

# DN11

DN11 DIALEX  
MD-11-DZDNA-B

EP-DZDNA-B-DL-B  
COPYRIGHT © 76-77  
FICHE 1 OF 1

DEC 1977  
**digital**  
MADE IN USA

This microfiche grid contains 240 frames (12 columns by 20 rows). The frames contain various data, including text and barcodes. The text is mostly illegible due to the low resolution of the scan. The barcodes are vertical and appear to be standard microfiche identification codes.



.REM %

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDNA-B-D  
PRODUCT NAME: DN11 DIALEX  
DATE : AUGUST, 1977  
MAINTAINER: DIAGNOSTIC GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1976, 1977 BY DIGITAL EQUIPMEN CORPORATION

## 1. ABSTRACT

THE DN11 DIAGNOSTIC CONSISTS OF TWO PARTS. THE FIRST IS A SERIES OF INCREMENTAL TESTS WHICH STATICLY CHECK OUT THE DN11 USING THE MAINTENANCE MODE. THE SECOND PART IS THE ON LINE EXERCISER WHICH ALLOWS THE USER TO DIAL ANY GIVEN PHONE IN HIS DIALING RANGE. UPON THE COMPLETION OF THE CALL THE PROGRAM WILL TERMINATE THE CALL AND TRY AGAIN.

## 2. REQUIREMENTS

## 2.1 EQUIPMENT

PDP-11 (MIN.4K)-WITH OR WITHOUT A HARDWARE SWITCH REGISTER  
TELETYPE  
DN11 (MAX.OF4 USED AT ONE ANY TIME)

## 2.2 STORAGE

DIALEX OCCUPIES THE FIRST 4K OF CORE.

## 3. LOADING PROCEDURE

## 3.1 METHOD OF LOADING DIALEX TAPE

PROGRAM FORMAT ABSOLUTE

- A. VERIFY THE BOOT LOADER IS IN MEMORY  
B. SET SWITCH REGISTER EQUAL TO #500

MEMORY	SIZE *
4K	17
8K	37
12K	57
16K	77
20K	117
24K	137
28K	157

- C. DEPRESS LOAD ADDRESS  
D. DEPRESS START

4. STARTING PROCEDURE

- A. LOAD ADDRESS 200.
- B. SET SWITCH REGISTER CORRESPONDING TO SEC 5.2  
-SEE D. FOR SOFTWARE SWITCH REGISTER LOADING-
- C. DEPRESS START.
- D. IF THE SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING  
WILL BE TYPED:  
SWR=XXXXXX NEW= (REFER TO SECTION 5.2 FOR OPTIONS)

4.1 SCOPE LOOP STARTING PROCEDURE

- A. LOAD ADDRESS 204.
- B. SET THE SWITCH REGISTER EQUAL TO THE ADDRESS  
OF THE DN11.  
\*\*\*WHEN SOFTWARE SWITCH REGISTER IS SELECTED THE  
OPERATOR WILL BE ABLE TO LOAD THE DN11 ADDRESS AFTER DEPRESSING START.
- C. DEPRESS START.
- D. SET SWITCH TO CORRESPOND TO SEC. 5.3
- E. DEPRESS CONTINUE.  
\*\*\*IF THE SOFTWARE SWITCH REGISTER IS USED DEPRESS CONTINUE  
THE MACHINE WILL THEN ASK FOR SOFTWARE SWITCH REGISTER CHANGE  
BY TYPING THE FOLLOWING: SWR=XXXXXX NEW= (REFER TO SECTION  
5.2 FOR OPERATOR OPTIONS)\*\*\*

4.2 RESTARTING AT LOC. 200

RESTARTING AT LOC. 200 WILL AUTOMATICALLY USE THE ADDRESS  
AND VECTOR ENTERED AT THE INITIAL START-UP.  
IF IT IS DESIRED, TO ENTER A NEW ADDRESS UPON RESTART,  
CLEAR LOCATION 1064 AND START AT LOC. 200.



## 5. OPERATING PROCEDURE

AT THE INITIAL START OF THE PROGRAM THE OPERATOR WILL BE ASKED FOR THE ADDRESS OF THE FIRST DN11, AND ITS VECTOR ASSIGNMENT.

\*\*\*\*IF SOFTWARE SWITCH REGISTER IS SELECTED THEN THE FOLLOWING WILL BE TYPED FIRST:  
SWR=XXXXXX NEW= (REFER TO SECTION 5.2 FOR OPTIONS)

DN11 REGISTER ADDRESS?XXXXXX

VECTOR ASSIGNMENT?XXX

## 5.1 DIALING PROCEDURES

THE OPERATOR WILL BE ASKED FOR A PHONE NUMBER FOR EACH DN11, IN THE FOLLOWING MANNER:  
WHEN THE MAXIMUM NUMBER OF DN11'S IS REACHED FOR THE SYSTEM THE OPERATOR MUST DEPRESS THE CARRIAGE-RETURN KEY WITHOUT DEPRESSING ANY OTHER CHARACTER.

PHONE #1? XXXXX

PHONE #2? XXXXXX

PHONE #3? XXXXXX

PHONE #4? XXXXXX

NOTE: DO NOT TYPE <IG> DURING THE INPUTING OF PHONE NUMBERS OR ERROR WILL OCCUR.



## 5.2 CONTROL SWITCH SETTINGS

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED.

## CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G (<?G>): THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.
- 2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE ''NEW='' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
  - A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)  
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.
  - B) IF A CONTROL U (<?U>) IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

NOTE: DUE TO THE USE OF RESET INSTRUCTION IT MAY BE NECESSARY TO DEPRESS <?G> MORE THAN ONCE. THIS IS CAUSED BY THE RESET INSTRUCTION NOT ALLOWING THE LOADING OF THE TTY RECEIVER BUFFER DURING THE RESET EXECUTION.



GO1

SR BIT15 SET=HALT ON ERROR  
SR BIT15 RESET=CONTINUE AFTER REPORTING ERROR

SR BIT14 SET=LOOP ON STATIC TEST SUB-SET  
SR BIT14 RESET=DO EACH STATIC TEST SUB-SET 15 TIMES.

SR BIT13 SET=DELET TYPE-OUT  
SR BIT13 RESET=REPORT EACH ERROR

SR BIT12 SET=TERMINATE CALL BY LOWERING CRQ (CALL REQUEST)  
SR BIT12 RESET=TERMINATE CALL BY ISSUING RESET

BIT11 SET=801 NEEDS EON TO COMPLETE CALL  
BIT11 RESET=EON NOT NEEDED TO COMPLETE CALL

SR BIT10 SET=LOOP ON ON-LINE TEST  
SR BIT10 RESET=SEQUENCE THROUGH PROGRAM

SR BIT9 SET=LOOP ON ALL STATIC TESTS  
SR BIT9 RESET=SEQUENCE THROUGH PROGRAM

SR BIT8 SET=RUN STATIC TEST ON DN11 SELECTED BY SRQ-1  
SR BIT8 RESET=PROGRAM WILL SEQUENCE THROUGH ALL DN11'S

SR BIT7 SET=DELETE TTY CONVERSATION FOR DIALING SEQUENCE  
SR BIT7 RESET=ENTER TTY CONVERSATION FOR DIALING SEQUENCE

SR        1        0        =SELECT DN11 FOR STATIC TEST  
          RESET   RESET=FIRST DN11  
          RESET   SET   =SECOND DN11  
          SET     RESET=THIRD DN11  
          SET     SET   =FOURTH DN11



5.3 SCOPE LOOP SWITCH SELECTION

IN THE SCOPE LOOP THE USER MAY SET ANY OR ALL OF THE DN11 STATUS BITS IN THE MAINTENANCE OR DYNAMIC MODE. IF THE USER SETS THE BITS IN THE DYNAMIC MODE THE PROGRAM WILL AUTOMATICALLY STICK IN THE CORRECT TIME DELAYS FOR THE PHONE LINE.  
\*\*\*\*\*REFER TO SECTIONS 4.1 AND 5.2 FOR SOFTWARE SWITCH REGISTER OPERATION\*\*\*\*\*

THE DETAILED DESCRIPTION OF DN11 STATUS BITS

BIT	NAME	DESCRIPTION
00	CALL REQUEST (FCRQ)	CONTROL LEAD TO ACU. THIS BIT STARTS THE AUTOMATIC CALLING SEQUENCE. (WRITE ONLY)
01	DIGIT PRESENT (FDPR)	CONTROL LEAD TO THE ACU. THIS BIT MUST BE SET BY THE PROGRAM AFTER IT LOADS THE NEXT DIGIT (IN RESPONSE TO A PND REQUEST) TO INFORM THE ACU TO CONTINUE WITH DIALING. THE INTERFACE AUTOMATICALLY CLEARS THIS BIT WHEN THE ACU CLEARS PND TO INDICATE ACCEPTANCE OF THE DIGIT. (READ/WRITE)
02	MASTER ENABLE (MINAB)	ALLOWS THE PROGRAM TO DISABLE THEN REENABLE ALL 4 ACU INTERRUPTS EASILY WITH ONE BIT. THIS BIT IS CONNECTED FOR ONLY ONE OF THE FOUR POSSIBLE LINES WHICH MOUNT IN ONE SYSTEM UNIT. (READ/WRITE)
03	MAINTENANCE (MAINT)	THIS BIT, WHEN SET, ALLOWS CHECKING OF THE INTERFACE WITHOUT A CONNECTED ACU. IT ALLOWS FCRQ TO BE READ AND SWITCHES THE ACU RESPONSE LINES-- PND, DSS, PWI AND ACR TO THE OUTPUT OF THE DIGIT LINES FOR TESTING PURPOSES. BIT DIGIT ACU LINE CTL BIT # 08 NB1 PND FPND 04 09 NB2 DSS FDSS 05 10 NB4 PWI PWO 13 11 NB8 ACR FACR 14
04	PRESENT NEXT DIGIT (FPND)	ALSO FORCES CRQ (TO ACU) OFF AND FORCES FDLD (BIT 12) ON. (READ/WRITE) CONTROL LEAD FROM THE ACU. THIS IS A REQUEST BY THE ACU FOR THE PROGRAM TO LOAD ANOTHER DIGIT DURING DIALING. IT IS ACCOMPANIED BY THE SETTING OF DONE TO OBTAIN AN INTERRUPT. IT IS CLEARED BY THE ACU WHEN THE DIGIT IS ACCEPTED (AFTER DPR IS SET) AND WILL

MD-11-DZDNA-B, DN11 DIALEX  
DZDNAB.P11 01-SEP-77 09:31

MACY11 30(1046) 01-SEP-77 09:57 PAGE 8

IO1

SEQ 0008

REMAIN OFF AT LEAST 600 MS BEFORE  
COMING UP FOR THE NEXT REQUEST.  
(READ ONLY)



05 DATA SET STATUS (FDSS) CONTROL LEAD FROM ACU. THIS IS A STATEMENT BY THE ACU THAT THE CALLED PARTY HAS ANSWERED AND THAT THE ASSOCIATED DATA SET NOW HAS CONTROL OF THE LINE. IT IS ACCOMPANIED BY THE SETTING OF DONE TO OBTAIN AN INTERRUPT. IT REMAINS SET UNTIL AFTER THE END OF THE CALL. (OR UNTIL THE DATA TERMINAL READY LEAD TO THE ASSOCIATED MODEM IS DROPPED WHICH THEN DROPS FDSS).

IF THE ASSOCIATED MODEM ANSWERS A CALL WHILE THE DIALER IS IN USE (CRQ=1) THEN DSS WILL BE ENABLED AND DONE SET. IF INTERRUPT ENABLE IS SET THERE WILL BE AN INTERRUPT. (READ ONLY)

06 INTERRUPT ENABLE (INTENB) THIS BIT ALLOWS THE SETTING OF DONE TO CAUSE AN INTERRUPT IF THE MASTER ENABLE BIT (BIT 02 LINE #1 OF A SYSTEM UNIT) IS SET. (READ/WRITE)

07 DONE THIS BIT IS SET TO INDICATE THAT THE ACU IS DONE WITH THE PREVIOUSLY REQUESTED ACTION AND READY TO ACCEPT NEW DATA, USUALLY THE NEXT DIGIT IN A SEQUENCE TO BE DIALED. THE CONDITIONS THAT SET DONE ARE LISTED (CRQ MUST BE A ONE):

1. TRANS. OF PND TO ONE (AFTER LAST SET OR PREV. DPR SET)
2. TRANS. OF DSS TO ONE (AFTER LAST DPR OR EON)
3. TRANS. OF ACR TO ONE (IF TIMEOUT ERR--ANYTIME)
4. TRANS. OF PLO TO ONE (IF POWER SWITCHED OFF) (READ/WRITE)

08-11 DIGIT BITS (NB1-4) THESE FOUR BITS ARE CONTROL LEADS TO THE ACU. THESE LOW ORDER BITS OF THE SECOND BYTE MAKE UP THE BCD DIGIT TO BE DIALED. SINCE THE HIGH-ORDER FOUR ARE READ ONLY, IT DOESN'T MATTER WHAT IS IN THEM DURING A LOAD, AND THE PROGRAMMER MAY USE THEM AS HE WISHES. IN MAINT MODE, THESE BITS ARE USED TO THE FOUR CONTROL LINES THAT CAN CAUSE INTERRUPTS. SEE BIT 03 FOR DESCRIPTION. (READ/WRITE)

- 12 DATA LINE OCCUPIED (FDLO) THIS BIT IS SET BY THE ACU WHENEVER THE LINE TO THE TELEPHONE CENTRAL OFFICE IS BEING USED BY THE ACU. IT ALLOWS THE PROGRAMMER TO TEST THE ACU TO SEE IF THE LAST CALL WAS USUCCESSFULLY TERMINATED BEFORE HE TRIES TO USE IT FOR THE NEXT ONE. (READ ONLY)
- 13 NOT USED
- 14 ABANDON CALL AND RETRY (ACR) A CONTROL LEAD FROM THE ACU. THIS BIT IS SET BY THE ACU WHENEVER AN INTERNAL TIMER TIMES OUT. THE TIMER IS RESET BY THE ACU WHENEVER IT GIVES PBD AND IS FOR DETECTING WRONG NUMBERS AND BUSY SIGNALS. IT IS INHIBITED BY THE PRESENECE OF DSS EXCEPT IF THE 001 OPTION "Y" IS IN USE IN WHICH CASE IT TIMES OUT EVEN THEN AND GIVES AN INTERRUPT (BY SETTING DONE). THIS IS USED WHEN THE PROGRAMMER WANTS A TIMER TO DETECT WRONG NUMBERS AND BUSY SIGNALS.
- 15 POWER IN (PWI) THIS BIT IS NORMALLY ZERO AND IS SET BY THE ACU WHENEVER POWER IS SWITCHED OFF AT THE UNIT. IF A CALL IS IN PROGRESS AT THAT TIME, DONE IS SET. (THIS CAUSES AN INTERRUPT IF ITENB AND MINAB=1). (READ ONLY)



6.1 ERROR REPORTS

6.1.1 XXX ERROR COUNT

XXXXXX DN11

EQUAL TO THE ERROR TAG IN  
THE LISTING. THIS ENABLES  
THE USER TO FOLLOW THE  
EXACT CODE THAT FAILED.

DEFINES WHICH DN11 FAILED THE  
STATIC TESTS. THIS IS EQUAL  
TO THE ADDRESS ASSIGNMENT.

6.1.2 XXXXXX GD DATA

XXXXXX BD DATA

THIS EQUALS THE DATA  
LOADED INTO A REGISTER  
BY THE PROGRAM.

THIS EQUALS THE DATA READ FROM  
A REGISTER BY THE PROGRAM.

6.1.3 XXX ERROR COUNT

XXXXXX DNCSR

XXXXXX DN11

EQUAL TO THE ERROR  
TAG IN PROGRAM  
LISTING

CONTENTS OF DN11  
STATUS REGISTER  
AT THE TIME  
ERROR

DEFINES WHICH  
DN11 THAT FAILED

6.2 PROGRAM TIMED OUT UNABLE TO COMPLETE CALL

THIS MESSAGE IS REPORTED AFTER A PERIOD OF TIME HAS PASSED IN  
WHICH THE PROGRAM HAD EXPECTED TO HAVE RECEIVED DATA SET STATUS  
AND DID NOT.

6.2.1 THE 801 IS OFF LINE

THIS MESSAGE IS REPORTED AT THE START OF THE STATIC TEST  
WHENEVER THE DN11 IN USE HAS NO 801 DAILING UNIT CONNECTED TO IT.  
THE TESTS THAT DO NOT NEED AN 801 WILL BE EXECUTED.

6.3 POWER FAIL OCCURRED

THIS MESSAGE IS REPORTED IN THE RESTART SEQUENCE OF THE POWER  
FAIL ROUTINE. WHENEVER A POWER FAIL HAS OCCURRED THE PROGRAM  
TRAPS TO 24 AND RESETS THE VECTOR AND HALTS. ON THE RESTART  
SEQUENCE THE PROGRAM REPORTS THE MESSAGE AND WAITS TWO SECONDS  
FOR THE PHONE LINES TO SETTLE DOWN, THEN IT JUMPS TO THE START  
OF THE PROGRAM.

6.4 END

THIS MESSAGE IS REPORTED AT THE END OF EACH PASS OF THE PROGRAM:

7. TIME  
AMOUNT OF TIME TO RUN STATIC TEST 1.5 MIN.  
AMOUNT OF TIME TO RUN ON-LINE TEST 3 MIN.

8. RESTRICTIONS  
THE POWER FAIL CAPABILITY OF THIS DEVICE MUST ONLY BE  
PERFORMED IN THE ON-LINE TEST.

9. \*\*RECOVERING FROM ERROR HALTS WITH A SOFTWARE SWITCH REGISTER\*\*  
IF THE SOFTWARE SWITCH IS TO BE CHANGE AFTER A HALT  
THEN THE OPERATOR SHOULD DEPRESS A <↑G> BEFORE DEPRESSING  
THE CONTINUE SWITCH.

10. LISTING  
\*\*\*\*\*

⌘









94	00002	000226
95	00003	000000
96	00004	000232
97	00005	000000
98	00006	000000
99	00007	000000
00	00008	000000
01	00009	000000
02	00010	000000
03	00011	000000
04	00012	000000
05	00013	000000
06	00014	000000
07	00015	000000
08	00016	000000
09	00017	000000
10	00018	000000
11	00019	000000
12	00020	000000
13	00021	000000
14	00022	000000
15	00023	000000
16	00024	000000
17	00025	000000
18	00026	000000
19	00027	000000
20	00028	000000
21	00029	000000
22	00030	000000
23	00031	000000
24	00032	000000
25	00033	000000
26	00034	000000
27	00035	000000
28	00036	000000
29	00037	000000
30	00038	000000
31	00039	000000
32	00040	000000
33	00041	000000
34	00042	000000
35	00043	000000
36	00044	000000
37	00045	000000
38	00046	000000
39	00047	000000
40	00048	000000
41	00049	000000
42	00050	000000
43	00051	000000
44	00052	000000
45	00053	000000
46	00054	000000
47	00055	000000
48	00056	000000
49	00057	000000
50	00058	000000
51	00059	000000
52	00060	000000
53	00061	000000
54	00062	000000
55	00063	000000
56	00064	000000
57	00065	000000
58	00066	000000
59	00067	000000
60	00068	000000
61	00069	000000
62	00070	000000
63	00071	000000
64	00072	000000
65	00073	000000
66	00074	000000
67	00075	000000
68	00076	000000
69	00077	000000
70	00078	000000
71	00079	000000
72	00080	000000
73	00081	000000
74	00082	000000
75	00083	000000
76	00084	000000
77	00085	000000
78	00086	000000
79	00087	000000
80	00088	000000
81	00089	000000
82	00090	000000
83	00091	000000
84	00092	000000
85	00093	000000
86	00094	000000
87	00095	000000
88	00096	000000
89	00097	000000
90	00098	000000
91	00099	000000
92	00100	000000
93	00101	000000
94	00102	000000
95	00103	000000
96	00104	000000
97	00105	000000
98	00106	000000
99	00107	000000

94	HALT
95	HALT
96	HALT
97	HALT
98	HALT
99	HALT
00	HALT
01	HALT
02	HALT
03	HALT
04	HALT
05	HALT
06	HALT
07	HALT
08	HALT
09	HALT
10	HALT
11	HALT
12	HALT
13	HALT
14	HALT
15	HALT
16	HALT
17	HALT
18	HALT
19	HALT
20	HALT
21	HALT
22	HALT
23	HALT
24	HALT
25	HALT
26	HALT
27	HALT
28	HALT
29	HALT
30	HALT
31	HALT
32	HALT
33	HALT
34	HALT
35	HALT
36	HALT
37	HALT
38	HALT
39	HALT
40	HALT
41	HALT
42	HALT
43	HALT
44	HALT
45	HALT
46	HALT
47	HALT
48	HALT
49	HALT
50	HALT
51	HALT
52	HALT
53	HALT
54	HALT
55	HALT
56	HALT
57	HALT
58	HALT
59	HALT
60	HALT
61	HALT
62	HALT
63	HALT
64	HALT
65	HALT
66	HALT
67	HALT
68	HALT
69	HALT
70	HALT
71	HALT
72	HALT
73	HALT
74	HALT
75	HALT
76	HALT
77	HALT
78	HALT
79	HALT
80	HALT
81	HALT
82	HALT
83	HALT
84	HALT
85	HALT
86	HALT
87	HALT
88	HALT
89	HALT
90	HALT
91	HALT
92	HALT
93	HALT
94	HALT
95	HALT
96	HALT
97	HALT
98	HALT
99	HALT





656	000564	000566	
657	000566	000000	H + N L T
658	000570	000572	H + N L T
659	000572	000000	H + N L T
660	000574	000576	H + N L T
661	000578	000000	H + N L T
662	000600	000602	H + N L T
663	000602	000000	H + N L T
664	000604	000606	H + N L T
665	000606	000000	H + N L T
666	000610	000612	H + N L T
667	000612	000000	H + N L T
668	000614	000616	H + N L T
669	000616	000000	H + N L T
670	000620	000622	H + N L T
671	000622	000000	H + N L T
672	000624	000626	H + N L T
673	000626	000000	H + N L T
674	000630	000632	H + N L T
675	000632	000000	H + N L T
676	000634	000636	H + N L T
677	000636	000000	H + N L T
678	000640	000642	H + N L T
679	000642	000000	H + N L T
680	000644	000646	H + N L T
681	000646	000000	H + N L T
682	000650	000652	H + N L T
683	000652	000000	H + N L T
684	000654	000656	H + N L T
685	000656	000000	H + N L T
686	000660	000662	H + N L T
687	000662	000000	H + N L T
688	000664	000666	H + N L T
689	000666	000000	H + N L T
690	000670	000672	H + N L T
691	000672	000000	H + N L T
692	000674	000676	H + N L T
693	000676	000000	H + N L T
694	000700	000702	H + N L T
695	000702	000000	H + N L T
696	000704	000706	H + N L T
697	000706	000000	H + N L T
698	000710	000712	H + N L T
699	000712	000000	H + N L T
700	000714	000716	H + N L T
701	000716	000000	H + N L T
702	000720	000722	H + N L T
703	000722	000000	H + N L T
704	000724	000726	H + N L T
705	000726	000000	H + N L T
706	000730	000732	H + N L T
707	000732	000000	H + N L T
708	000734	000736	H + N L T
709	000736	000000	H + N L T
710	000740	000742	H + N L T
711	000742	000000	H + N L T

712	000744	000746	.	+	2
713	000746	000000	HALT		
714	000750	000752	.	+	2
715	000752	000000	HALT		
716	000754	000756	.	+	2
717	000756	000000	HALT		
718	000760	000762	.	+	2
719	000762	000000	HALT		
720	000764	000766	.	+	2
721	000766	000000	HALT		
722	000770	000772	.	+	2
723	000772	000000	HALT		
724	000774	000776	.	+	2
725	000776	000000	HALT		
726		000020	.	+	20
727	000020	006630	LOOP		
728	000022	000340	340		
729	000024	006704	PWRDWN		
730	000026	000340	340		
731		000030	.	+	30
732	000030	006616	EMTRP		
733	000032	000340	340		
734					



```

735                                     :SOFTWARE SWITCH REGISTER*****
736                                     :=176
737 000176 000176 000000                SWREG: 0                ;SOFTWARE SWITCH REGISTER
738                                     ;PROGRAM START*****
739                                     :=200
740                                     :
741 000200 000137 001104                JMP START                ;GO TO THE START OF THE TEST
742 000204 000137 006034                JMP MASTER                ;ENTER THE SCOPE LOOP ROUTINE
743                                     :=1000
744                                     :I/O REGISTERS
745 001000 177570                        SR: 177570                ;SWITCH REGISTER
746 001002 177776                        CSR: 177776                ;PROCESSOR STATUS REGISTER
747 001004 177566                        TPB: 177566                ;TELETYPE REGISTERS
748 001006 177562                        TKB: 177562
749 001010 177564                        TPS: 177564
750 001012 177560                        TKS: 177560
751                                     :INDEX=SELECT+2
752                                     :
753                                     :DN11 REGISTERS
754                                     :
755 001014 175200                        DNCSR1: 175200
756 001016 175202                        DNCSR2: 175202
757 001020 175204                        DNCSR3: 175204
758 001022 175206                        DNCSR4: 175206
759                                     :
760                                     :
761                                     :
762                                     :
763 001024 000300                        VECTOR: 300
764 001026 000004                        PRIORITY: 4
765                                     :
766                                     :
767                                     :

```

```

768 :PROGRAM WORK REGISTER
769 :
770 001030 177777 FTITLE: 177777 :TITLE PRINTED FLAG
771 001032 000000 DSSCNT: 0
772 001034 000000 WORK: 0
773 001036 000000 WORK1: 0
774 001040 000000 COUNT: 0
775 001042 000000 TIME: 0
776 001044 000000 TIME1: 0
777 001046 000000 SAVE: 0
778 001050 000000 ERCOUNT: 0
779 001052 175200 STATUS: 175200
780 001054 000000 PNT1: 0
781 001056 000000 PNT2: 0
782 001060 000000 PNT3: 0
783 001062 000000 PNT4: 0
784 001064 000000 FLAG: 0
785 001066 000000 PASS: 0
786 001070 000000 MASK: 0
787 001072 011722 STKLINK: STACK
788 001074 000 000 MAP: .BYTE 0,0,0,0
789 001077 000
790 :
791 :
792 :
793 001100 001054 ENTRY: PNT1
794 001102 006362 POINT: PH01

```



```

795
796
797
798
799
800 001104 000005
801 001106 013706 001072
802 001112 005237 001030
803 001116 001002
804 001120 104001
805 001122 010670
806 001124 004737 010222
807 001130 005737 001064
808 001134 001071
809 001136 052737 177777 001064
810 001144 012703 006340
811 001150 104001
812 001152 007471
813 001154 004737 006212
814 001160 012702 001014
815 001164 004737 007144
816 001170 013737 001014 001016
817 001176 062737 000002 001016
818 001204 013737 001016 001020
819 001212 062737 000002 001020
820 001220 013737 001020 001022
821 001226 062737 000002 001022
822 001234 013737 001014 001052
823 001242 012703 006340
824 001246 104001
825 001250 007521
826 001252 004737 006212
827 001256 012702 001024
828 001262 004737 007144
829 001266 022737 001000 001024
830 001274 101762
831 001276 062737 000002 001024
832 001304 012777 000200 177512
833 001312 162737 000002 001024
834 001320 013737 001014 001052
835 001326 032777 002000 177444
836 001334 001402
837 001336 000137 004310
  
```

```

: THIS ROUTINE IS USED TO INITIALIZE THE PROGRAM TO THE CORRECT
: DN11 REGISTER ASSIGNMENTS THIS ROUTINE IS ONLY ENTERED ONCE
: UPON THE FIRST START OF THE PROGRAM
START: RESET
      MOV STKLINK,%6 ;SET UP THE STACK
      INC FTITLE
      BNE IS ;SKIP TITLE IF ALREADY PRINTED
      EMT+1 ;GO TYPE OUT THE TITLE
      MTITLE
IS: JSR PC,SUSWR ;GO TO SWITCH REGISTER SIZING ROUTINE
     TST FLAG ;TEST FOR THE PASS
     BNE NOTFIRST ;BRANCH NOT THE FIRST PASS
     BIS #177777,FLAG ;SET PASS INDICATOR
     MOV #TEXBUF,%3 ;SET UP TO RECEIVE DATA FROM TTY
     EMT +1 ;ASK OPERATOR FOR FIRST DN11 ADDRESS
     DNADDR
     JSR %7,TYST ;GO FETCH ADDRESS FROM TTY
     MOV #DNCSR1,%2
     JSR %7,NEXCHAR ;CONVERT OCTAL TO ASCII
     MOV DNCSR1,DNCSR2 ;SET UP ALL DN11 ADDRESSES
     ADD #2,DNCSR2
     MOV DNCSR2,DNCSR3
     ADD #2,DNCSR3
     MOV DNCSR3,DNCSR4
     ADD #2,DNCSR4
     MOV DNCSR1,STATUS
GETVEC: MOV #TEXBUF,%3 ;SET UP TO ASK FOR VECTOR ASSIGNMENT
        EMT +1
        VECDN
        JSR %7,TYST ;FETCH VECTOR ADDRESS FROM TTY
        MOV #VECTOR,%2
        JSR %7,NEXCHAR ;CONVERT OCTAL TO ASCII
        CMP #1000,VECTOR ;IS VECTOR ADDRESS LESS THAN 1000
        BLOS GETVEC ;BRANCH THE ADDRESS IS GREATER THAN 1000
        ADD #2,VECTOR ;POINT TO VECTOR PSW
        MOV #200,VECTOR ;SET PRIORITY AT 4
        SUB #2,VECTOR ;ADJ. VECTOR
NOTFIRST: MOV DNCSR1,STATUS
          BIT #BIT10,JSR ;TEST TO ENTER ON-LINE TEST ONLY
          BEQ .+6 ;ENTER STATIC
          JMP BEGIN ;ENTER ON-LINE TEST
  
```

00000  
00001  
00002  
00003  
00004  
00005  
00006  
00007  
00008  
00009  
00010  
00011  
00012  
00013  
00014  
00015  
00016  
00017  
00018  
00019  
00020  
00021  
00022  
00023  
00024  
00025  
00026  
00027  
00028  
00029  
00030  
00031  
00032  
00033  
00034  
00035  
00036  
00037  
00038  
00039  
00040  
00041  
00042  
00043  
00044  
00045  
00046  
00047  
00048  
00049  
00050  
00051  
00052  
00053  
00054  
00055  
00056  
00057  
00058  
00059  
00060  
00061  
00062  
00063  
00064  
00065  
00066  
00067  
00068  
00069  
00070  
00071  
00072  
00073

```

:*****DIALEX-11*****
:DN11 TEST PART1
:AUTOMATIC DIALER INTERFACE
:
:THE FIRST PART OF THIS TEST CONSISTS OF
:INTER-ACTION BETWEEN THE OPERATOR AND
:THE PROGRAM
:
:IS PWO CLEARED
ST1: JSR PC,CKSWP ;CHECK FOR <IG>
;BITB,JSR ;DOES THE OPERATOR WANT TO SELECT ONE DN11
BEQ ST1X ;NO RUN NORMAL
MOV JSR,WORK ;FETCH WHICH DN11 HE WANTS TO RUN
BIC #177774,WORK ;MASK COUNT
CLC ;COUNT TIMES 2
ROL WORK ;CLEAR DN11 NUMBER SELECTED PREVIOUSLY
BIC #7,STATUS ;SET UP SELECTED DN11
ADD WORK,STATUS ;TEST FOR 801
ST1X: TST #STATUS ;NO 801 PRESENT
;+12
MOV #177777,MASK
BR ;+14
MOV #77777,MASK
EMT +1
MES1
ST1XE: BIC #177777,#STATUS ;TEST STATUS BIT
;BIT #STATUS ;BRANCH IF POWER OFF
BPL ST2XE ;*** ERROR 0 ***
ERRO: MOV #0,ERCOUNT ;REPORT ERROR
N=N+1
JSR %5,STAER
ST2XE: SCOPE
ST1XE

```

001342	004737	010326	
001346	032777	000400	177424
001354	001417		
001356	017737	177416	001034
001364	042737	177774	001034
001372	000241		
001374	006137	001034	
001400	042737	000007	001052
001406	063737	001034	001052
001414	005777	177432	
001420	100404		
001422	012737	177777	001070
001430	000405		
001432	012737	077777	001070
001440	104001		
001442	007200		
001444	042777	177777	177400
001452	033777	001070	177372
001460	100005		
001462	012737	000000	001050
000001			
004537	007716		
000004			
001444			

```
874  
875  
876  
877  
878  
879 001500 042777 177777 177344 ST2X: BIC #177777,STATUS  
880 001506 032777 040000 177336 BIT #BIT14,STATUS  
881 001514 001405 BEQ ST3E ;BRANCH IF ACR IS CLEARED  
882 001516 012737 000001 001050 ERR1: MOV #1,ERCOUNT ;*** ERROR 1 ***  
883 000002 N=N+1  
884 001524 004537 007716 JSR %5,STAER  
885 001530 000004 ST3E: SCOPE  
886 001532 001500 ST2X  
887  
888  
889  
890 001534 042777 177777 177310 ST3: BIC #177777,STATUS  
891 001542 032777 020000 177302 BIT #BIT13,STATUS  
892 001550 001405 BEQ ST4E ;BIT 13 CLEAR EXIT  
893 001552 012737 000002 001050 ERR2: MOV #2,ERCOUNT ;*** ERROR 2 ***  
894 000003 N=N+1  
895 001560 004537 007716 JSR %5,STAER  
896 001564 000004 ST4E: SCOPE  
897 001566 001534 ST3  
898  
899  
900  
901 001570 042777 177777 177254 ST4: BIC #177777,STATUS ;CLEAR THE WORLD  
902 001576 032777 010000 177246 BIT #BIT12,STATUS ;TEST FOR DATA LINE NOT OCCUPIED  
903 001604 001405 BEQ ST5E ;BRANCH IF LINE NOT OCCUPIED  
904 001606 012737 000003 001050 ERR3: MOV #3,ERCOUNT ;*** ERROR 3 ***  
905 000004 N=N+1  
906 001614 004537 007716 JSR %5,STAER ;REPROT ERROR  
907 001620 000004 ST5E: SCOPE  
908 001622 001570 ST4  
909  
910  
911  
912 001624 042777 177777 177220 ST5: BIC #177777,STATUS ;CLEAR THE WORLD  
913 001632 032777 007400 177212 BIT #7400,STATUS ;TEST BCD DIGITS  
914 001640 001405 BEQ ST6E ;BITS SHOULD BE CLEARED  
915 001642 012737 000004 001050 ERR4: MOV #4,ERCOUNT ;*** ERROR 4 ***  
916 000005 N=N+1  
917 001650 004537 007716 JSR %5,STAER ;REPORT ERROR  
918 001654 000004 ST6E: SCOPE  
919 001656 001624 ST5
```



```

920
921
922
923
924
925 001660 042777 177777 177164 ST6: BIC #177777,STATUS
926 001666 012777 007400 177156      MOV #7400,STATUS ;SET BITS
927 001674 017737 177152 001034      MOV STATUS,WORK
928 001702 042737 170377 001034      BIC #170377,WORK ;MASK ALL OTHER BITS
929 001710 022737 007400 001034      CMP #7400,WORK
930 001716 001412          BEQ ST7E ;BRANCH IF BITS ALL SET
931 001720 012737 000005 001050 ERRS: MOV #5,ERCOUNT ;*** ERROR 5 ***
932          000006          N=N+1
933 001726 012737 007400 001036      MOV #7400,WORK1 ;BCD BITS THAT SHOULD BE SET
934 001734 004537 010014          JSR %5,STAER1 ;REPORT ERROR
935 001740 004537 007716          JSR %5,STAER
936 001744 000004          ST7E: SCOPE
937 001746 001660          ST6
938          :CAN WE FLOAT A ONE THROUGH THE BCD BITS
939 001750 042777 177777 177074 ST7: BIC #177777,STATUS
940 001756 012737 000400 001036      MOV #400,WORK1

```

```

941 001764 013777 001036 177060 ST7X: MOV WORK1, @STATUS ; SET UP BCD BITS
942 001772 017737 177054 001034 MOV @STATUS, WORK ; READ BACK BCD BITS
943 002000 042737 170377 001034 BIC @170377, WORK ; MASK BITS
944 002006 023737 001036 001034 CMP WORK1, WORK ; DO THE BITS EQUAL WHAT WAS LOADED
945 002014 001007 ST7ER BNE ST7ER ; ERROR IN BITS READ BACK
946 002016 022737 007400 001034 CMP @7400, WORK ; EXIT PATTERN COMPLETE
947 002024 001412 BEQ ST10E ; SETUP NEXT PATTERN
948 002026 105237 001037 INCB WORK1+1 ; LOAD THE NEXT PATTERN
949 002032 000754 BR ST7X
950 002034 ST7ER: MOV #6, ERCOUNT ; *** ERROR 6 ***
951 002034 ERR6: N=N+1 ; REPORT ERROR
952 000007 JSR %5, STAER1
953 002042 004537 010014 JSR %5, STAER
954 002046 004537 007716 ST10E: SCOPE
955 002052 000004 ST7
956 002054 001750 ; IS DONE CLEARED
957
958
959
960
961 002056 042777 177777 176766 ST10: BIC @177777, @STATUS ; CLEAR THE WORLD
962 002064 105777 176762 TSTB @STATUS ; TEST FOR NOT DONE
963 002070 100005 BPL ST11E ; BRANCH IF DONE NOT SET
964 002072 012737 000007 001050 ERR7: MOV #7, ERCOUNT ; *** ERROR 7 ***
965 000010 N=N+1
966 002100 004537 007716 JSR %5, STAER ; REPORT DONE SET
967 002104 000004 ST11E: SCOPE
968 002106 002056 ST10

```

```

969
970
971
972 002110 042777 177777 176734 ST11: BIC #177777,@STATUS ;CLEAR THE WORLD
973 002116 032777 000100 176726 BIT #BIT6,@STATUS ;WAS BIT6 CLEARED
974 002124 001405 BEQ ST12E ;BRANCH IF BIT6 CLEARED
975 002126 012737 000010 001050 ERR10: MOV #10,ERCOUNT ;*** ERROR 10 ***
976 000011 N=N+1
977 002134 004537 007716 JSR %5,STAER ;REPORT BIT6 SET
978 002140 000004 ST12E: SCOPE
979 002142 002110 ST11
980
981
982
983 002144 012777 000340 176630 ST12: MOV #340,@CSR ;LOCK UP CPU
984 002152 052777 000100 176672 BIS #BIT6,@STATUS ;SET INTERRUPT ENABLE
985 002160 032777 000100 176664 BIT #BIT6,@STATUS ;WAS THE BIT SET
986 002166 001006 BNE ST12EX ;BIT SET CLEAR INTERRUPT
987 002170 000005 RESET ;CLEAR INTERRUPTS
988 002172 012737 000011 001050 ERR11: MOV #11,ERCOUNT ;*** ERROR 11 ***
989 000012 N=N+1
990 002200 004537 007716 JSR %5,STAER ;REPORT ERROR
991 002204 005077 176642 ST12EX: CLR @STATUS ;CLEAR INTERRUPTS
992 002210 000004 SCOPE
993 002212 002144 ST12

```



999999  
999999  
999999  
999999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016

002214	042777	177777	176630
002214	032777	000040	176622
002230	001405		
002232	012737	000012	001050
	000013		
002240	004537	007716	
002244	000004		
002246	002214		
002250	042777	177777	176574
002256	032777	000020	176566
002264	001405		
002266	012737	000013	001050
	000014		
002274	004537	007716	
002300	000004		
002302	002250		

:: IS DSS CLEARED

ST13:	BIC	#177777,STATUS	: CLEAR THE WORLD
	BIT	#BIT5,STATUS	: IS DSS SET
	BEO	ST14	: BRANCH IF DSS IS NOT SET
ERR12:	MOV	#12,ERCOUNT	: *** ERROR 12 ***
	N=N+1		
	JSR	%5,STAER	: REPORT DSS SET
ST14E:	SCOPE		
	ST13		

:: IS PND CLEARED

ST14:	BIC	#177777,STATUS	: CLEAR THE WORLD
	BIT	#BIT4,STATUS	: IS PND SET
	BEO	ST15	: BRANCH IF PND NOT SET
ERR13:	MOV	#13,ERCOUNT	: *** ERROR 13 ***
	N=N+1		
	JSR	%5,STAER	: REPORT PND SET
ST15E:	SCOPE		
	ST14		

```

1017
1018
1019
1020 002304 042777 177777 176540
1021 002312 032777 000010 176532
1022 002320 001405
1023 002322 012737 000014 001050
1024
1025 002330 004537 007716
1026 002334 000004
1027 002336 002304
1028
1029
1030
1031 002340 042777 177777 176504
1032 002346 052777 000010 176476
1033 002354 032777 000010 176470
1034 002362 001005
1035 002364 012737 000015 001050
1036
1037 002372 004537 007716
1038 002376 000004
1039 002400 002340
1040
1041
1042
1043 002402 042777 177777 176442
1044 002410 032777 000004 176434
1045 002416 001405
1046 002420 012737 000016 001050
1047
1048 002426 004537 007716
1049 002432 000004
1050 002434 002402
1051
1052
1053
1054 002436 042777 177777 176406
1055 002444 012777 000340 176330
1056 002452 052777 000004 176372
1057 002460 032777 000004 176364
1058 002466 001005
1059 002470 012737 000017 001050
1060
1061 002476 004537 007716
1062 002502 000004
1063 002504 002436
1064
1065
1066
1067 002506 042777 177777 176336
1068 002514 032777 000002 176330
1069 002522 001405
1070 002524 012737 000020 001050
1071
1072 002532 004537 007716

```

```

: IS THE MAINTENANCE BIT CLEARED
ST15: BIC      #177777,STATUS : CLEAR THE WORLD
      BIT      #BIT3,STATUS  : IS MAINTENANCE BIT CLEAR
      BEQ      ST15XE        : BRANCH IF MAINTENANCE CLEAR
ERR14: MOV     #14,ERCOUNT   : *** ERROR 14 ***
      N=N+1
      JSR     %5,STAER       : REPORT MAINTENANCE SET
ST15XE: SCOPE
      ST15

: CAN WE SET THE MAINTENANCE BIT
ST15X: BIC     #177777,STATUS : CAN WE SET MAINTENANCE
      BIS     #BIT3,STATUS  : IS MAINTENANCE SET
      BIT     #BIT3,STATUS  : YES EXIT
      BNE     ST16E        : *** ERROR 15 ***
ERR15: MOV     #15,ERCOUNT
      N=N+1
      JSR     %5,STAER
ST16E: SCOPE
      ST15X

: IS MASTER ENABLE CLEARED
ST16:  BIC     #177777,STATUS : CLEAR THE WORLD
      BIT     #BIT2,STATUS  : IS MASTER ENABLE CLEARED
      BEQ     ST16XE        : BRANCH IF CLEARED
ERR16: MOV     #16,ERCOUNT   : *** ERROR 16 ***
      N=N+1
      JSR     %5,STAER       : REPORT MASTER ENABLE STILL SET
ST16XE: SCOPE
      ST16

: CAN WE SET MASTER ENABLE
ST16X: BIC     #177777,STATUS
      MOV     #340,CCSR
      BIS     #BIT2,STATUS  : CAN WE SET MASTER ENABLE
      BIT     #BIT2,STATUS  : IS MASTER ENABLE SET
      BNE     ST17E        : YES EXIT
ERR17: MOV     #17,ERCOUNT   : *** ERROR 17 ***
      N=N+1
      JSR     %5,STAER       : REPORT ERROR
ST17E: SCOPE
      ST16X

: IS DPR CLEARED
ST17:  BIC     #177777,STATUS : CLEAR THE WORLD
      BIT     #BIT1,STATUS  : IS DATA PRESENT CLEARED
      BEQ     ST17XE        : BRANCH IF DATA PRESENT AS CLEARED
ERR20: MOV     #20,ERCOUNT   : *** ERROR 20 ***
      N=N+1
      JSR     %5,STAER       : REPORT DATA PRESENT IS SET

```

MD-11-DZDNR-B, DN11 DIALEX  
DZDNRB.P11 01-SEP-77 09:31

MACY11 30(1046) 01-SEP-77 09:57 PAGE 29

SEQ 0029

1073 002536 000004  
1074 002540 002506

ST17XE: SCOPE  
ST17



```

1075
1076          ;CAN WE SET DPR (DIGIT PRESENT)
1077
1078          ST17X: BIC      #177777, @STATUS
1079          MOV      #BIT0, @STATUS ; SET PND
1080          BIS      #BIT1, @STATUS ; SET DPR
1081          BIT      #BIT1, @STATUS
1082          BNE     ST20E
1083          MOV     #21, ERRCOUNT ; *** ERROR 21 ***
1084          N=N+1
1085          JSR     %5, STAER
1086          ST20E: SCOPE
1087          ST17X
1088
1089          ; IS CRQ (CALL REQUEST) CLEARED
1090
1091          ST20:  BIC      #177777, @STATUS
1092          BIT      #BIT0, @STATUS ; IS CALL REQUEST SET
1093          BEQ     ST21E ; NO! IT SHOULD NEVER BE SET
1094          MOV     #22, ERRCOUNT ; *** ERROR 22 ***
1095          N=N+1
1096          JSR     %5, STAER ; REPORT ERROR
1097          ST21E: SCOPE
1098          ST20
1099          ; TEST THE ABILITY OF SETTING CRQ
1100          ST21:  BIC      #177777, @STATUS ; CLEAR THE WORLD
1101          BIT      MASK, @STATUS ; IS THE DN11 CLEAR
1102          BEQ     ST22 ; DN11 OK
1103          MOV     #23, ERRCOUNT ; *** ERROR 23 ***
1104          N=N+1
1105          JSR     %5, STAER ; REPORT ERROR "STATUS REG. NOT CLEAR"
1106          BR      ST22E ; EXIT ERROR OCCURRED
1107          BIS      #BIT0, @STATUS ; SET CALL REQUEST
1108          BIT      #BIT0, @STATUS ; IS CALL REQUEST SET
1109          BNE     ST22X ; YES! EXIT
1110          MOV     #24, ERRCOUNT ; *** ERROR 24 ***
1111          N=N+1
1112          JSR     %5, STAER ; REPORT ERROR
1113          BR      ST22E ; LOOP ON ERRO
1114          ST22X: TSTB    @STATUS ; DONE SHOULD NOT BE SET
1115          BPL     ST22E ; BRANCH IF DONE NOT SET
1116          MOV     #25, ERRCOUNT ; *** ERROR 25 ***
1117          N=N+1
1118          JSR     %5, STAER ; REPORT THE ERROR
1119          ST22E: SCOPE
1120          ST21
1121
1122          ; DOES MAINTENANCE SET DLO
1123
1124          ST23:  BIC      #177777, @STATUS ; CLEAR THE WORLD
1125          BIT      MASK, @STATUS ; IS REG CLR
1126          BEQ     ST24 ; YES! EXIT
1127          MOV     #26, ERRCOUNT ; *** ERROR 26 ***
1128          N=N+1
1129          JSR     %5, STAER ; REPORT ERROR
1130          BR      ST25E

```

1131	003010	052777	000010	176034	ST24:	BIS	#BIT3,%STATUS	
1132	003016	032777	010000	176026		BIT	#BIT12,%STATUS	: IS DLO SET
1133	003024	001005				BNE	ST25E	: YES EXIT
1134	003026	012737	000027	001050	ERR27:	MOV	#27,ERCOUNT	: *** ERROR 27 ***
1135		000030				N=N+1		
1136	003034	004537	007716			JSR	%5,STAER	: REPORT ERROR
1137	003040	000004			ST25E:	SCOPE		
1138	003042	002756				ST23		
1139								

```

1140
1141
1142
1143
1144          ;CAN WE SET PND IN MAINTENANCE MODE
1145
1146 003044 042777 177777 176000 ST25: BIC      #177777,@STATUS
1147 003052 033777 001070 175772      BIT      MASK,@STATUS          ;IS REG CLEAR
1148 003060 001406                BEQ      ST25X          ;YES! REG CLEAR
1149 003062 012737 000030 001050 ERR30: MOV      #30,ERCOUNT      ;*** ERROR 30 ***
1150                N=N+1
1151 003070 004537 007716                JSR      %5,STAER
1152 003074 000425                BR       ST26E          ;LOOP ON ERROR
1153 003076 052777 000411 175746 ST25X: BIS      #BIT8!BIT3!BIT0,@STATUS ;SET PND IN MAINTENANCE MODE.
1154 003104 032777 000020 175740      BIT      #BIT4,@STATUS
1155 003112 001016                BNE     ST26E
1156 003114 012737 000031 001050 ERR31: MOV      #31,ERCOUNT      ;PND SET
1157                N=N+1          ;*** ERROR 31 ***
1158 003122 004537 007716                JSR      %5,STAER          ;PND NOT SET
1159 003126 000410                BR       ST26E          ;EXIT ERROR OCCURRED
1160
1161          ;DOES PND SET DONE
1162
1163 003130 105777 175716                ST26: TSTB   @STATUS          ;IS DONE SET
1164 003134 100405                BMI     ST26E          ;BRANCH IF DONE SET
1165 003136 012737 000032 001050 ERR32: MOV      #32,ERCOUNT      ;*** ERROR 32 ***
1166                N=N+1
1167 003144 004537 007716                JSR      %5,STAER
1168 003150 000004                ST26E: SCOPE
1169 003152 003044                ST25
1170
1171          ;CAN WE SET DSS IN MAINTENANCE MODE
1172
1173 003154 042777 177777 175670 ST26X: BIC      #177777,@STATUS
1174 003162 033777 001070 175662      BIT      MASK,@STATUS          ;IS REG. CLEAR
1175 003170 001406                BEQ      ST27          ;BRANCH IF CLEAR
1176 003172 012737 000033 001050 ERR33: MOV      #33,ERCOUNT      ;*** ERROR 33 ***
1177                N=N+1
1178 003200 004537 007716                JSR      %5,STAER
1179 003204 000425                BR       ST27E          ;REPORT ERROR
1180 003206 052777 001011 175636 ST27: BIS      #BIT9!BIT3!BIT0,@STATUS ;SET DSS IN MAINTENANCE MODE
1181 003214 032777 000040 175630      BIT      #BIT5,@STATUS
1182 003222 001006                BNE     ST30
1183 003224 012737 000034 001050 ERR34: MOV      #34,ERCOUNT      ;*** ERROR 34 ***
1184                N=N+1
1185 003232 004537 007716                JSR      %5,STAER
1186 003236 000410                BR       ST27E          ;LOOP ON ERROR

```



```

1187
1188
1189
1190 003240 105777 175606 ST30: TSTB @STATUS ; WAS DONE SET BY DSS
1191 003244 100405 ST27E ; BRANCH IF YES
1192 003246 012737 000035 001050 ERR35: MOV #35,ERCOUNT ; *** ERROR 35 ***
1193 000036 N=N+1
1194 003254 004537 007716 JSR %5,STAER ; REPORT ERROR
1195 003260 000004 ST27E: SCOPE
1196 003262 003154 ST26X
1197
1198 003264 042777 177777 175560 : CAN WE SET PWO IN MAINTENANCE MODE
1199 003272 033777 001070 175552 ST31: BIC #177777,@STATUS
1200 003300 001406 BEQ ST31X ; IS STATUS REG CLEAR
1201 003302 012737 000036 001050 ERR36: MOV #36,ERCOUNT ; BRANCH IF REG CLEAR
1202 000037 ; *** ERROR 36 ***
1203 003310 004537 007716 JSR %5,STAER ; REPORT ERROR
1204 003314 000432 BR ST32E ;
1205 003316 052777 000010 175526 ST31X: BIS #BIT3,@STATUS ; SET MAINTENANCE
1206 003324 042777 177760 175520 BIC #177760,@STATUS
1207 003332 052777 002001 175512 BIS #BIT10!BIT0,@STATUS ; SET PWO
1208 003340 005777 175506 TST @STATUS
1209 003344 100406 ST32
1210 003346 012737 000037 001050 ERR37: MOV #37,ERCOUNT ; *** ERROR 37 ***
1211 000040 N=N+1
1212 003354 004537 007716 JSR %5,STAER
1213 003360 000410 BR ST32E ; REPORT ERROR

```

```

1214
1215
1216 003362 105777 175464 :WAS DONE SET
1217 003366 100405 ST32: TSTB @STATUS ;TEST DONE
1218 003370 012737 000040 001050 ERR40: BMI ST32E ;YES EXIT
1219 000041 N=N+1 ;*** ERROR 40 ***
1220 003376 004537 007716 JSR %5,STAER ;DONE NOT SET REPORT ERROR
1221 003402 000004 ST32E: SCOPE
1222 003404 003264 ST31
1223 :CAN WE SET ABANDON CALL AND RETRY IN MAINTENANCE MODE
1224 003406 042777 177777 175436 ST32X: BIC #177777,@STATUS
1225 003414 033777 001070 175430 BIT MASK,@STATUS ;IS REG. CLEAR
1226 003422 001406 BEQ ST33 ;YES EXIT
1227 003424 012737 000041 001050 ERR41: MOV #41,ERCOUNT ;*** ERROR 41 ***
1228 000042 N=N+1
1229 003432 004537 007716 JSR %5,STAER
1230 003436 000425 BR ST34E ;REPORT REG. NOT CLEAR
1231 003440 052777 004011 175404 ST33: BIS #BIT11!BIT3!BIT0,@STATUS ;SET ACR IN MAINTENANCE MODE
1232 003446 032777 040000 175376 BIT #BIT14,@STATUS
1233 003454 001006 BNE ST34
1234 003456 012737 000042 001050 ERR42: MOV #42,ERCOUNT ;*** ERROR 42 ***
1235 000043 N=N+1
1236 003464 004537 007716 JSR %5,STAER ;ACR NOT SET
1237 003470 000410 BR ST34E
1238 :DID ACR SET DONE
1239 003472 105777 175354 ST34: TSTB @STATUS ;WAS DONE SET BY ABANDON CALL AND RETRY
1240 003476 100405 BMI ST34E ;YES EXIT
1241 003500 012737 000043 001050 ERR43: MOV #43,ERCOUNT ;*** ERROR 43 ***
1242 000044 N=N+1
1243 003506 004537 007716 JSR %5,STAER ;DONE NOT SET
1244 003512 000004 ST34E: SCOPE
1245 003514 003406 ST32X
1246 :DOES PND CLEAR DIGIT PRESENT
1247 003516 042777 177777 175326 ST34X: BIC #177777,@STATUS
1248 003524 033777 001070 175320 BIT MASK,@STATUS ;IS REG CLR
1249 003532 001406 BEQ ST35 ;YES BRANCH
1250 003534 012737 000044 001050 ERR44: MOV #44,ERCOUNT ;*** ERROR 44 ***
1251 000045 N=N+1
1252 003542 004537 007716 JSR %5,STAER
1253 003546 000434 BR ST35E ;LOOP ON ERROR
1254 003550 052777 000410 175274 ST35: BIS #BIT8!BIT3,@STATUS ;SET PND
1255 003556 052777 000002 175266 BIS #BIT1,@STATUS ;SET DIGIT PRESENT
1256 003564 032777 000002 175260 BIT #BIT1,@STATUS ;IS IT SET
1257 003572 001006 BNE ST35X ;YES BRANCH
1258 003574 012737 000045 001050 ERR45: MOV #45,ERCOUNT ;*** ERROR 45 ***
1259 000046 N=N+1
1260 003602 004537 007716 JSR %5,STAER ;REPORT DIGIT PRESENT NOT SET
1261 003606 000414 BR ST35E ;LOOP ON ERROR
1262
1263

```

```

1264                                     :WAS DIGIT PRESENT CLEARED
1265 003610 042777 000410 175234 ST35X: BIC #BIT8!BIT3,STATUS :CLEAR PND
1266 003616 032777 000002 175226      BIT #BIT1,STATUS :WAS DIGIT PRESENT CLEARED
1267 003624 001405                                     BEQ ST35E :YES BRANCH
1268 003626 012737 000046 001050 ERR46: MOV #46,ERCOUNT :*** ERROR 46 ***
1269                                     N=N+1
1270 003634 004537 007716      JSR %5,STAER :REPORT ERROR
1271 003640 000004      ST35E: SCOPE
1272 003642 003516      ST34X
1273                                     :CAN WE SET AND CLEAR DONE
1274 003644 105777 175202      ST36: TSTB STATUS :TEST DONE
1275 003650 100006      BPL ST36X :DONE CLEAR BRANCH
1276 003652 012737 000047 001050 ERR47: MOV #47,ERCOUNT :*** ERROR 47 ***
1277                                     N=N+1
1278 003660 004537 007716      JSR %5,STAER :REPORT ERROR
1279 003664 000427                                     BR ST36E :LOOP ON ERROR
1280 003666 052777 000200 175156 ST36X: BIS #BIT7,STATUS :SET DONE

```



```

1281
1282
1283
1284 003674 105777 175152 :CAN WE CLEAR DONE
1285 003700 100406 CLRDN: TSTB @STATUS ;IS DONE SET
1286 003702 012737 000050 001050 ERR50: MOV #50,ERCOUNT ;YES DONE SET
1287 000051 N=N+1 ;*** ERROR 50 ***
1288 003710 004537 007716 JSR %5,STAER ;REPORT ERROR
1289 003714 000413 BR ST36E ;LOOP ON ERROR
1290 003716 042777 000200 175126 DONSET: BIC #BIT7,@ST1RUS ;CLR DONE
1291 003724 105777 175122 TSTB @STATUS ;IS DONE CLEARED
1292 003730 100005 BPL ST36E ;YES EXIT
1293 003732 012737 000051 001050 ERR51: MOV #51,ERCOUNT ;*** ERROR 51 ***
1294 000052 N=N+1
1295 003740 004537 007716 JSR %5,STAER ;DONE NOT CLEARED
1296 003744 000004 ST36E: SCOPE
1297 003746 003644 ST36
1298 :CAN WE GENERATE AN INTERRUPT
1299 003750 000005 ST37: RESET
1300 003752 012777 004002 175044 MOV #INTER,@VECTOR ;SET UP INTERRUPT VECTOR
1301 003760 012777 000240 175014 MOV #240,@CSR ;SET CPU PRIORITY LEVEL TO BR5
1302 003766 052777 000300 175056 BIS #BIT6!BIT7,@STATUS ;SET INTERRUPT ENABLE AND DONE
1303 003774 000240 NOP ;NO INTERRUPT SHOULD MASTER ENABLE NOT SET
1304 003776 000137 004030 JMP ST37X ;GO SET MASTER ENABLE TO SEE IF WE CAN INTERRUPT
1305 ;REPORT AN INTERRUPT OCCURRED THAT IS AN ERROR
1306 004002 012777 000340 174772 INTER: MOV #340,@CSR
1307 004010 005077 175036 CLR @STATUS
1308 004014 012737 000052 001050 ERR52: MOV #52,ERCOUNT ;*** ERROR 52 ***
1309 000053 N=N+1
1310 004022 004537 007716 JSR %5,STAER
1311 004026 000441 BR ST38E
1312 004030 012777 004052 174766 ST37X: MOV #SECINT,@VECTOR ;SET UP INTERRUPT VECTOR
1313 004036 052777 000004 174750 BIS #BIT2,@DNCSR1 ;SET MASTER ENABLE
1314 004044 000240 NOP ;NO INTERRUPT SHOULD OCCUR PROCESSOR IS AT LEVEL 5
1315 004046 000137 004066 JMP ST38 ;WE SHOULD NOT INTERRUPT LEVEL TO HIGH
1316 004052
1317 004052 012737 000053 001050 SECINT: ERR53: MOV #53,ERCOUNT ;*** ERROR 53 ***
1318 000054 N=N+1
1319 004060 004537 007716 JSR %5,STAER ;REPORT THE DN11 INTERRUPTED
1320 004064 000422 BR ST38E ;ENTER SCOPE LOOP
1321 004066 012777 004132 174730 ST38: MOV #ST38E,@VECTOR ;SET UP FOR INTERRUPT
1322 004074 005077 174702 CLR @CSR ;LOWER PROCESSOR PRIORITY
1323 004100 005037 001042 CLR TIME ;SET UP TIMER
1324 004104 005237 001042 INC TIME ;WAIT FOR INTERRUPT
1325 004110 001375 BNE -4
1326 004112 012777 000340 174662 ERR54: MOV #340,@CSR ;LOCK UP CPU DN11 DID NOT INTERRUPT
1327 004120 012737 000054 001050 MOV #54,ERCOUNT ;*** ERROR 54 ***
1328 000055 N=N+1
1329 004126 004537 007716 JSR %5,STAER ;REPORT DN11 DID NOT INTERRUPT AT BRO
1330 004132 013706 001072 ST38E: MOV STKLINK,%6 ;RESET STACK
1331 004136 000004 SCOPE
1332 004140 003750 ST37
1333 004142 032777 000400 174630 ST40: BIT #BIT8,@SR ;DID THE OPERATOR SELECTED A DN11
1334 004150 001402 BEQ +6 ;NO! NORMAL RUN
1335 004152 000137 001342 JMP ST1 ;THE OPERATOR DID SELECTED A DN11
1336 004156 023737 001022 001052 CMP DNCSR4,STATUS

```

1337	004164	001414				BEQ	RESTART	
1338	004166	062737	000002	001052		ADD	#2,STATUS	
1339	004174	052777	000100	174650		BIS	#BIT6,STATUS	;SET INTERRUPT ENABLE
1340	004202	032777	000100	174642		BIT	#BIT6,STATUS	;IF SET DN11 IS POSSABILE THERE
1341	004210	001402				BEQ	+6	
1342	004212	000137	001342			JMP	ST1	
1343	004216	013737	001014	001052	RESTART:	MOV	DNCSR1,STATUS	
1344	004224	032777	001000	174546		BIT	#BIT9,SR	;TEST IF THE OPERATOR WANTS TO LOOP ON STATIC TESTS
1345	004232	001410				BEQ	MYCNT+2	;BRANCH TO ON-LINE TEST IF BIT 9 NOT SET
1346	004234	005737	004252			TST	MYCNT	
1347	004240	001405				BEQ	MYCNT+2	
1348	004242	005337	004252			DEC	MYCNT	
1349	004246	000137	001414			JMP	ST1X	;LOOP ON STATIC TESTS BIT9 SET
1350	004252	000100			MYCNT:	MOV	100	
1351	004254	012737	000100	004252		MOV	#100,MYCNT	
1352	004262	012737	177770	001044		MOV	#177770,TIME1	;WAIT FOR PHONE LINE TO SETTLE
1353	004270	005037	001042			CLR	TIME	
1354	004274	005237	001042			INC	TIME	
1355	004300	001375				BNE	-4	
1356	004302	005237	001044			INC	TIME1	
1357	004306	001372				BNE	.-12	
1358								
1359								
1360								
1361								
1362								
1363								
1364								
1365								
1366								
1367								
1368								
1369								
1370								
1371								
1372								
1373								
1374								
1375								
1376								
1377								
1378								
1379								
1380								
1381								
1382								
1383								
1384								

```

1385 ;DIALEX 11
1386 ;THE OPERATOR MUST ASSIGN PHONE NUMBER TO EACH DN11
1387 BEGIN: MOV STKLINK,%6 ;SET UP STACK
1388 JSR PC,CKSWR ;CHECK FOR CNTL G
1389 RESET
1390 MOV #INT,AVECTOR ;SET UP VECTOR
1391 CLR %4
1392 TST NOFLAG
1393 BNE .+6
1394 BR NEWNO
1395 NOFLAG: 0
1396 JSR PC,CKSWR ;CHECK FOR <?G>
1397 BIT #BIT7,ASR ;TEST FOR TTY CONVERSATION
1398 BNE DNDIAL ;BRANCH IF NO CONVERSATION
1399 CLR MAP ;CLEAR PHONE MAP
1400 CLR MAP+2
1401 BIS #177777,NOFLAG
1402 CLR COUNT
1403 MOV #PH01,%3 ;SET UP PHONE #1 BUFFER
1404 EMT +1
1405 PH1
1406 JSR %7,TYST ;FETCH KEYBOARD CHAR
1407 TSTB PH01
1408 BEQ NO1 ;OPERATOR FAILED TO TYPE A PHONE NUMBER
1409 INC COUNT
1410 BISB #377,MAP(4) ;LOAD MAP
1411 INC %4
1412 MOV #PH02,%3
1413 EMT +1
1414 PH2
1415 JSR %7,TYST ;FETCH KEYBOARD CHARACTER
1416 TSTB PH02 ;DID THE OPERATOR TYPE A PHONE NUMBER
1417 BEQ NO2 ;THE OPERATOR ONLY GAVE THE PROGRAM ONE NUMBER
1418 INC COUNT ;THE OPERATOR TYPED AN NUMBER
1419 BISB #377,MAP(4) ;LOAD MAP
1420 INC %4
1421 MOV #PH03,%3
1422 EMT +1
1423 PH3
1424 JSR %7,TYST ;FETCH KEYBOARD CHARACTER
1425 TSTB PH03 ;DID THE OPERATOR TYPE A NUMBER FOR LINE THREE
1426 BEQ NO3 ;NO NUMBER FOR LINE THREE
1427 INC COUNT ;OPERATOR TYPE D A NUMBER
1428 BISB #377,MAP(4) ;LOAD MAP
1429 INC %4
1430 MOV #PH04,%3
1431 EMT +1
1432 PH4
1433 JSR %7,TYST ;FETCH KEYBOARD CHARACTER
1434 TSTB PH04 ;TEST IF THE OPERATOR TYPED A NUMBER FOR THIS LINE
1435 BEQ DNDIAL ;OPERATOR DID NOT TYPE A NUMBER
1436 BISB #377,MAP(4) ;LOAD MAP
1437 INC COUNT
1438 MOV DNCSR1,STATUS
1439 MOV COUNT,SAVE
1440 CLR %4

```



```

1441
1442
1443
1444
1445 004606 004737 010326
1446 004612 105764 001074
1447 004616 001423
1448 004620 005777 174226
1449 004624 001415
1450 004626 017737 174220 001034
1451 004634 013737 001052 001036
1452 004642 012737 000055 001050 ERR55:
1453 000056
1454 004650 004537 010102
1455 004654 000000
1456 004656 000777
1457 004660 005337 001046 MODOK:
1458 004664 001405
1459 004666 005204 EXMODOK:
1460 004670 062737 000002 001052 INC
1461 004676 000743 BR
1462
1463
1464
1465 004700 005037 001054
1466 004704 005037 001056
1467 004710 005037 001060
1468 004714 005037 001062
1469 004720 005004
1470 004722 013737 001040 001046
1471 004730 013737 001014 001052
1472 004736 105764 001074 SETCRQ:
1473 004742 001423
1474 004744 012777 000101 174100
1475 004752 032777 010000 174072
1476 004760 001411
1477 004762 012737 000056 001050 ERR56:
1478 000057
1479 004770 017737 174056 001034
1480 004776 004537 010102
1481 005002 000755
1482 005004 005337 001046
1483 005010 001405
1484 005012 005204 EXCRQ:
1485 005014 062737 000002 001052
1486 005022 000745
1487
1488
1489 005024 005077 173752
1490 005030 013737 001040 001032
1491 005036 012737 177700 001044
1492 005044 012737 000000 001042
1493 005052 052777 000004 173734
1494 005060 005237 001042 TWOSEC:
1495 005064 001375
1496 005066 005237 001044 INC

```

```

: TEST FOR DN11'S TO BE IN READY STATE
: REGISTERS SHOULD BE CLEARED BECAUSE
: OF RESET COMMAND ISSUED AT START
RINGO: JSR PC_CKSWR ; CHECK FOR CNTL G
TSTB MAP(4) ; IS THIS LINE ACTIVE
BEO EXMODOK ; BRANCH NOT ACTIVE
TST @STATUS
BEQ MODOK ; DN11 READY OK
MOV @STATUS, WORK ; FETCH CONTENTS OF REGISTER
MOV STATUS, WORK1 ; FETCH ADDRESS OF REGISTER
MOV #55, ERCOUNT ; *** ERROR 55 ***
N=N+1
JSR %5, STAER2 ; REPORT DN11 NOT READY
HALT
BR ; YOU CAN NOT CONTINUE
; ON UNTIL DN11 IS MADE READY
; GO SET CALL REQUEST
MODOK: DEC SAVE
SETPT
EXMODOK: INC %4
ADD #2, STATUS ; SET UP FOR NEXT DN11
BR RINGO ; TEST NEXT DN11

: SET UP DIGIT POINTERS FOR PHONE NUMBERS
: FOR DAILING SEQUENCE
SETPT: CLR PNT1 ; SET UP DIGIT POINTER ONE
CLR PNT2 ; SET UP DIGIT POINTER TWO
CLR PNT3 ; SET UP DIGIT POINTER THREE
CLR PNT4 ; SET UP DIGIT POINTER FOUR
%4
MOV COUNT, SAVE ; SET UP TO ENABLE CRQ
MOV DNCSR1, STATUS ; SET UP DN11 POINTER
TSTB MAP(4) ; IS THIS ACTIVE
BEQ EXCRQ
MOV #101, @STATUS ; SET CRQ - INT. ENABLE - MASTER ENABLE
BIT #BIT12, @STATUS ; TEST FOR DLO SET
BEQ DLOTST ; DLO SET OK!
MOV #56, ERCOUNT ; *** ERROR 56 ***
N=N+1
MOV @STATUS, WORK ; FETCH CONTENTS OF STATUS REGISTER
JSR %5, STAER2
BR SETCRQ
DLOTST: DEC SAVE
WAITIN ; GO WAIT FOR INTERRUPTS
EXCRQ: INC %4
ADD #2, STATUS ; SET UP FOR NEXT DN11
BR SETCRQ ; SET UP NEXT DN11

: SET UP TO COUNT DSS INTERRUPTS AND TIME OUT IF ALL
: DSS INTERRUPTS DO NOT OCCUR
WAITIN: CLR @CSR
MOV COUNT, DSSCNT ; NUMBER OF DN11
MOV #177700, TIME1
MOV #0, TIME
BIS #BIT2, @DNCSR1 ; SET MASTER ENABLE
INC TIME ; WAIT FOR DSS
BNE -4
INC TIME1

```

```

1497 005072 001372 BNE THOSEC
1498 005074 012777 000340 173700 MOV #340,DCSR ;LOCK UP CPU
1499 005102 104001 ENT #1 ;REPORT TIME OUT
1500 005104 007411 TIMO
1501 005106 005004 CLR #4
1502 005110 012737 000057 001050 ERR57: MOV #57,ERCOUNT ;*** ERROR 57 ***
1503 000060 N=N+1
1504 005116 013737 001040 001046 MOV COUNT,SAVE ;SET UP TO FETCH DN11 REGISTERS
1505 005118 013737 001014 001052 DNSTATE: MOV DNCSR1,STATUS ;TEST IF THE LINE IS ACTIVE
1506 005120 105764 001074 TSTB MAP(4) ;LINE NOT ACTIVE CHECK NEXT
1507 005122 001410 BEQ MPDN ;FETCH DN11 STATUS
1508 005124 017737 173706 001034 MOV #STATUS,WORK ;REPORT STATUS
1509 005126 004537 010102 JSR %5,STAER2
1510 005128 005337 001046 DEC SAVE
1511 005130 001521 BEQ REPEND ;GO REPORT END
1512 005132 003204 MPDN: INC #4
1513 005134 002737 000002 001052 ADD #2,STATUS ;SET UP TO TEST NEXT DN11
1514 005136 000760 BR DNSTATE
1515 005138 012737 177770 001044 END: MOV #177770,TIME1 ;SET UP TIME TO LET PHONE RING
1516 005140 005037 001042 CLR TIME
1517 005142 005237 001042 INC TIME
1518 005144 001375 BNE #4
1519 005146 005237 001044 INC TIME1
1520 005148 001372 BNE #12
1521 005150 032777 010000 173552 BIT #BIT12,DCSR ;TEST HOW DO WE TERMINATE THE CALL
1522 005152 001457 BEQ NOTCRQ ;CALL TERMINATED BY RESET
1523 005154 013737 001040 001046 MOV COUNT,SAVE
1524 005156 013737 001014 001052 CLRDN: MOV DNCSR1,STATUS
1525 005158 005004 CLR #4
1526 005160 105764 001074 TSTB MAP(4) ;IS THIS LINE ACTIVE
1527 005162 001440 BEQ EXCLRDN ;LINE IS NOT ACTIVE
1528 005164 042777 000001 173570 BIC #BIT0,STATUS ;CLEAR CRQ
1529 005166 013737 177770 001044 MOV #177770,TIME1
1530 005168 005037 001042 CLR TIME
1531 005170 005237 001042 INC TIME ;WIAT FOR DSS TO COME BACK
1532 005172 001375 BNE #4
1533 005174 005237 001044 INC TIME1
1534 005176 001372 BNE #12
1535 005178 005310 032777 000040 173534 BIT #BIT5,STATUS ;TEST FOR DSS
1536 005180 005316 001413 BEQ DSSCLR ;DSS CLEARED BY CRQ
1537 005182 013737 001052 001036 MOV STATUS,WORK1 ;SET UP FOR ERROR REPORT
1538 005184 017737 173520 001034 MOV #STATUS,WORK
1539 005186 005334 012737 000060 001050 ERR60: MOV #60,ERCOUNT ;*** ERROR 60 ***
1540 000061 N=N+1
1541 005342 004537 010102 JSR %5,STAER2 ;REPORT ERROR
1542 005346 005337 001046 DSSCLR: DEC SAVE
1543 005352 001423 BEQ REPEND ;RECYCLE
1544 005354 005204 EXCLRDN: INC #4
1545 005356 062737 000002 001052 ADD #2,STATUS ;GO CLEAR NEXT DN11
1546 005364 000730 BR CLRDN+2
1547
1548 ;
1549 ;DN11'S MUST BE CLEARED BY RESET IN THIS TEST
1550
1551 005366 004737 010326 NOTCRQ: JSR PC,CKSWR ;CHECK FOR CNTL G
1552 005372 000005 RESET ;CLEAR THE WORLD

```

1553	005374	012737	177770	001044	MOV	8177770,TIME1	;SET UP TO WAIT FOR LINES TO SETTLE DOWN
1554	005400	005037	001042		CLR	TIME	
1555	005400	005237	001042		INC	TIME	
1556	005400	001372			BNE	-4	
1557	005400	005237	001044		INC	TIME1	
1558	005400	001372			BNE	-12	
1559	005400	104001			EMT	+1	
1560	005400	007464		REPEND:	MESEND		;REPORT END
1561	005400	000137	001320		JMP	NOTFIRST	;RECYCLE TEST
1562				...			
1563				...			
1564				...			



```

1565
1566
1567          ; INTERRUPT HANDLER FOR DIALEX
1568          ; ON LINE TEST
1569
1570 005432 042777 000004 173354 INT:   BIC   #BIT2, DNCSR1 ; CLEAR MASTER ENABLE
1571 005440 013737 001040 001046      MOV   COUNT, SAVE ; SET UP TO COUNT DN11'S
1572 005446 013737 001014 001052      MOV   DNCSR1, STATUS ; SET UP ADDRESS ASSIGNMENT
1573 005454 012737 006362 001102      MOV   #PHO1, POINT ; FETCH NUMBER POINT
1574 005462 012737 001054 001100      MOV   #PNT1, ENTRY
1575 005470 005004          CLR   %4
1576 005472 105764 001074 DNTST: TSTB  MAP(4) ; IS THE LINE ACTIVE
1577 005476 001511          BEQ   EXINC ; BRANCH THE LINE IS NOT ACTIVE
1578 005500 105777 173346      TSTB  #STATUS ; IS THE DONE FLAG SET
1579 005504 100077          BPL   INCDN ; NO INTERRUPT FROM THIS DN11
1580 005506 032777 160000 173336      BIT   #160000, #STATUS ; ERROR ? (PHO-ACR-BIT13 UNUSED)
1581 005514 001404          BEQ   NOERROR ; BRANCH NO ERROR
1582 005516 005777 173356      TST   #ENTRY ; IS IT THE END OF CALL
1583 005522 100525          BMI   DSSSET ; YES ACR SET END OF CALL
1584 005524 000532          BR    REPORT ; REPORT ERROR OCCURRED
1585 005526 032777 000040 173316 NOERROR: BIT   #BIT5, #STATUS ; IS DSS SET
1586 005534 001120          BNE   DSSSET ; BRANCH IF DSS SET
1587 005536 032777 000020 173306      BIT   #BIT4, #STATUS ; TEST FOR PND
1588 005544 001006          BNE   PNDSET ; PND SET OK!
1589 005546 012737 000061 001050 ERR61: MOV   #61, ERCOUNT ; *** ERROR 61 ***
1590          N=N+1
1591 005554 004537 010102          JSR   %5, STAER2
1592 005560 000451          BR    INCDN
1593 005562 013737 001102 005604 PNDSET: MOV   POINT, INDEX ; SET UP TO FETCH DIGIT
1594 005570 017703 173304          MOV   #ENTRY, %3 ; SET UP DIGIT POINTER
1595 005574 005777 173300          TST   #ENTRY
1596 005600 100441          BMI   INCDN
1597 005602 116337 000000 001034 SELECT: MOVB 0(3), WORK ; FETCH DIGIT
1598 005610 032737 000377 001034      BIT   #377, WORK ; IS THIS THE LAST DIGIT
1599 005616 001012          BNE   LASTDG ; BRANCH IF NOT LAST DIGIT
1600 005620 052777 100000 173252      BIS   #BIT5, #ENTRY ; SET END OF CALL FLAG
1601 005626 032777 004000 173144      BIT   #BIT11, #SR ; TEST FOR EON OPTION
1602 005634 001060          BNE   DSSSET ; 801 DOES NOT HAVE EON OPTION
1603 005636 012737 000012 001034      MOV   #12, WORK ; LOAD END OF NUMBER CODE
1604 005644 042737 000360 001034 LASTDG: BIC   #360, WORK
1605 005652 013700 001052          MOV   STATUS, %0 ; LOAD DIGIT INTO TOP BYTE OF DN11 REGISTER
1606 005656 005200          INC   %0
1607 005660 113710 001034          MOVB WORK, %0 ; LOAD BCD DIGIT
1608 005664 042777 000200 173160 SETDPR: BIC   #BIT7, #STATUS ; CLEAR DONE
1609 005672 052777 000002 173152      BIS   #BIT1, #STATUS ; SET DPR
1610 005700 005277 173174          INC   #ENTRY
1611 005704 023737 001052 001022 INCDN: CMP   STATUS, DNCSR4 ; TEST FOR LAST DN11
1612 005712 001416          BEQ   EXDSS ; BRANCH ALL DN11 OPERATING
1613 005714 005337 001046          DEC   SAVE
1614 005720 001413          BEQ   EXDSS
1615 005722 005204          INC   %4
1616 005724 062737 000002 001100 EXINC: ADD   #2, ENTRY ; SET UP FOR NEXT DN11 POLE
1617 005732 062737 000002 001052      ADD   #2, STATUS ;
1618 005740 062737 000022 001102      ADD   #2, POINT ;
1619 005746 000651          BR    DNTST ; TEST NEXT DN11
1620

```

```

1621          :TEST DSS FOR OVERFLOW AND EXIT
1622          EXDSS:  TST      DSSCNT          ; DID WE RECEIVE DSS FROM ALL
1623          005750 001404          ; YES EXIT
1624          005754 001404          ; SET MASTER ENABLE
1625          005756 052777 000004 173030  BIC      @BIT2,@DNCSR1
1626          005764 000002          ;
1627          005766 012706 001000  RESTE:  MOV      @1000,%6          ; RESET STACK
1628          005772 000137 005172  JMP      END              ; RECYCLE PROGRAM
1629          :
1630          :
1631          :ROUTINE DEC DSS COUNT
1632          DSSSET: DEC      DSSCNT          ;
1633          005776 005337 001032          ;
1634          006002 042777 000200 173042  BIC      @BIT7,@STATUS ; CLEAR DONE
1635          006010 000735          ;
1636          :
1637          :ROUTINE TO REPORT DN11 ERROR
1638          REPORR:
1639          006012 012737 000062 001050  ERR62:  MOV      @62,ERCOUNT ; *** ERROR 62 ***
1640          000063          ;
1641          006020 004537 010102          ;
1642          006024 042777 000201 173020  JSR      %5,STAER2
1643          006032 000724          ;
1644          :
1645          :
1646          :
1647          :
1648          :
1649          :
1650          :MAINTENANCE ROUTINE FOR SETTING PULSER
1651          :
1652          006034 012737 000340 001002  MASTER:  MOV      @340,CSR          ; LOOK UP CPU. PRIORITY
1653          006042 013706 001072          ;
1654          006046 004737 010222          ;
1655          006052 017737 172722 001052  JSR      PC,SUSWR          ; CHECK FOR HARDWARE SWITCH REGISTER
1656          006060 000000          ; STORE ADDRESS
1657          006062 004737 010326          ; LOADSR FROM DOC. 5.3
1658          006066 017777 172706 172756  EXMAST:  JSR      PC,CKSWR          ; CHECK FOR <IG>
1659          006074 000240          ; MOVE SR INTO DN11 REGISTER
1660          006076 000240          ;
1661          006100 000240          ;
1662          006102 000240          ;
1663          006104 032777 000010 172666  BIT      @BIT3,@SR          ; TEST FOR MAINTENANCE MODE
1664          006112 001015          ; CLWAT
1665          006114 012737 177770 001040  MOV      @177770,COUNT ; BRANCH WE ARE IN MAINTENANCE MODE
1666          006122 005037 001042          ; WAIT 2 SECONDS FOR 801 SIGNALS
1667          006126 005237 001042          ;
1668          006132 001375          ;
1669          006134 005237 001040          ;
1670          006140 001370          ;
1671          006142 005077 172704          ;
1672          006146 032777 000010 172624  CLRWAT:  BIT      @BIT3,@SR          ; CLEAR DN11
1673          006154 001013          ; ARE WE IN MAINTENANCE MODE
1674          006156 012737 177770 001040  CLAREG  CLAREG          ; BRANCH NO NEED TO WAIT
1675          006164 005037 001042          ; WAIT FOR 801
1676          006170 005237 001042          ;
1677          CLRTIM:  INC      TIME

```

1677 006174 001375  
1678 006176 005237 001040  
1679 006202 001370  
1680 006204 005077 172642  
1681 006210 000724  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690 006212 010337 006472  
1691 006216 012777 000340 172556  
1692 006224 105777 172562  
1693 006230 100375  
1694 006232 117777 172550 172544  
1695 006240 105777 172544  
1696 006244 100375  
1697 006246 122777 000377 172532  
1698 006254 001007  
1699 006256 020337 006472  
1700 006262 001760  
1701 006264 104001  
1702 006266 007542  
1703 006270 005303  
1704 006272 000754  
1705 006274 105737 010324  
1706 006300 001016  
1707 006302 022777 000215 172476  
1708 006310 001411  
1709 006312 020327 006472  
1710 006316 001003  
1711 006320 104001  
1712 006322 007227  
1713 006324 000737  
1714 006326 117723 172454  
1715 006332 000734  
1716 006334 105023  
1717 006336 000207  
1718 006340 000000  
1719 006362  
1720 006362 000000  
1721 006404 006404  
1722 006404 000000  
1723 006426 006426  
1724 006426 000000  
1725 006450 006450  
1726 006450 000000  
1727 006472 006472  
1728 006472 000000  
1729  
1730  
1731  
1732

```

BNE CLRTIM
INC COUNT
BNE CLRTIM-4
BR @STATUS ;RECYCLE
EXMAST

:
:
: ROUTINE TO HANDLE KEYBOARD CONVERSATION
: ROUTINE ACCEPTS NUM. OR ALPHA CHARACTERS
:
TYST: MOV %3, BUFLIM ;SAVE FIRST ADDRESS OF INPUT BUFFER
      MOV @340, @CSR ;LOCK UP INTERRUPTS
TSTFLG: TSTB @TK5 ;CHECK FOR FLAG
      BPL TSTFLG ;
      MOV @TKB, @TPB ;CHARACTER IN BUFFER
      TSTB @TPS ;ECHO CHARACTER
      BPL -4 ;
      CMPB @377, @TKB ;CHECK FOR RUB-OUT
      BNE CKCH ;EXIT IF NOT RUB-OUT
      CMP %3, BUFLIM ;HAVE WE RUBBED-OUT BACK TO BEGINNING?
      BEQ TSTFLG ;YES, DO NOT RUBOUT ANYMORE
      EMT +1 ;
      MESB ;REPORT RUB-OUT ACKNOWLEDGED
      DEC %3 ;
      BR TSTFLG ;GO WAIT FOR NEW CHAR.
      TSTB TFIN ;CHECK TO SEE IF LOADING SWREG
      BNE 1$ ;IF SO GET OUT
      CMP @2:5, @TKB ;CHECK FOR CARRIAGE RETURN
      BEQ 2$ ;
      CMP %3, @BUFLIM ;ARE WE RUNNING OUT OF BUFFER SPACE?
      BNE 3$ ;NO, SKIP OVER ERROR MESSAGE
      EMT+1 ;YES, REPORT ERROR
      MES2 ;
      BR TSTFLG ;LOAD CHARACTER IN BUFFER
      MOV @TKB, (3)+ ;
      BR TSTFLG ;
      CLRB (3)+ ;EXIT DELIMITER TYPED
      RTS %7 ;
TEXBUF: 0
PH01: 0 =.+20
PH02: 0 =.+20
PH03: 0 =.+20
PH04: 0 =.+20
BUFLIM: 0 =.+20
      .EVEN

: SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.

```



```
1733 006474 011600 TYP: MOV @%6,%0 ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
1734 006476 062716 000002 ADD @2,@%6 ;SET UP EXIT.
1735 006502 011000 MOV @%0,%0 ;ADDRESS OF MESSAGE TO RO.
1736 006504 112037 006614 TYPA: MOV @0,TYPDAT ;GET CHARACTER
1737 006510 122737 000100 006614 CMPB @100,TYPDAT ;CHECK FOR "3" CHARACTER
1738 006516 001001 BNE TYPB ;BRANCH IF NOT "3"
1739 006520 000002 RTI ;TERMINATOR CHAR. DONE. EXIT.
1740 006522 122737 000045 006614 TYPB: CMPB @45,TYPDAT ;CHECK FOR "%".
1741 006530 001416 BEQ TYPF ;BRANCH IF "%".
1742 006532 122737 000042 006614 CMPB @42,TYPDAT ;NOT "%": CHECK FOR "#".
1743 006540 001417 BEQ TYPG ;BRANCH IF "#".
1744 006542 004737 006550 JSR %7,TYPD ;TYPE CHAR IN TYPDAT
1745 006546 000756 BR TYPB
1746 006550 113777 006614 172226 TYPD: MOV TYPDAT,@TPB ;OUTPUT CHARACTER TO PRINTER
1747 006556 105777 172226 TSTB @TPS ;WAIT FOR DONE FLAG.
1748 006562 100375 BPL -4
1749 006564 000207 TYEXIT: RTS %7 ;EXIT
1750 006566 112737 000015 006614 TYPF: MOVB @15,TYPDAT ;MOVE CARRIAGE RETURN CODE TO TYPDAT
1751 006574 004737 006550 JSR %7,TYPD ;GO TYPE CHAR.
1752 006600 112737 000012 006614 TYPG: MOVB @12,TYPDAT ;MOVE LF CODE TO TYPDAT.
1753 006606 004737 006550 JSR %7,TYPD ;GO TYPE CHAR.
1754 006612 000734 BR TYPB
1755 006614 000000 TYPDAT: 0
```

```
1756
1757 ;ROUTINE TO DECODE EMT CALLS FOR TTY
1758 006616 011600 EMTRP: MOV (6),%0
1759 006620 022740 104001 CMP @EMT+1,-(0) ;WAS CALL EMT+1
1760 006624 001101 BNE TYPB ;EMT+0
1761 006626 000722 BR TYPB
```

```
1762
1763
1764
1765 ;INDIVIDUAL STATIC TEST SCOPE LOOP ROUTINE
1766
1767
1768
1769 ;IF BIT 14 IS SET BYPASS THIS ROUTINE AND JUST LOOP ON THE TEST
1770 ;IF BIT 14 IS NOT SET LOOP ON EACH TEST 15 TIMES THEN GO TO THE NEXT TEST
1771
1772
```

```
1773 006630 004737 010326 LOOP: JSR PC,CKSWR ;CHECK FOR (16)
1774 006634 032777 040000 172136 BIT @BIT14,@SR ;TEST IF BIT 14 IS SET
1775 006642 001402 / BEQ +6 ;BRANCH IF BIT 14 IS NOT SET
1776 006644 013646 MOV @6+,-(6) ;PLAYING WITH THE STACK
1777 006646 000002 RTI ;LOOP ON TEST WITHOUT ENTERING THIS ROUTINE
1778 006650 005737 001066 TST PASS ;TEST IF THE PASS COUNT IS ZERO
1779 006654 001003 BNE +10 ;PASS COUNT NOT ZERO KEEP COUNTING
1780 006656 012737 000035 001066 MOV @35,PASS ;SET UP PASS COUNT FIRST TIME THROUGH
1781 006664 005337 001066 DEC PASS ;-1 PASS THIS TIME THROUGH
1782 006670 001402 BEQ +6 ;PASS ZERO ENTER NEXT TEST
1783 006672 013646 MOV @6+,-(6) ;PLAYING WITH THE STACK AGAIN
1784 006674 000002 RTI ;RE-ENTER TEST
1785 006676 062716 000002 ADD @2,(6) ;INC STACK FOR THE NEXT TEST
1786 006702 000002 RTI ;EXIT TO THE NEXT TEST
1787
1788
```

```
1789          :POWER FAIL SEQUENCE
1790
1791 006704 012737 006714 000024 PWRDWN: MOV      #PWRUP,24      ;SET UP POWER FAIL VECTOR FOR RESTART
1792 006712 000000                HALT                ;HALT AND WAIT FOR POWER TO COME BACK
1793
1794          :
1795          :THIS THE POWER UP SEQUENCE REPORT POWER HAS FAILED AND
1796          :WAIT TWO SECONDS FOR THE PHONE LINES TO SETTLE
1797          :
1798
1799 006714 012737 006704 000024 PWRUP:  MOV      #PWRDWN,24    ;SET UP POWER FAIL VECTOR FOR POWER DOWN
1800 006722 012706 001000                MOV      #1000,%6    ;SET UP THE STACK
1801 006726 104000                EMT      +0
1802 006730 007670                HED6
1803 006732 177777                -1                ;REPORT THE POWER HAS FAILED
1804 006734 012737 177770 001044 INCTM: MOV      #177770,TIME1 ;SET THE TWO SECOND TIMER
1805 006742 005237 001042                INC      TIME
1806 006746 001375                BNE     INCTM
1807 006750 005237 001044                INC      TIME1
1808 006754 001372                BNE     INCTM
1809 006756 022737 000176 001000        CMP      #SWREG,SR   ;CHECK FOR SWREG USE
1810 006764 001002                BNE     IS          ;IF NOT GO TO IS
1811 006766 004737 010376                JSR     PC,CNTLU    ;GO LOAD SWREG FROM TTY
1812 006772 000137 001104        IS:    JMP      START   ;GO TO THE BEGINNING OF THE PROGRAM AND RESTART
1813          :ROUTINE TO SAVE REGISTERS
1814 006776 010046        SAVEREG: MOV    %0,-(6) ;SAVE REGISTER 0
1815 007000 010146                MOV    %1,-(6) ;SAVE REGISTER 1
1816 007002 010246                MOV    %2,-(6) ;SAVE REGISTER 2
1817 007004 010346                MOV    %3,-(6) ;SAVE REGISTER 3
1818 007006 010446                MOV    %4,-(6) ;SAVE REGISTER 4
1819 007010 000115                JMP    (5)      ;EXIT ROUTINE
1820
1821          :ROUTINE TO RESTORE REGISTERS
1822 007012 005726        RESTORE: TST   (6)+
1823 007014 012604                MOV    (6)+,%4    ;RESTORE REGISTER 4
1824 007016 012603                MOV    (6)+,%3    ;RESTORE REGISTER 3
1825 007020 012602                MOV    (6)+,%2    ;RESTORE REGISTER 2
1826 007022 012601                MOV    (6)+,%1    ;RESTORE REGISTER 1
1827 007024 012600                MOV    (6)+,%0    ;RESTORE REGISTER 0
1828 007026 000205                RTS     %5        ;EXIT ROUTINE
1829
1830          :
```

```

1831
1832
1833
1834 007030 011600
1835 007032 062716 000002
1836 007036 011037 007056
1837 007042 022737 177777 007056
1838 007050 001001
1839 007052 000002
1840 007054 104001
1841 007056 000000
1842 007060 000763
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857 007062 013537 007142
1858 007066 012501
1859 007070 012502
1860 007072 060201
1861
1862 007074 013703 007142
1863 007100 042703 177770
1864 007104 062703 000060
1865 007110 110341
1866 007112 042737 000007 007142
1867 007120 006037 007142
1868 007124 006037 007142
1869 007130 006037 007142
1870 007134 005302
1871 007136 001356
1872 007140 000205
1873 007142 000000

```

```

;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
TYP5:  MOV    2%6,%0      ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
        ADD    2,%6      ;UPDATE TO NEXT MESSAGE ADDRESS
        MOV    3%0,TYP5B ;ADDRESS OF MESSAGE TO TYP5B
        CMP    #-1,TYP5B ;CHECK FOR TERMINATOR
        BNE    TYP5A     ;BRANCH IF NOT TERMINATOR.
        RTI                    ;TERMINATOR. EXIT.
TYP5A:  EMT    +1        ;CALL ON TYP SUB TO TYPE MESSAGE
TYP5B:  0
        BR    TYP5      ;ADDRESS OF MESSAGE GOES HERE
                    ;GO PROCESS NEXT MESSAGE

:
:OCTAL TO ASCII CONVERT ROUTINE
:ENTER ROUTINE AS FOLLOWS
:JSR %5,CONV
:ADDR# = ADDRESS OF NUMBER TO BE CONVERTED
:ADDR BYTE = LSB OF WHERE ASCII IS GOING
:ASCII# = THE NUMBER OF ASCII CHAR. TO BE CONVERTED
:
CONV:   MOV    2(5)+,ACNVX ;VALUE OF # TO BE CONVERTED
        MOV    (5)+,%1    ;ASCII ADDR
        MOV    (5)+,%2    ;# OF ASCII CHAR
        ADD    %2,%1
ACVN:   MOV    ACNVX,%3
        BIC    #177770,%3 ;ISOLATE LEAST SIGNIFICANT OCTAL #
        ADD    #60,%3     ;SET UP ASCII #
        MOVB  %3,-(1)    ;STORE ASCII CHAR
        BIC    #7,ACNVX
        ROR   ACNVX     ;ROTATE OCTAL #
        ROR   ACNVX
        ROR   ACNVX
        DEC   %2        ;-1 FROM ASCII CHAR COUNT
        BNE  ACVN
        RTS   %5       ;EXIT # CONVERTED
ACNVX:  0              ;WORK REGISTER

```



```

1874                                     ; THIS ROUTINES IS USED TO CONVERT ASCII INPUT TO OCTAL
1875                                     ;
1876                                     ;
1877 007144 012703 006340 NEXCHAR: MOV  #TEXTBUF,%3      ; FETCH ASCII POINTER
1878 007150 005012          CLR  %2                    ;
1879 007152 105713          TSTB %3                    ; TEST FOR LAST CHACTER
1880 007154 001410          BEQ  EXNEX                  ; LAST CHACTER EXIT
1881 007156 000241          CLC                          ;
1882 007160 006312          ASL  %2                    ;
1883 007162 006312          ASL  %2                    ;
1884 007164 006312          ASL  %2                    ;
1885 007166 142713 000370 BICB %370,%3      ; MASK OUT HI ORDER BITS
1886 007172 152312          BICB (%3)+,%2            ; LOAD OCTAL VALUE
1887 007174 000766          BR   NEXCHAR+6
1888 007176 000207          EXNEX: RTS  %7              ; EXIT LAST CHARACTER PROCESSED

```

1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944

007200	020045	044124	020105
007206	030070	020051	051511
007214	047440	043106	046040
007222	047111	027105	100
007227	045	042524	052130
007234	041040	043125	042506
007242	020122	051511	043040
007250	046125	027114	020040
007256	047516	046440	051117
007254	020105	047111	052520
007272	020124	046101	047514
007300	042527	020104	054105
007306	042503	052120	036040
007314	052522	047502	052125
007322	020076	051117	036040
007330	051103	027076	100
007335	045	044120	047117
007342	020105	030443	040077
007350	050045	047510	042516
007356	021440	037462	100
007363	045	044120	047117
007370	020105	031443	040077
007376	050045	047510	042516
007404	021440	037464	100
007411	045	051120	043517
007416	040522	020115	044524
007424	042515	020104	052517
007432	020124	047125	041101
007440	042514	052040	020117
007446	047503	050115	042514
007454	042524	041440	046101
007462	040114		

TTY OUTPUT FOR DIALEX-11

MES1: .ASCII /% THE BO1 IS OFF LINE.2/

MES2: .ASCII /%TEXT BUFFER IS FULL. NO MORE INPUT ALLOWED EXCEPT <RUBOUT> OR <CR>.2/

PH1: .ASCII /%PHONE #1?2/

PH2: .ASCII /%PHONE #2?2/

PH3: .ASCII /%PHONE #3?2/

PH4: .ASCII /%PHONE #4?2/

↑IMO: .ASCII /%PROGRAM TIMED OUT UNABLE TO COMPLETE CALL2/

1945						
1946	007464	042445	042116	100	MESEND: .ASCII	/%END@/
1947						
1948						
1949						
1950	007471	045	047104	030461	DNADDR: .ASCII	/%DN11 REGISTER ADDRESS?@/
1951	007476	051040	043505	051511		
1952	007504	042524	020122	042101		
1953	007512	051104	051505	037523		
1954	007520	100				
1955						
1956						
1957						
1958	007521	045	042526	052103	VECDN: .ASCII	/%VECTOR ADDRESS?@/
1959	007526	051117	040440	042104		
1960	007534	042522	051523	040077		
1961						
1962						
1963						
1964						
1965						
1966						
1967						
1968						
1969	007542	040057	047		MESB: .ASCII	/%@'%/
1970						
1971	007545	045	100		HEDO: .ASCII	/%@/
1972						
1973						
1974						
1975	007547	040	020040	042440	HED1: .ASCII	/ ERROR COUNT @/
1976	007554	051122	051117	041440		
1977	007562	052517	052116	020040		
1978	007570	040040				
1979						
1980						
1981						
1982	007572	020040	020040	020040	HED2: .ASCII	/ GD DATA @/
1983	007600	043440	020104	040504		
1984	007606	040524	020040	040040		
1985						
1986						
1987						
1988	007614	020040	020040	020040	HED3: .ASCII	/ BD DATA@/
1989	007622	041040	020104	040504		
1990	007630	040524	100			
1991						
1992						
1993						
1994	007633	040	020040	020040	HED4: .ASCII	/ DN11 @/
1995	007640	042040	030516	020061		
1996	007646	020040	100			
1997						
1998						
1999						
2000	007651	040	020040	020040	HED5: .ASCII	/ DNCSR @/



2001	007656	042040	041516	051123
2002	007664	026040	040040	
2003				
2004				
2005				
2006	007670	050045	053517	051105
2007	007676	043040	044501	020114
2008	007704	041517	052503	051122
2009	007712	042105	100	
2010				
2011				
2012				
2013	007716			

.....

HED6: .ASCII /%POWER FAIL OCCURRED3/

.EVEN

```

2014
2015
2016
2017
2018 007716 032777 020000 171054 STAER: BIT #BIT13,JSR ;TEST TO DELETE TYPE-OUT
2019 007724 001401 BEQ .+4 ;BRANCH TO TYPE
2020 007726 000205 RTS %5 ;DELETE TYPE-OUT
2021 007730 004537 006776 JSR %5,SAVEREG ;SAVE REGISTERS
2022 007734 004537 007062 JSR %5,CONV ;CONVERT OCTAL TO ASCII
2023 007740 001050 ERCOUNT
2024 007742 007547 HED1
2025 007744 000003 3
2026 007746 004537 007062 JSR %5,CONV
2027 007752 001052 STATUS
2028 007754 007633 HED4
2029 007756 000006 6
2030 007760 104000 EMT +0 ;REPORT ERROR NUMBER
2031 007762 007545 HED0
2032 007764 007547 HED1
2033 007766 007633 HED4
2034 007770 177777 -1
2035 007772 005777 171002 TST JSR ;TEST TO HALT ON ERROR
2036 007776 100001 BPL IS
2037 010000 000000 HALT
2038 010002 004737 010326 JSR PC,CKSWR ;CHECK FOR <IG>
2039 010006 004537 007012 JSR %5,RESTORE ;RESTORE REGISTERS
2040 010012 000205 RTS %5 ;EXIT
2041
2042
2043
2044 010014 032777 020000 170756 STAER1: BIT #BIT13,JSR ;TEST TO DELETE TYPE-OUT
2045 010022 001401 BEQ .+4 ;BRANCH TO TYPE
2046 010024 000205 RTS %5 ;BIT13 SET DELETE TYPE-OUT
2047 010026 004537 006776 JSR %5,SAVEREG ;SAVE REGISTERS
2048 010032 004537 007062 JSR %5,CONV ;CONVERT OCTAL TO ASCII
2049 010036 001036 WORK1 ;WHAT REGISTER SHOULD CONTAIN
2050 010040 007572 HED2
2051 010042 000006 6
2052 010044 004537 007062 JSR %5,CONV ;CONVERT OCTAL TO ASCII
2053 010050 001034 WORK ;WHAT REGISTER CONTAINED
2054 010052 007614 HED3
2055 010054 000006 6
2056 010056 104000 EMT +0 ;REPORT MESSAGE
2057 010060 007545 HED0
2058 010062 007572 HED2
2059 010064 007614 HED3
2060 010066 177777 -1
2061 010070 004537 007012 JSR %5,RESTORE ;RESTORE REGISTERS
2062 010074 004737 010326 JSR PC,CKSWR ;CHECK FOR <IG>
2063 010100 000205 RTS %5
2064
2065
2066
2067 010102 032777 020000 170670 STAER2: BIT #BIT13,JSR ;TEST TO DELETE TYPE-OUT
2068 010110 001401 BEQ .+4 ;BRANCH TO TYPE
2069 010112 000205 RTS %5 ;DELETE TYPE-OUT

```

```

2070 010114 004537 006776 JSR %5,SAVEREG ;SAVE REGISTERS
2071 010114 004537 007062 JSR %5,CONV ;CONVERT OCTAL TO ASCII
2072 010114 001050 ERCOUNT
2073 010114 007547 HED1
2074 010114 000003 J
2075 010114 017737 170714 001034 MOV @STATUS,WORK
2076 010114 004537 007062 JSR %5,CONV ;CONVERT OCTAL TO ASCII
2077 010114 001034 WORK
2078 010114 007651 HED5
2079 010114 000006 J
2080 010114 004537 007062 JSR %5,CONV ;CONVERT OCTAL TO ASCII
2081 010114 001052 STATUS
2082 010114 007633 HED4
2083 010114 000006 J
2084 010114 104000 EMT +0
2085 010114 007547 HED0
2086 010114 007547 HED1
2087 010114 007651 HED5
2088 010114 007633 HED4
2089 010114 177777 -1
2090 010200 005777 170574 TST @SR ;TEST TO DELETE HALT ON ERROR
2091 010204 000001 BPL IS ;BRANCH IF NO HALT WANTED
2092 010206 000000 HALT
2093 010210 004737 010326 JSR PC,CKSWR ;CHECK FOR (IG)
2094 010214 004537 007012 JSR %5,RESTORE ;RESTORE REGISTERS
2095 010220 000205 RTS %5
2096
2097
2098 ;HARDWARE SWITCH REGISTER SIZING ROUTINE*****
2099
2100
2101 010222 012737 177570 001000 SUSWR: MOV @177570,SR ;INITIALIZE SWITCH REGISTER ADDRESS
2102 010230 013746 000006 MOV @286,-(SP) ;SAVE VECTORS
2103 010234 013746 000004 MOV @284,-(SP)
2104 010240 012737 010260 000004 MOV @648,@284 ;SET UP FOR TIMEOUT
2105 010246 022777 170524 000004 CMP @-1,@SR ;REFERENCE HARDWARE SWITCH REGISTER
2106 010254 001402 BEQ 658
2107 010256 000404 BR 668
2108 010260 022636 648: CMP (SP)+,(SP)+ ;ADJUST STACK
2109 010262 012737 000176 001000 658: MOV @SWREG,SR ;POINT TO SOFTWARE SWITCH REG
2110 010270 012637 000004 668: MOV (SP)+,@284 ;RESTORE VECTORS
2111 010274 012637 000006 MOV (SP)+,@286
2112 010300 022737 000176 001000 CMP @SWREG,SR ;IS SWREG USED
2113 010306 001002 BNE 678
2114 010310 004737 010376 JSR PC,CNTLU ;ALLOW SWREG TO BE LOADED
2115 010314 000207 678: RTS PC
2116
2117

```