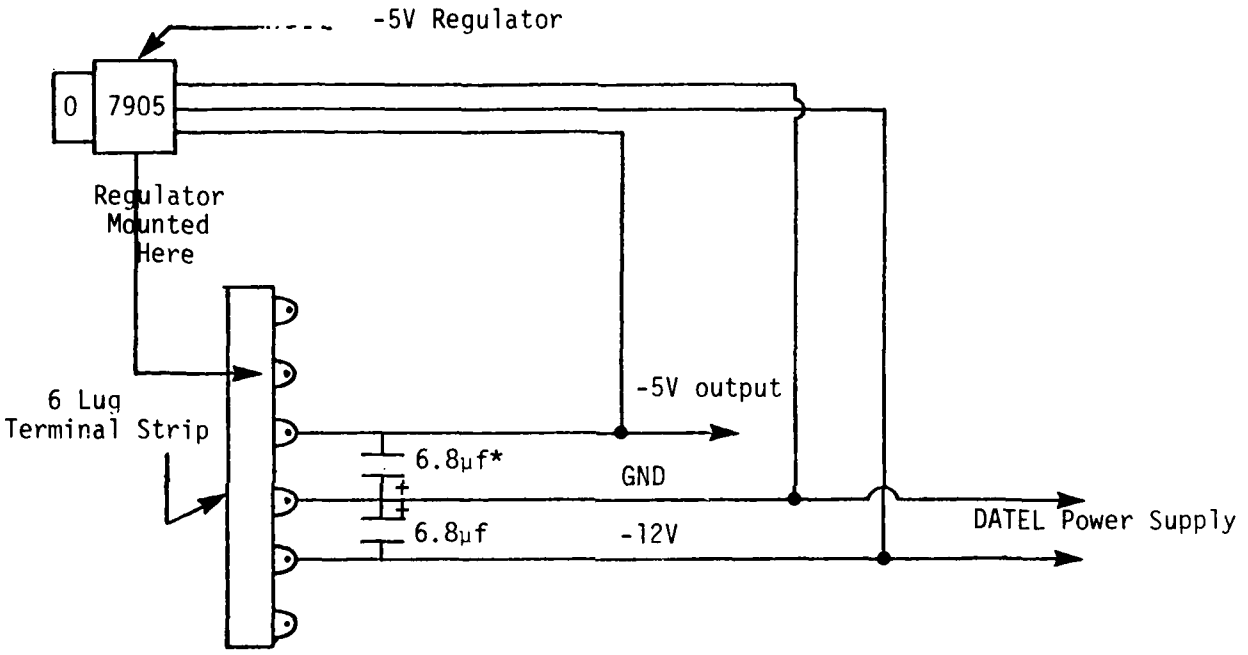


Figure 2.4 -5V Power Source

All of the wires from the power supply terminate into a 6 pin female MOLEX connector. Figure 2.2 of the Multibus chassis section presents the pin configuration for power connector. The 115VAC input is wired to the transformer on the power supply as per the instructions.

/NOTE/ * ±20% tolerance, tantalum capacitors

2.6 3D LORAN-C CHASSIS

Description:

The 3D Loran-C chassis is composed entirely of the components described previously in Sections 2.1 through 2.5. All of the components are securely fastened to a piece of .080 aluminum plate that measures 17.75 inches by 14.75 inches. All of the components are fastened to the plate using standard 10-32 aircraft bolts with elastic-stop nuts. Aircraft mounting is accomplished by utilizing a standard 19 inch electronics rack. This is easily accomplished since the Multibus chassis is a standard 19 inch RETMA rack mount. Figure 2.5 shows a bottom view of the 3D Loran-C chassis. As shown in the figure, the chassis itself is composed of four major components. They are: the Multibus chassis, the DATEL power supply, barrier strip and the power connector with fuse.

Chassis Power:

The 3D Loran-C chassis operates on 115 VAC power, either at 60 Hz or 400 Hz. The power connection is made at the chassis using a male 3 pin Cannon plug (MS 3057-6A), rated at six (6) amperes. Figure 2.6 shows the power schematic for the 3D Loran-C chassis. 20 AWG wire was more than sufficient for this installation.

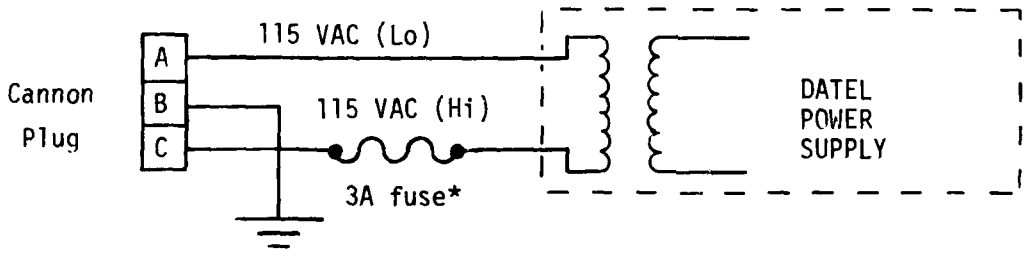


Figure 2.6 3D Loran-C Power Schematic

/NOTE/ *Standard chassis mount 3A fuse.

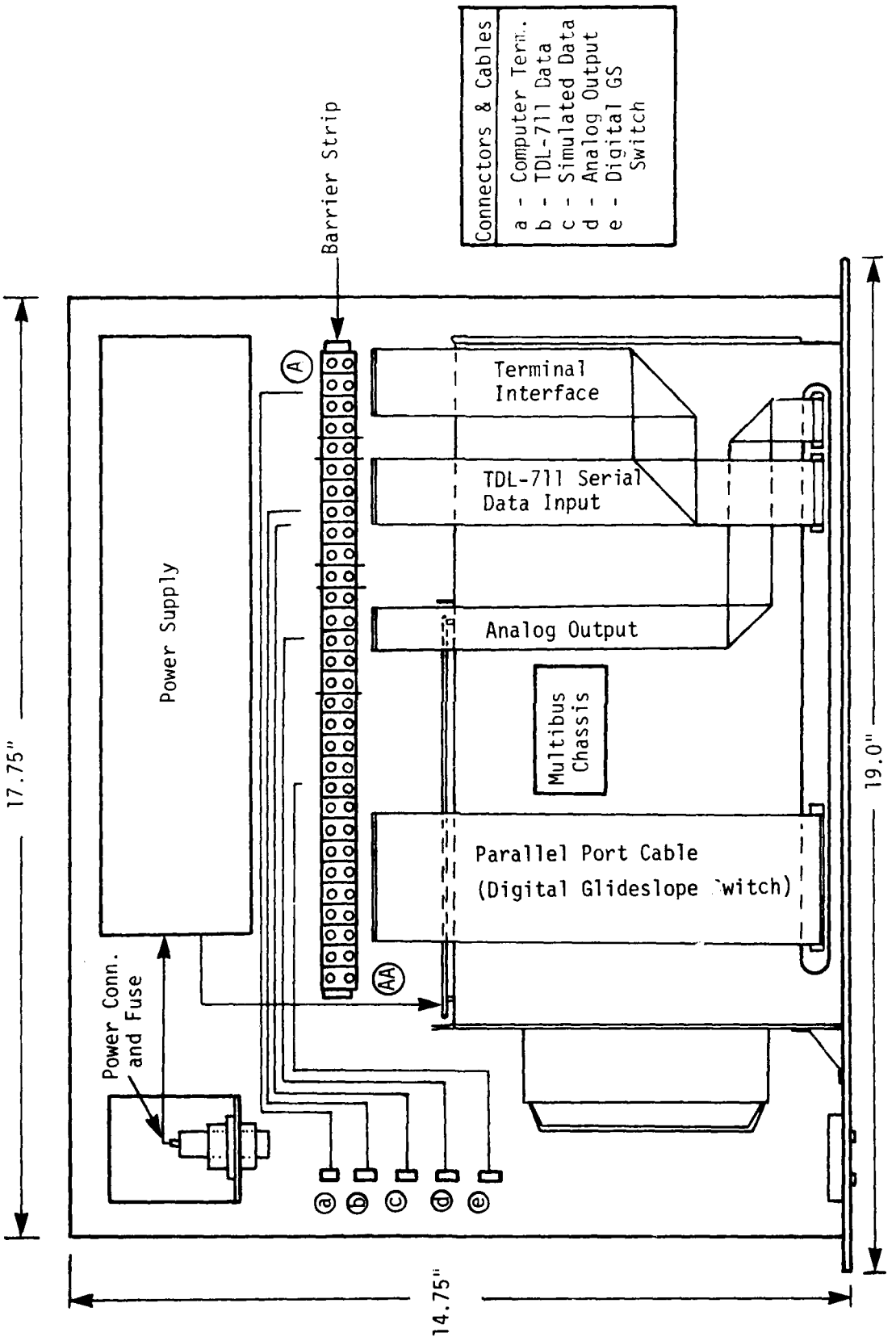


Figure 2.5 3D Loran-C Chassis

Chassis Interconnection:

Figure 2.7 presents a detailed wiring schematic for the 3D Loran-C chassis. A 30 lug barrier strip was utilized to make interconnection between peripheral devices and chassis components. Interconnection between the wirewrap board (MSC 8204), in addition to serial port 1 on the processor board, was made using a 26 conductor ribbon cable. Similarly, a 50 conductor ribbon cable was utilized for the interconnection between the processor board's parallel port and the barrier strip. Likewise, the A/D - D/A converter required a 20 conductor ribbon cable to make the appropriate connections. All of these ribbon cables are securely fastened to the Multibus chassis using two bolts and a flat aluminum plate. Those connections indicated in the dashed boxes are for testing and prototype development purposes only. For normal operation, these connections would not be made. Most of the connections made on the peripheral side of the barrier strip are made with 4 and 12 pin Molex connectors (see Figure 2.8 for the appropriate part numbers). Wiring for these connections was accomplished using standard 24 AWG wire.

As shown in the lower lefthand corner of Figure 2.7, a small modification was necessary to supply the appropriate flag voltage to the VNAV Alert unit. The circuit is a very simple transistor-resistor network utilizing a standard NPN Silicon (2N2484) transistor. The circuit is activated through the A/D board output signal [0V (Hi), 2V (Lo)]. When activated, 18VDC is transmitted to the VNAV Alert unit.

Test/Operational Configuration:

Several items on the 3D Loran-C chassis must be either connected or disconnected to make the transition from test to operational configuration. As mentioned earlier, on Figure 2.7 those items utilized for testing purposes only are indicated by the dashed boxes. Basically, to operate the 3D Loran-C in the test mode, the following operations must be completed.

1) Connect pins 8 & 9 on the processor board parallel port connector. (A plug is provided on the ribbon cable for this purpose.) This selects 4800 baud rather than 20833 baud used for the TDL-711.

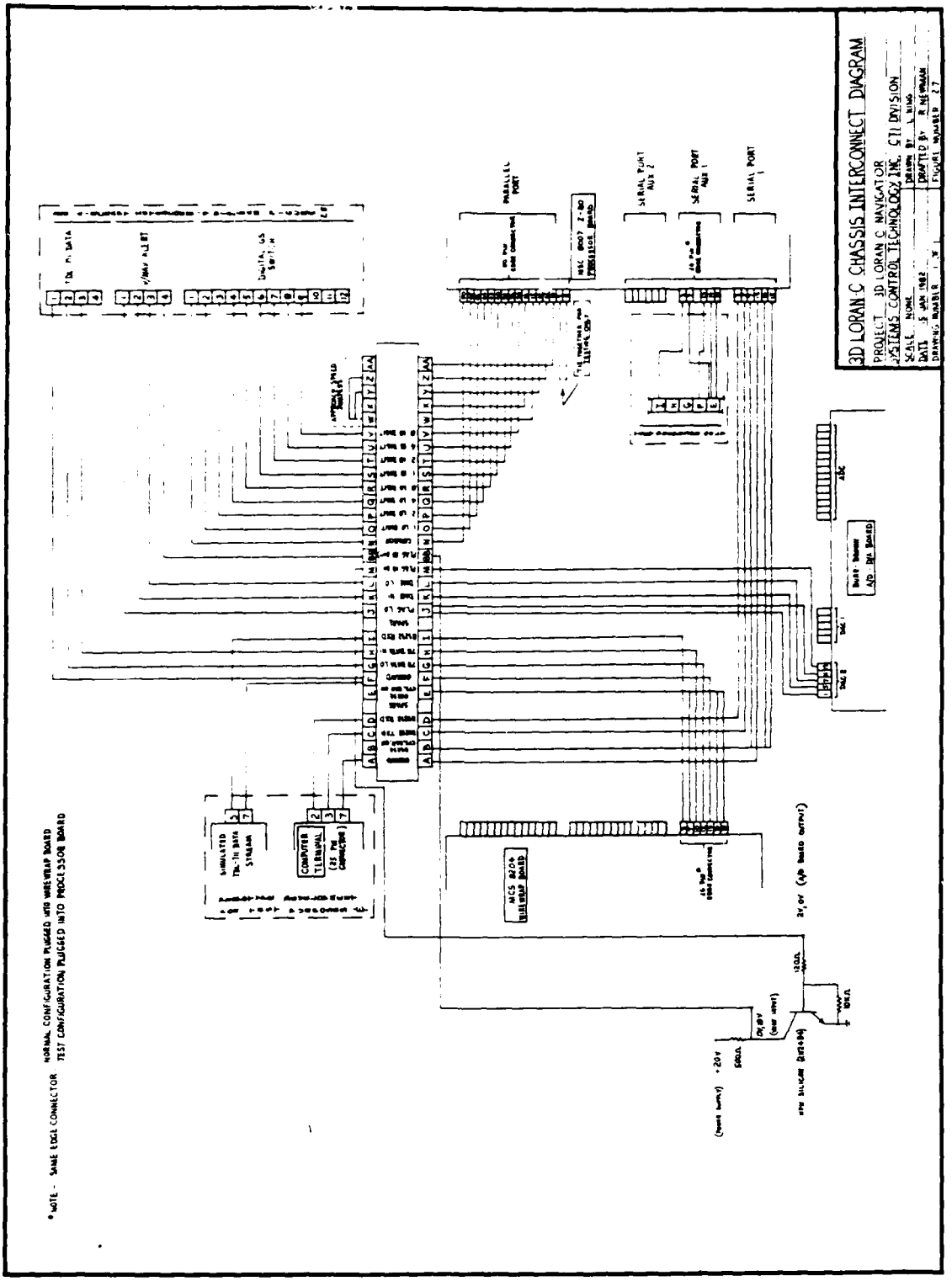


Figure 2.7 3D Loran-C Chassis Instrument Diagram

2) Remove the wirewrap board and connect the 26 pin edge connector normally plugged into the wirewrap board into aux serial port 1. (Plug J3)

3) Connect the computer terminal plug and the simulated TDL-711 data stream plug to their respective devices.

4) Insert the 9th pin on the MC1489 chip at location U9B on the processor board. (Substitute a normal 1489 chip)

To operate the 3D Loran-C navigator in its normal mode of operation, the following conditions must be checked.

1) Disconnect pin 8 & 9 on the processor board parallel port.

2) Insert the wirewrap board, and plug in the Loran data stream 26 pin edge connector.

3) Disconnect and secure all external computer plugs.

4) Remove the 9th pin on the MC1489 chip at location U9B on the processor board.

2.7 AIRCRAFT INTERCONNECTION

The 3D Loran-C navigator unit is connected to three major components in the aircraft. They are the TDL-711 RPU, Intercontinental Dynamics V/NAV ALERT (541-24912-112) unit and the digital GS switch (AMP 8012). The 3D Loran-C receives distance to the waypoint information from the TDL-711 RPU where it is in turn filtered, smoothed and converted to an analog output signal. This analog output signal is transmitted directly to the V/NAV ALERT unit. The 3D Loran-C navigator itself operates on 115 VAC power supplied by the static inverter onboard the aircraft. Glide path descent angle information is supplied to the 3D Loran-C navigator by a digital switch mounted on the aircraft's instrument panel. Figure 2.8 shows the 3D Loran-C navigator interconnection diagram.

Figure 2.9 shows the interconnection between the V/NAV ALERT unit, the altimeter and the glideslope indicator. The diagram is self-explanatory.

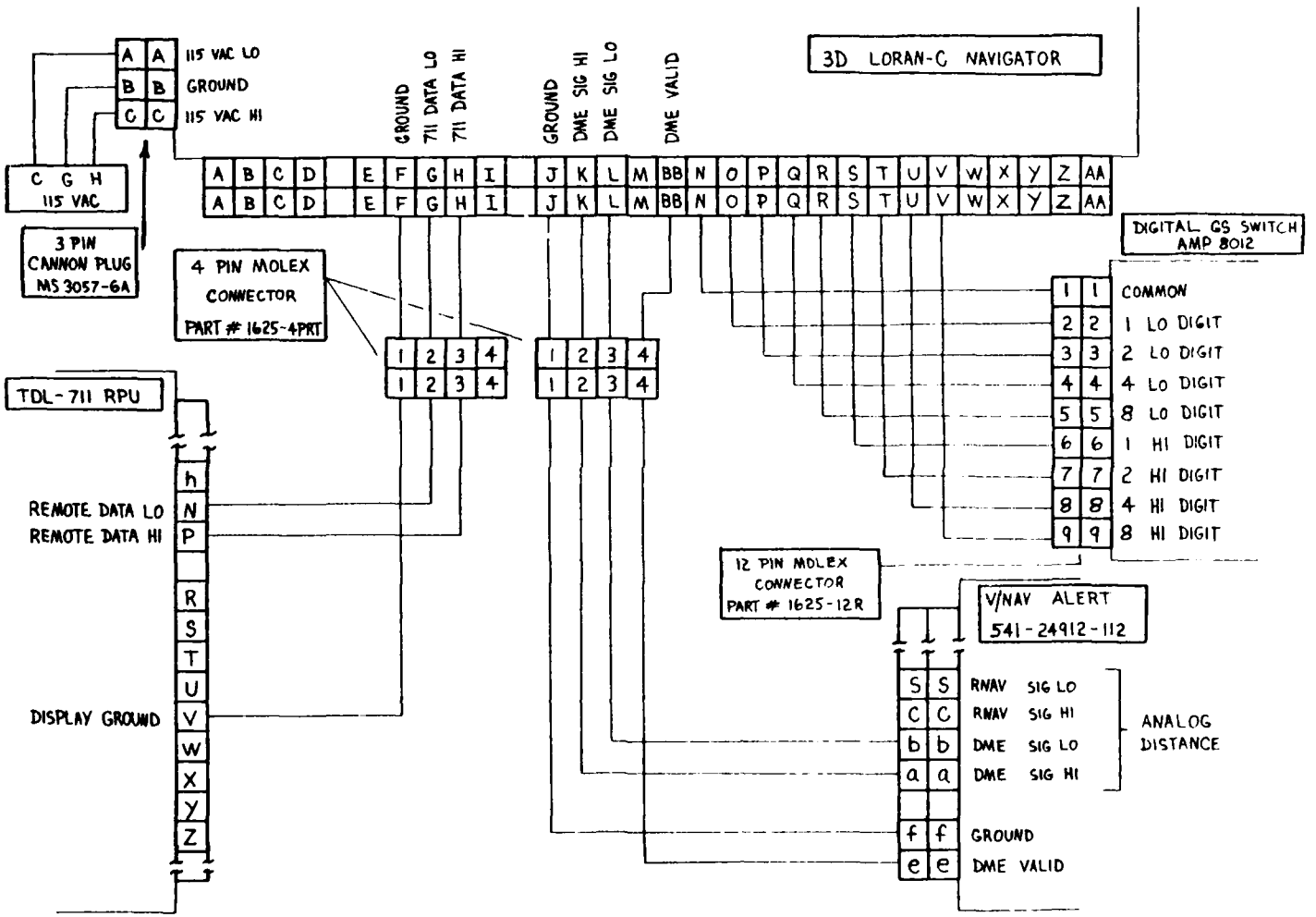


Figure 2.8 3D Loran-C Navigator Aircraft Interconnect Diagram

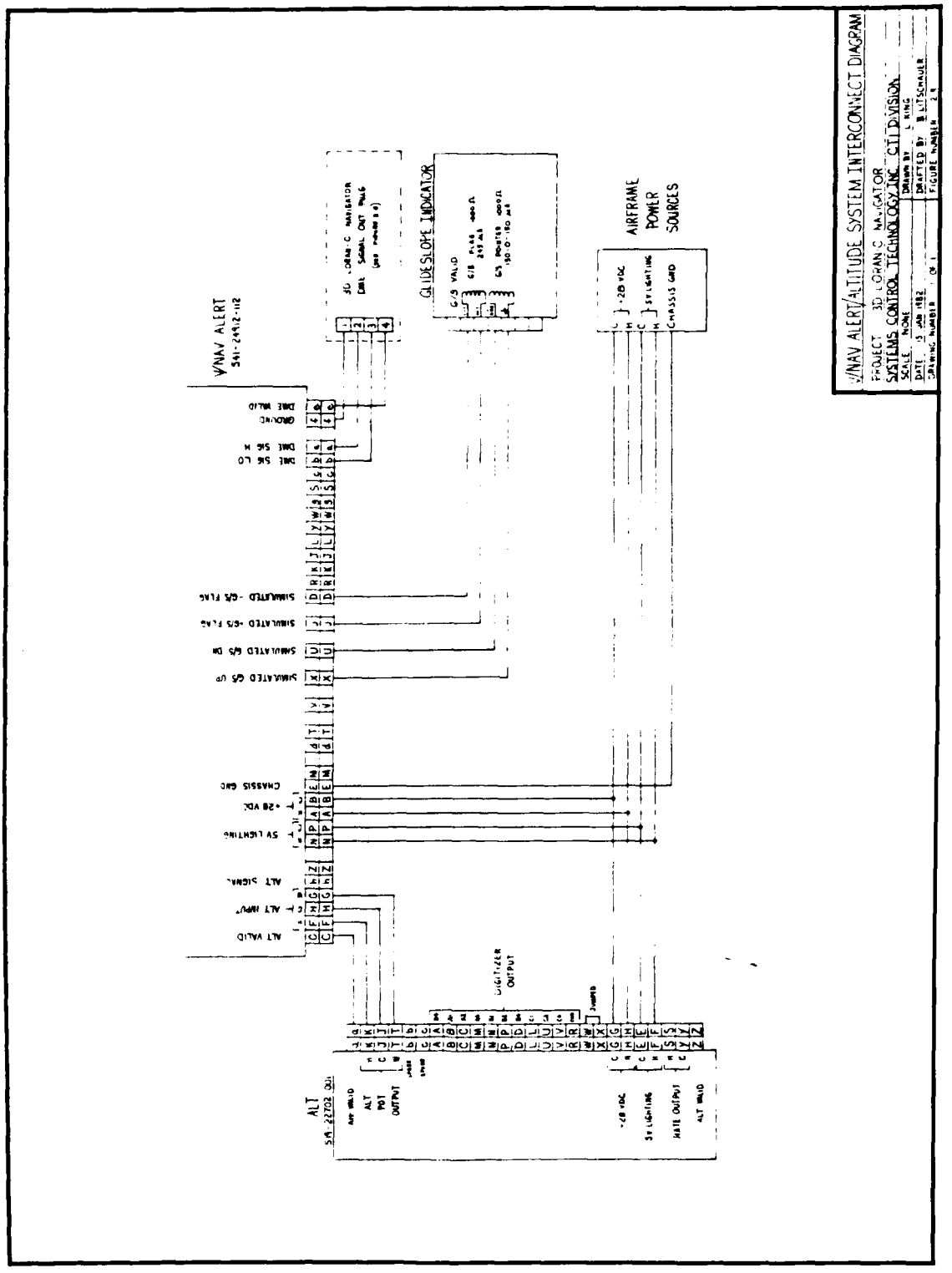


Figure 2.9 VNAV Alert/Altitude System Interconnect Diagram

REFERENCES

1. Anonymous, "MSC 8201 Maintenance Manual Multibus Chassis", Monolithic Systems Corp., 1978.

2. Anonymous, "TDL-711 Interface CICTS", Sierra Nevada Corp., (Circuit Diagram), January 8, 1981.

3. Anonymous, "User's Manual For MSC 8007", Monolithic Systems Corp., 1980.

4. Anonymous, "Microcomputer Analog I/O Systems", MP8418, Burr-Brown Research Corporation, 1978.

5. T.E. Scalise, E.H. Bolz, and E.D. McConkey, "West Coast Loran-C Flight Test", Systems Control, Inc. (Vt.), Champlain Technology Industries Division for the Federal Aviation Administration Systems Research and Development Service, FAA-RD-80-28, March 1980.

APPENDICES

APPENDIX A -- "MSC 8201 Maintenance Manual Multibus Chassis", Monolithic Systems Corp., 1978. (Included by Reference).

APPENDIX B -- "Maintenance Manual For Universal Card MSC 8204", Monolithic Systems Corp., 1978. (Included by Reference).

APPENDIX C -- "DS7820A/DS8820A Dual Line Receiver", National Semiconductor Corp. Also "TDL-711 Interface CICTS", Sierra Nevada Corp., (Circuit Diagram), January 8, 1981. (Included by Reference).

APPENDIX D -- "User's Manual For MSC 8007", Monolithic Systems Corp., 1980. (Included by Reference).

APPENDIX E -- "Reference Manual For Uniform Monitor Program MSC 8301", Monolithic Systems Corp., 1978. (Included by Reference).

APPENDIX F -- "Microcomputer Analog I/O Systems MP 8418", Burr-Brown Research Corporation, 1978. (Included by Reference).

SOURCES

Monolithic Systems Corp.
84 Inverness Circle East
Englewood, Colorado 80112
(800)525-7661

National Semiconductor Corp.
2900 Semiconductor Dr.
Santa Clara, California 95051
(408)732-5000

Sierra Nevada Corp.
P.O. Box 06190
Stead Station
Reno, Nevada 89506

Burr-Brown Research Corp.
Box 11400
Tucson, Arizona 85734
(602)746-1111


ATE
LME