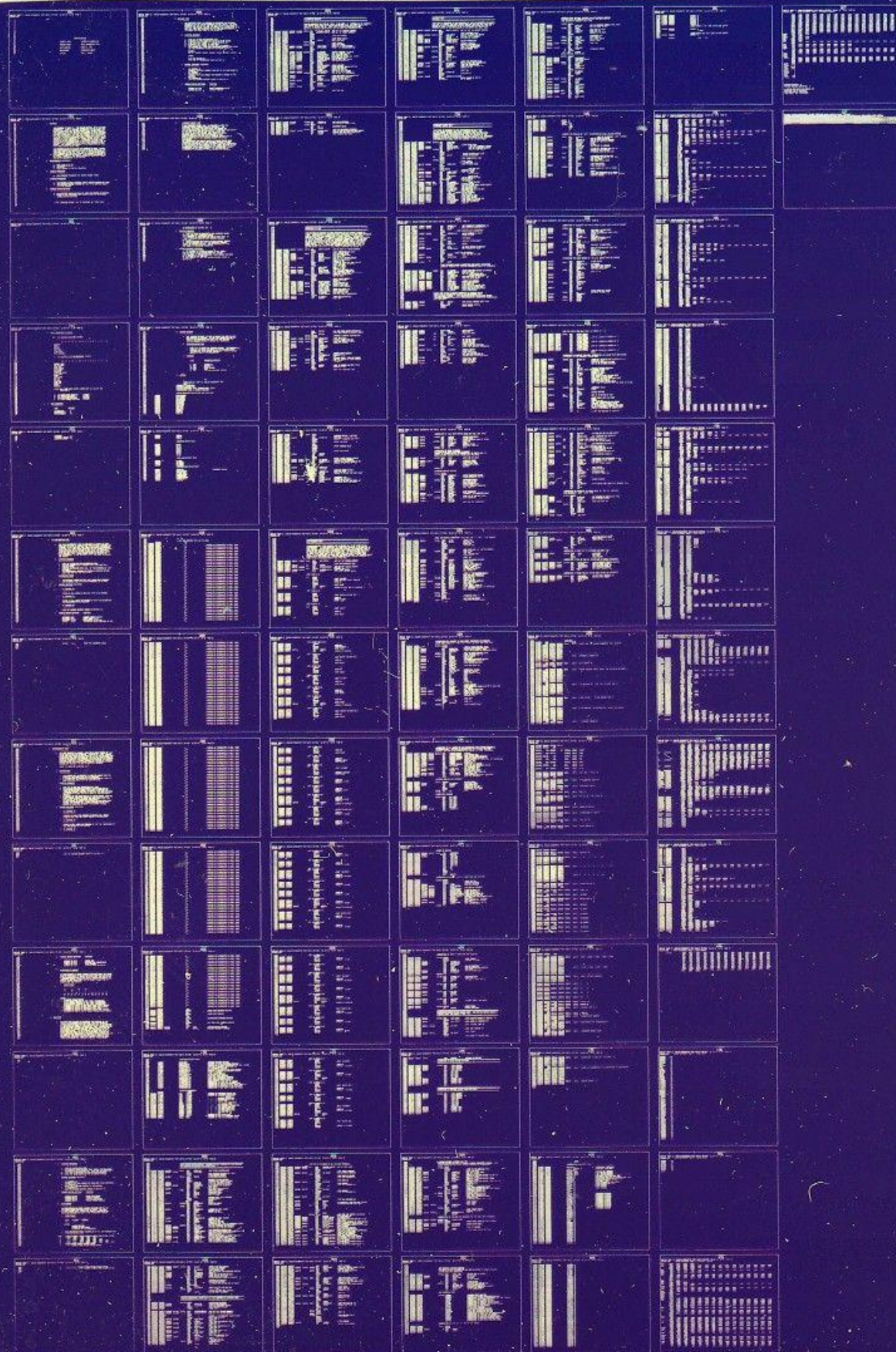


ADF11

ANALOG TESTS
MD-11-DZADH-A

EP-DZADH-A-DL-A
COPYRIGHT © 1976
FICHE 1 OF 1

NOV 1976
digital
MADE IN U.S.A



.REM X

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DZADH-A-D
PRODUCT NAME:	ADF11 DIAGNOSTIC TEST
DATE CREATED:	MARCH 4, 1974
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	EARL L. BOUSE

THIS PROGRAM IS PART II, THE ANALOG SECTION, OF A TWO PART DIAGNOSTIC. PART I (MAINDEC 11-0-208-A) IS THE LOGIC TEST SECTION AND SHOULD BE RUN AND PROVED FULLY OPERATIONAL BEFORE RUNNING THIS TEST SECTION. THE PROGRAM 1 IS SELF STARTING AND WHEN LOADED WILL TYPE OUT THE PROGRAM 1 TITLE AND REQUEST FOR THE A/D LENGTH TO BE TYPED. THE PROGRAM WILL ACCEPT A 10 TO 15 BIT UNIPOLAR OR BIPOLAR INPUT. EXAMPLE: 10(CR)* WOULD INDICATE A 10 BIT UNIPOLAR A/D; WHILE TYPING 10+(CR) WOULD INDICATE A 10 BIT BIPOLAR A/D. A SENTENCE IS THEN TYPED GIVING THE LETTER DESIGNATORS TO TYPE 0 TO 3 OF THE FOUR (4) SUBTESTS OF WHICH THIS PROGRAM IS COMPRISED. THE PROGRAM THEN WAITS IN A KEYBOARD INPUT MODE FOR A TEST LETTER TO BE TYPED.

THE PROGRAM IS SET UP TO GIVE THE OPERATOR A FLOW CONTROL OVER THE PROGRAM AS POSSIBLE VIA THE TELETYPE. BY PRESSING A 'C' (OBTAINED VIA TYPING THE 'CNTR' AND 'C' KEYS) THE PROGRAM WILL WHILE RUNNING ANY TEST WILL ENABLE THE FLOW CONTROL TO THE KEYBOARD MONITOR AND AWAITS A NEW LETTER DESIGNATOR TO BE TYPED. TYPING A '1A' WHILE IN MONITOR MODE WILL ENABLE THE LETTER DESIGNATORS TO BE RETYPED.

2. REQUIREMENTS (EQUIPMENT)

- A. POP-11/05,15,20,45
B. TELETYPE
C. ADF11 ANALOG TO DIGITAL CONVERTER

3. LOADING PROCEDURE

- A. USE STANDARD PROCEDURE FOR LOADING BINARY TAPES.

4. STARTING PROCEDURE

- A. THE PROGRAM IS SELF STARTING WITH A RESTART ADDRESS OF '200'.
B. THE ABSOLUTE RESTART ADDRESS IS '174' IF A NEW A/D LENGTH IS TO BE ENTERED.

5. CONSOLE SWITCH SETTINGS

- A. ALL SWITCHES SHOULD BE DOWN (0) WHEN THE PROGRAM IS STARTED.
B. REFER TO THE INDIVIDUAL TEST DESCRIPTIONS FOR APPLICABLE
CONSOLE SWITCH SETTINGS

* TYPE 'CARRIAGE RETURN' (CR) TO TERMINATE ALL INPUT DATA.

001

REF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 4
DLADHA.CF8

81

-1-

F01

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 6
DZADWA.CMB

138
139
140
141

RECOVERY 11.
INCREMENT MEMORY 12.
COMMAND DECODER 13.

-2-

G01

B. A/D CALIBRATION TEST

A. THE 'A/D CALIBRATION' TEST IS DESIGNED TO ACCEPT AN INPUT FROM THE TELETYPE TO INDICATE THE TYPE OF SYNC (EXTERNAL OR INTERNAL) TO BE USED AND THEN TAKES CONTINUOUS CONVERSIONS USING THE 'CH.' AND 'GAIN' SELECTED VIA THE CONSOLE SWITCHES. THESE SETTINGS MAY BE CHANGED AT ANY TIME. THE CONVERTED VALUE MAY BE PLACED IN RO AND DISPLAYED IN THE DATA LIGHTS WITH SWITCHES '9-11' WHICH ISSUES RESETS OR PRINTED ON THE TELEPRINTER. IF THE SWITCHES '9-11' ARE DOWN, NO RESETS ARE ISSUED.

B. STARTING SEQUENCE

1. TYPE 'C' TO RUN THE A/D CALIBRATION TEST.
2. THE TEST HEADER PLUS A REQUEST FOR A SYNC TYPE WILL THEN BE TYPED.
3. TYPE IN THE DESIRED SYNC (CR), 'I' FOR INTERNAL; 'E' FOR EXTERNAL.
4. THE PROGRAM WILL RESPOND VIA TYPING A CARRIAGE RETURN-LINE FEED AND THE TEST WILL START.

C. CALIBRATION ERROR

1. THE PRINTOUT 'ERROR BIT SET' WILL OCCUR WHILE RUNNING WITH EXTERNAL SYNC IF THE SYNC FREQUENCY IS TOO FAST.

D. CONTROL SWITCHES (TELETYPE)

1. ↑A (CONTROL A)

TYPING ↑A WILL ENABLE A NEW SYNC TYPE TO BE ENTERED.

2. ↑C (CONTROL C)

TYPING ↑C WILL CAUSE THE PROGRAM TO EXIT THE CALIBRATION TEST AND RETURN TO THE MONITOR.

3. ↑P (CONTROL P)

LOAD THE COMMAND DECODER (REFER TO SECTION 13.)

E. CONSOLE SWITCH SETTINGS

FUNCTION

SWITCHES '0-8'
 SWITCHES '9-11'
 SWITCHES '13-14'
 SWITCH '15=0'

CHANNEL SELECT (0-777)
 RESET COUNT (0-7)
 GAIN SELECT (1,2,4,8)
 CONVERSION VALUE DISPLAYED IN 'RO'

1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200

H01

ROF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 8
DZADHA.CMB

199
199
200
201

SWITCH '15=1'

PRINT THE CONVERTED VALUE

-3-

9. REPEATIBILITY TEST

- A. THE TEST REQUESTS A CHANNEL, GAIN AND A COUNT SPREAD OF '1-4' TO BE TYPED IN BY THE OPERATOR. A SERIES OF '1024' CONVERSIONS ARE THEN TAKEN ON THE INPUT CH.(S) AT THE SELECTED GAIN. CONVERSIONS ARE THEN AVERAGED OUT AND IF THE COUNT SPREAD IS FOUND TO BE GREATER THAN REQUEST, THE RESULTS OF THE CONVERSIONS ARE TYPED OUT. A SINGLE CHANNEL OR A SERIES OF CHANNELS MAY BE TESTED VIA TYPING EITHER 'N(CR)*' TO SELECT A SINGLE CHANNEL OR 'N,N(CR)' TO TEST A SERIES OF CHANNELS.

#NOTE 'N' WOULD BE A DECIMAL NO.

B. RESTRICTIONS

1. IF A SECOND CH. IS ENTERED, IT MUST BE LARGER THAN THE FIRST CH. THE INPUT WILL NOT BE ACCEPTED.
2. A VOLTAGE MUST BE PROVIDED TO THE SELECTED CH. WHICH IS LESS THAN FULL SCALE FOR THE SELECTED GAIN.

C. STARTING SEQUENCE

1. TYPE 'R' TO RUN THE 'REPEATIBILITY' TEST.
2. THE PROGRAM AUTOMATICALLY DOES A CORE SIZING AND IF THERE IS SUFFICIENT MEMORY AVAILABLE TO SUPPORT THE 'INCREMENT MEMORY' FEATURE OF THE A/D, THE USER IS GIVEN THE OPTION OF USING THIS AS A MEANS OF TESTING REPEATABILITY.
3. A REQUEST IS THEN MADE FOR CH.(S) TO BE TESTED, GAIN AND COUNT SPREAD (RANGE IN WHICH ALL 1024 COUNT MUST FALL FOR THE CH. TO BE CONSIDERED ACCEPTABLE).
4. IF THE CHANNEL IS FOUND TO BE WITHIN THE SELECTED COUNT SPREAD, THE PROGRAM WILL EITHER CONTINUE TO THE NEXT CHANNEL IF SELECTED OR RETEST THE CURRENT CHANNEL.

D. CONTROL SWITCHES (TELETYPE)

1. 1A (CONTROL A)

TYPING A 1A WHILE THE PROGRAM IS RUNNING WILL ENABLE A NEW CH.(S), GAIN AND COUNT SPREAD TO BE SELECTED.

2. 1C (CONTROL C)

TYPING 1C WILL CAUSE THE PROGRAM TO EXIT THE 'REPEATIBILITY' TEST AND RETURN TO THE MONITOR.

3. 1P (CONTROL P)

J01

ADFLA PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 10
DZADHA.CMB

LOAD THE COMMAND DECODER (REFER TO SECTION 13.)

-4-

258
259
260
261

E. CONSOLE SWITCH SETTINGSFUNCTION

CONSOLE SW 12=0
CONSOLE SW 12=1
CONSOLE SW 13=0
CONSOLE SW 13=1

NORMAL RUN
PRINTOUT ALL CONVERSIONS
PRINT ERRORS
INHIBIT ERROR PRINTOUTS

F. REPEATABILITY ERRORS

ON ENCOUNTERING AN ERROR (CONSOLE SWITCHES DOWN) THE ERROR DATA IS TYPED OUT. IT SHOULD BE NOTED THAT THIS MAY NOT BE A REPRESENTATION OF ALL '1024' COUNTS WHEN USING THE 'INC' FEATURE SINCE NO ATTEMPT IS MADE TO CATEGORIZE COUNTS WHICH FALL 'OUT OF RANGE' (MORE + OR -5 COUNTS FROM THE AVERAGE).

1. ERROR FORMAT

CH.	LO	AV	HI									
A	B	C	D									
LO	-5	-4	-3	-2	-1	0	+1	+2	+3	+4	+5	HI
E	F	G	H	I	J	K	L	M	N	O	P	Q

WHERE:

A=CHANNEL BEING TESTED
B=THE LOWEST READING OF THE '1024' CONVERSIONS
C=THE AVERAGE READING OF THE '1024' CONVERSIONS
D=THE HIGHEST READING OF THE '1024' CONVERSIONS
E=NUMBER OF COUNTS IN EACH PART LOWER THAN 5 COUNTS
F-J=NUMBER OF COUNTS IN EACH PART LOWER THAN AVERAGE.
K=NUMBER OF COUNTS AT AVERAGE OF THE '1024'
L-P=NUMBER OF COUNTS IN EACH PART HIGHER THAN AVERAGE.
Q=NUMBER OF COUNTS 'OUT OF RANGE' HIGHER THAN 5 COUNTS

10. GAIN TEST

- A. THE GAIN TEST IS USED TO DETERMINE THE ACCURACY OF THE 'ADF11' AT DIFFERENT GAIN SETTINGS. THE TEST REQUESTS 16 SPECIFIED VOLTAGES (16 FOR A UNIPOLAR A/D) TO BE APPLIED TO THE SELECTED CHANNEL. A SERIES OF '1024' CONVERSIONS ARE TAKEN FOR EVERY VOLTAGE AND APPLICABLE GAIN SETTINGS AND THE AVERAGE IS COMPARED AGAINST THE TRUE VALUE FOR THAT SPECIFIED SETTING. IF THE AVERAGE IS MORE THAN + OR -1 COUNT FROM THE TRUE VALUE IT IS CONSIDERED IN ERROR AND THE CONVERSIONS RESULTS ARE TYPED OUT. AFTER TESTING ALL THE VOLTAGES AT THE SPECIFIED GAIN SETTINGS A TABLE OF THE RESULTS ARE TYPED OUT. WHEN THE COMPLETE TABLE HAS BEEN TYPED THE PROGRAM WILL REQUEST A NEW CH. TO BE TESTED.

302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317

318
319

-5-

B. STARTING PROCEDURE

1. TYPE 'G' TO RUN GAIN TEST.
2. THE MESSAGE "GAIN ACCURACY TEST. SUPPLY THE FOLLOWING VOLTAGES TO THE SELECTED CH. TYPE 'CR' TO START TEST."
3. A CH. AND A SPECIFIED VOLTAGE IS THEN REQUESTED.
4. TYPE 'CR' AFTER SUPPLYING THE REQUESTED VOLTAGE.

C. CONTROL SWITCHES (TELETYPE)

1. 1A (CONTROL A)
TYPING A '1A' WILL ENABLE THE GAIN TEST TO BE RESTARTED.
2. 1C (CONTROL C)
TYPE A 1C TO RETURN CONTROL TO THE MONITOR.
3. 1P (CONTROL P)
LOAD THE COMMAND DECODER (REFER TO SECTION 13.)

D. CONSOLE SWITCH SETTINGS FUNCTION

CONSOLE SW13=0	PRINT GAIN ERROR
CONSOLE SW13=1	INHIBIT TYPEOUT
CONSOLE SW14=0	LOOP ON GAIN ERROR
CONSOLE SW14=1	INHIBIT ERROR LOOPING

E. GAIN ERRORS

ON ENCOUNTERING AN ERROR (CONSOLE SWITCHES SET TO '0') THE ERROR HEADER AND ERROR DATA IS TYPED. THE TEST IS THEN LOOPED UNTIL EITHER THE CORRECT CONVERSION RESULTS ARE OBTAINED OR SW14 IS SET INHIBITING LOOPING.

1. ERROR FORMAT

GAIN	VOLTAGE	AVERAGE
A	B	C

WHERE:

A=GAIN SETTING
B=TRUE VOLTAGE VALUE
C=AVERAGE OF THE CONVERSION

2. GAIN CONVERSION TABLE (EXAMPLE OF AN '11' BIT BIPOLAR A/D)

GAIN	5.0000	2.50000	1.2500	0.6250	0.3125	0.1563	0.0781	0.0390
1	2000	1000	400	200	100			
2	----	2000	1000	400	200	100		
4	----	----	2000	1000	400	200	100	
8	----	----	----	2000	1000	400	200	100
16	776000	777000	777400	777600	777700			
32	-----	776000	777000	777400	777600	777700		

NO1

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 14
DZADMA.CMB

376
377
378
379

4
8

----- 776000 777000 777400 777600 777700
----- 776000 777000 777400 777600 777700

-6-

11. RECOVERY TEST

- A. THE "RECOVERY TEST" IS DESIGNED TO DETERMINE THE RECOVERY CAPABILITY OF THE ANALOG COMPONENTS IN THE 'ADF11'. THE TEST REQUESTS FOR TWO (2) CH. AND TWO GAIN INPUTS TO BE TYPED IN. THE TEST THEN TAKES A SERIES OF SIXTEEN (16) CONVERSIONS (8 ON EACH CH.) AND THEN TYPES OUT THE 'B' CONVERSION VALUES IN THE ORDER THEY WERE TAKEN ON THE SECOND CH.

B. STARTING SEQUENCE

1. TYPE 'E' TO RUN THE RECOVERY TEST.
2. A REQUEST IS THEN MADE FOR THE CH.S TO BE TESTED.
3. TYPE 'N,N (CR)' WHERE 'N' IS ANY DECIMAL CH.
4. A REQUEST FOR 'GAINS' IS THEN MADE.
5. TYPE 'N,N (CR)' WHERE 'N' WILL BE '1,2,4 OR 8.'
6. THE PROGRAM WILL THEN TAKE CONTINUOUS CONVERSIONS TYPING OUT THE CONVERSION VALUES FOR THE SECOND CH.

EXAMPLE:

CH. A XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX

WHERE:

A=TO THE SECOND CH.

X=TO THE 'B' CONVERSIONS TAKEN ON THAT CH.

C. CONTROL SWITCHES (TELETYPE)

1. 1A (CONTROL A)
TYPING A '1A' WILL ENABLE A NEW SET OF CH.S AND GAINS TO BE ENTERED.
2. 1C (CONTROL C)
TYPING A '1C' WILL ENABLE THE PROGRAM TO RETURN TO THE MONITOR.
3. 1P (CONTROL P)
LOAD THE COMMAND DECODER (REFER TO SECTION 13.)

D. CONSOLE SWITCH SETTINGS**FUNCTION**

CONSOLE SW 13=0
CONSOLE SW 13=1

PRINT CONVERSION VALUES
INHIBIT PRINTOUT

13. COMMAND DECODER

- A. THE 'COMMAND DECODER' IS USED TO ENABLE THE USER TO EXIT ANY SEQUENCE OF TESTING. HE CAN THEN CHANGE ANY ONE OR MORE OF THE PARAMETERS FOR THAT TEST AND CONTINUE THE TESTING SEQUENCE WITHOUT CHANGING ALL PARAMETERS.
- B. OPERATE INSTRUCTIONS
1. TYPE 'IP' TO ENTER THE COMMAND DECODER. THE PROGRAM WILL RESPOND VIA TYPING AN ASTRIC (*) TO INDICATE READY. THE USER CAN THEN CHANGE: CHANNEL(S), GAINS(S), COUNT SPREAD OR SYNC AT HIS DISCRETION.

C. COMMANDS

TYPE	PARAMETER AFFECTED
C (CR)	CHANNEL(S)
G (CR)	GAIN(S)
CS (CR)	COUNT SPREAD
S (CR)	SYNC
ST (CR)	EXIT AND CONTINUE TESTING

14. LISTING

-9-

X
TITLE ADF11A PART II, ANALOG DIAGNOSTIC TEST
ABS
;MAINDEC-11-DZADNA-A-0
;COPYRIGHT 23-JULY-75
;DIGITAL EQUIPMENT CORP. MAYNARD MASS. 01754
;PROGRAMMERS: EARL L. BOUSE/R.BALDWIN
;SWITCH REGISTER DEFINITIONS AND FUNCTIONS:

SW15=100000
SW14=40000
SW13=20000
SW12=10000
SW11=4000
SW10=2000
SW9=1000
SW8=400
SW7=200
SW6=100
SW5=40
SW4=20
SW3=10

100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010

473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

[illegible]

000004
000002
000001

S402=4
S401=2
S400=1

REGISTER DEFINITIONS

000000
000001
000002
000003
000004
000005
000006
000007

RD=X0
R1=X1
R2=X2
R3=X3
R4=X4
R5=X5
SP=X6
PC=X7

;GAIN EQUIVALENCE TABLE



G1=000
G2=20000
G4=40000
G8=60000

; INSTRUCTIONS DEFINITIONS

000004
015746
015726
010046
012500
021546
022526
000240

```
INTVC=X4  
PUSH1SP=5746  
POP1SP=5726  
PUSHRD=10046  
POP RD=12600  
PUSH2SP=24646  
POP2SP=22626  
NOP=240
```

```

; INTERRUPT VECTOR

```


;TRAP EQUIVALENCE TABLE:

834		104000	PRINT=ENT
835		104001	DECOCT=ENT+1
836		104002	ROMEM=ENT+2
837		104003	GAININ=ENT+3
838		104004	CMPUTE=ENT+4
839		104005	CATORIZ=ENT+5
840		104006	BINDEC=ENT+6
841		104007	SPACE=ENT+7
842		104010	PRTCT=ENT+10
843		104011	TTYIN=ENT+11
844		104012	WAITCM=ENT+12
845		104013	TAKECM=ENT+13
846		104014	PRTAVG=ENT+14
847		104015	SIXDSH=ENT+15
848		104016	TSTTKS=ENT+16
849		104017	GETCHA=ENT+17
850		104020	SAVEPG=ENT+20
851		104021	GETREG=ENT+21

;REGISTER ADDRESSES

852	001200	177776	PSW:	177776
853	001200	177560	TKS:	177560
854	001200	177562	TKB:	177562
855	001200	177564	TPS:	177564
856	001200	177566	TPB:	177566
857	001200	177570	SWR:	177570
858	001200	177571	SWRO:	177571
859	001200	164006	ADCR:	164006
860	001200	164010	ADCSR:	164010
861	001200	164000	ADSWR:	164000
862	001200	164004	ADWCR:	164004
863	001200	164012	ADDBR:	164012
864	001200	164002	ADWRA:	164002
865	001200	164014	ADWRB:	164014
866	001200	164016	ADADR:	164016
867	001236	000274	ADINT:	0274
868	001240	000276	ADLVL:	0276

```

;MESSAGE PRINTER ROUTINE
;DECIMAL TO OCTAL CONVERSION ROUTINE
;SUBROUTINE TO READ CATEGORIZE INC. MEM. VALUES
;ROUTINE TO REQUEST GAIN FROM TTY
;A/D AVERAGING ROUTINE
;ROUTINE TO CALCULATE THE COUNT SPREAD
;BINARY TO DECIMAL CONVERSION ROUTINE
;TYPE 'N' SPACES
;OCTAL PRINT ROUTINE
;TELETYPE INPUT ROUTINE
;GAIN TEST CONVERSION ROUTINE
;GAIN TEST CONVERSION ROUTINE
;GAIN AVERAGE PRINT ROUTINE
;SUBROUTINE TO TYPE OUT '6' DASHES
;SUBROUTINE TO TEST FOR KEYBOARD FLAG
;SUBROUTINE TO DECODE A CH(S) INPUT FROM TTY.
;SUBROUTINE TO SAVE REG.'S ON THE STACK
;SUBROUTINE TO GET REG.'S FROM THE STACK

```

```

;ADDRESS OF PROCESSOR STATUS REG.
;ADDRESS OF KEYBOARD STATUS REG.
;ADDRESS OF BUFFER
;PRINTER STATUS REG.
;PRINTER BUFFER REG.
;SWITCH REG.
;HIGH BYTE
;A/D CONTROL REG.
;A/D CONTROL & STATUS REG.
;A/D STATUS WORD REG.
;A/D WORD COUNT REG.
;A/D DATA BUFFER REG.
;A/D WORD REG 'A'
;A/D WORD REG 'B'
;ADDRESS OF A/D OFFSET REG.
;ADDRESS OF THE A/D INTERRUPT VECTOR
;ADDRESS OF THE A/D INTERRUPT LEVEL

```

```

874                                     ;*****
875                                     ;TEST INITIALIZATION ROUTINE. PROGRAM IS SELF STARTING TO THIS ROUTINE.
876                                     ;THE ROUTINE IS EXECUTED ON LOADING ONLY
877                                     ;*****
878
879 001242 016706 011530 INIT: MOV STACK,SP ;INIT STACK POINTER=1000
880 001246 012777 000340 177724 MOV #340,SPSW
881 001254 104000 PRINT ;CALL MESSAGE PRINTER VIA 'EMT'
882 001256 011001 TITLE ;TYPE PROGRAM HEADER.
883 001260 005067 011504 INIT1: CLR ADSIGN ;UNIPOLAR=0,BIPOLAR=1
884 001264 104000 PRINT
885 001266 011105 MES2 ;REQUEST THE A/D LENGTH
886 001270 104001 DECOCT ;CONVERT A/D LENGTH TO OCTAL
887 001272 012701 003776 MOV #3776,R1 ;INIT AS 'INC MEM' OFFSET
888 001276 012702 001000 MOV #1000,R2 ;= TO +5V VALUE FOR 10 BITS
889 001302 162767 000012 004554 SUB #12,BCDTAB ;A/D LENGTH = TO 10 BITS?
890 001310 001414 BEQ LDSize ;YES, EXIT
891 001312 012703 000005 MOV #5,R3 ;NO, TEST UP TO 15 BITS
892 001316 006301 SIZE: ASL R1 ;BUMP MEM. OFFSET
893 001320 006302 ASL R2 ;ALSO A/D SIZE
894 001324 005367 004536 DEC BCDTAB ;DECREMENT COUNT
895 001328 001405 BEQ LDSize ;EXIT IF DONE
896 001330 005303 DEC R3
897 001332 100371 BPL SIZE ;BRANCH UNTIL 15 IS REACHED
898 001334 104000 PRINT ;ILLEGAL ENTRY
899 001336 011463 GMARK ;PRINT '?'
900 001340 000747 BR INIT1 ;RETRY
901 001342 012700 013172 LDSize: MOV #POS500,R0 ;LOAD +5V VALUE
902 001346 010203 MOV R2,R3 ;SAVE AD SIZE
903 001350 010320 LDPOS: MOV R3,(R0)+
904 001352 006203 ASR R3
905 001354 022700 013204 CMP #NEG500,R0 ;LOADED ALL POS. VALUES?
906 001358 001373 BNE LDPOS ;BRANCH IF NO
907 001362 010203 MOV R2,R3 ;RESET +5V VALUE
908 001364 005403 NEG R3
909 001366 010320 LDNEG: MOV R3,(R0)+
910 001370 006203 ASR R3
911 001372 022700 013216 CMP #NEG312+2,R0 ;LOADED ALL NEG. VALUES?
912 001376 001373 BNE LDNEG ;BRANCH IF NO.
913 001380 005767 011364 TST ADSIGN ;TEST FOR SIGN BIT
914 001384 001401 BEQ CORSIZ ;BRANCH IF NOT SET
915 001386 006301 ASL R1 ;OTHERWISE ADD 1 BIT TO CONVERTER LENGTH.
916 001390 052701 000776 CORSIZ: BIS #776,R1 ;SET ALL POSSIBLE SHIFTED BITS
917 001394 012737 001476 000004 MOV #INITA,2#4 ;INITIAL THE TIME OUT ADDRESS
918 001398 012737 000340 000006 MOV #340,2#6
919 001402 005067 011350 CLR INCFLG ;CLR INCREMENT MEMORY FLAG
920 001406 012767 020000 011346 MOV #20000,OFFSET ;INC MEM STARTS @ 100 FOR <13BITS+++
921 001410 032701 020000 BIT #20000,R1 ;TEST FOR 13 BIT CONVERTER+++
922 001414 001405 BEQ .+14 ;BRANCH IF NOT+++
923 001418 062701 020000 ADD #20000,R1 ;ADD AN ADDITIONAL 4K OFFSET FOR 13+++
924 001422 012767 040000 011326 MOV #40000,OFFSET ;13 BITS INC MEM STARTS AT +++
925 001426 062701 020000 ADD #20000,R1 ;ADD 4K OFFSET TO A/D LENGTH+++
926 001430 005711 TST 2#1 ;TEST IF MEMORY IS AVAIL, 3LE+++
927 001434 012767 000377 011306 MOV #377,INCFLG ;YES SETTINGS FOR INCREMENT MEMORY

```



```

981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030

```

001730	012767	001756	011042	CALBRT:	MOV	#CALBT1,AVECTR	;SET UP '1A' RESTART ADDRESS
001736	012767	002004	011036		MOV	#CALBT2,PVECTR	;SET UP '1P' START ADDRESS.
001744	104000				PRINT		
001746	011513				MES7		;TYPE TEST HERE OR
001750	104000				PRINT		
001752	011157				MES10		;TEXT 'SYNC I OR E'
001754	000432				ER	CALB1A	;WAIT FOR INPUT
001756	104300			CALBT1:	PRINT		
001760	012441				MES39		;TEST 'SYNC'
001762	104011			CALB1A:	TTYIN		;WAIT FOR INPUT.
001764	016767	003700	011024		MOV	INBUF,PROC	;SAVE IT IN TEMP STORAGE
001772	012767	000011	011050		MOV	#11,KSTOR3	;PRINT '9' CONVERSIONS/LINE
001774	104300				PRINT		
001776	011453				CRLF		
001778	012767	000001	011012	CALBT2:	MOV	#1,ICOUNT	;SETUP TO PRINT '1' VALUE
001780	017767	177174	011016		MOV	#2,FINAL	;GET CH. FROM SW. REG.
001782	042767	177000	011010		BIC	#177000,FINAL	;CLR UNWANTED BITS
001784	017767	177160	011000		MOV	#2,INITAL	;ALSO GET AS INITIAL CH. & GAIN
001786	042767	107000	010772		BIC	#107000,INITAL	;CLR UNWANTED BITS
001788	052767	110000	010764		BIS	#110000,INITAL	;SELECT SEQ. DMA, INITAL
001790	017767	177136	010766		MOV	#2,R,KSTOR1	;GET DELAY BITS 9-11
001792	000367	010762			STB	KSTOR1	
001794	006267	010756			RAR	KSTOR1	
001796	042767	177770	010750		BIC	#177770,KSTOR1	;DELAY NOW SET
001798	017767	177112	010744		MOV	#2,SWR,KSTOR2	;SAVE ORIGINAL SWITCH SETTING.
001800	022767	000105	010706		CMR	#105,PROC	;TEST SYNC SELECT
001802	001003				BNE	CALB1A	;BRANCH IF NOT 'E'
001804	052767	004000	010714		BIS	#400,INITAL	;SET 'EXT' SYNC ENABLE
001806	012777	177777	177076	CALB2A:	MOV	#-1,ADWCR	;SET UP FOR '1' CONVERSION
001808	104016				TSTTKS		
001810	004767	003740			JSR	PC,ADCMVT	;TAKE AND STORE THE CONVERSIONS
001812	005777	177052			TST	#2,SWR	;TEST FOR SW15 TO PRINT
001814	100015				BPL	CALB1A	;BRANCH IF NOT SET.
001816	012767	013434	004766		MOV	BALWTF,AVGTAB	;SET UP TO PRINT VALUE
001818	104014				PRTAVG		;PRINT IT
001820	005367	010672			DEC	KSTOR3	
001822	001015				BNE	CALBT4	
001824	012767	000011	010662		MOV	#11,KSTOR3	
001826	104000				PRINT		
001828	011453				CRLF		
001830	000407				BR	CALBT4	;TEST FOR LOOP

```

*****
;CALIBRATION ROUTINE
*****

```

```

;ROUTINE REQUESTS THE TYPE OF 'SYNC' TO BE USED ('I' INTERNAL OR 'E' EXTERNAL).
;THE PROGRAM THEN TAKES CONTINUOUS CONVERSIONS (SEQ. DMA MODE) USING DATA
;SW'S 0-8 TO SELECT THE CH. SW'S 13&14 TO SELECT GAIN & 0 EITHER
;SW'S 9-11 TO SELECT DELAY OR SW '15' TO PRINT THE CONVERSION VALUE.

```

C03

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 29
DZADHA.CM3

1031	002174	016700	011234	CALB28: MOV	ADBUFF, R0	; SET UP A/D BUFFER.
1032	002175	016701	010640		KSTOR1, R1	; SET UP DELAY (RESET COUNT)
1033	002176	000005		CALBT3: RESET		
1034	002206	005301		DEC	R1	; DECREMENT DELAY
1035	002210	100375		BPL	CALBT3	
1036	002212	104016		CALBT4: TSTKS		; TEST FOR KEYBOARD INTERRUPT
1037	002214	026777	010626 176770	CMP	KSTOR2, 2SWR	; TEST IF SWITCH REGISTER HAS CHANGED
1038	002222	001736		BEG	CALB2A	; BRANCH AND TAKE NEXT CONVERSION
1039	002224	000667		BR	CALBT2	; YES, COMPUTE NEW INPUT

```

1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088

```

002226	012767	002240	010544	REPTST:	MOV	AREPT1,AVECTR	;SET UP CNTR 'A' VECTOR ADDRESS
002234	104000				PRINT		
002236	011640				MES13		;TEXT 'REPEATIBILITY TEST'
002240	005067	010546		REPT1:	CLR	SOFLAG	;CLR SOFTWARE FLAG
002244	005067	010564			CLR	INITAL	;CLR CH. STORAGE REG.
002250	005067	010620			CLR	MESPR1	;CLR PRINT SW.
002254	012767	002350	010520		MOV	AREPT2,PVECTR	;SET UP 'P' VECTOR ADDRESS
002258	005767	010516			TST	INCFLG	;TEST IF FLAG IS SET
002262	001420				BEQ	REPT1A	;BRANCH IF NO
002270	104000				PRINT		;OTHERWISE GIVE INC. MEM. OPTION
002272	011466				MES6		;TEXT 'INC MEM. (Y OR N)?'
002274	104011				TTYIN		;WAIT FOR REPLY
002276	022767	000131	003364		CMP	0131,INBUF	;WAS 'Y' TYPED?
002304	001011				BNE	REPT1A	;BRANCH IF NO
002306	012767	000377	010476		MOV	0377,SOFLAG	;YES, SET SOFTWARE FLAG.
002314	052767	002000	010512		BIS	02000,INITAL	;SET INCREMENT MEMORY BIT
002322	016777	010462	176704		MOV	OFFSET,ADADR	;LOAD THE OFFSET REG.+++++
002330	104017			REPT1A:	GETCHA		;REQUEST & STORE CH(S)
002332	104003				GAININ		;SET UP GAIN
002334	104000				PRINT		
002336	011701				MES16		;TEXT 'COUNT SPREAD ''
002340	104001				DECOCT		;DECODE TO OCTAL
002342	016767	003516	010500		MOV	BCDTAB,KSTOR3	;SAVE IT
002350	016767	010462	010466	REPT2:	MOV	FINAL,KSTOR1	;SAVE STARTING CH.
002356	052767	110000	010450		BIS	0110000,INITAL	;SELECT: SEQ. DMA
002364	016767	010454	010444	REPT2A:	MOV	KSTOR1,FINAL	;RESET FINAL ADDRESS
002372	042767	001777	010434		BIC	01777,INITAL	
002400	056767	010432	010426		BIS	FINAL,INITAL	;RESET THE INIAL CH.
002406	016767	010432	010436		MOV	KSTOR1,KSTOR4	;SAVE STARTING CH.
002414	004767	002706		REPT3:	JSR	PC,CLACOR	;CLR MEM FOR 'INC'
002420	012777	176000	176576		MOV	0-2000,ADWCR	;SET FOR '1024' CONVERSIONS
002426	104016				TSTTKS		
002430	004767	003440			JSR	PC,ADCHVT	;TAKE THE CONVERSIONS
002434	005767	010352			TST	SOFLAG	;INC. MEM?

E03

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 31
DZADHA.CMB

1079	002440	001402			BEQ	REPT3A		;NO, EBA (EARL BOUSE ARITHMETIC)
1080	002442	104002			RD MEM			;YES, READ MEMORY & COLLECT VALUES
1091	002444	000402			BR	REPT3B		
1092	002446	104004			REPT3A: CHPUT			;AVERAGE & COMPUTE DISTRIBUTION
1093	002450	104005			CATOR12			
1094	002452	032777	010000	176532	REPT3B: BIT	BSW12,2SWR		;TEST DATA SW12
1095	002460	001025			BNE	PCPT4		;IF SET, FORCE TYPE OUT
1096	002462	032777	020000	176522	TSTCT4: BIT	BSW13,2SWR		;TEST FOR INHIBIT TYPEOUT
1097	002470	001073			BNE	PCPT7		;BRANCH IF SW SET
1098	002472	012700	000001		MOV	81, R0		
1099	002476	012701	013162		MOV	8XSP1, R1		
1100	002502	016702	010342		MOV	KST01, R2		
1101	002506	020002			TSTCNT: CMP	R0, R1		
1102	002510	001406			BEQ	CHKCNT		
1103	002512	002700			INC	R0		
1104	002514	005721			TST	(R1)+		;UPDATE COUNT ADDRESS
1105	002516	020127	013172		CMP	R1, 8XSPR04+2		;CHECKED '1-4'?
1106	002520	001371			BNE	TSTCNT		;NO
1107	002522	000403			BR	REPT4		;ILLEGAL ENTRY, TYPE OUT COUNTS
1108	002524	022711	002000		CHKCNT: CMP	8'000, (R1)		;ARE ALL COUNTS IN COUNT SPREAD?
1109	002526	001452			BEQ	REPT7		;YES, EXIT
1110	002530	104130			REPT4: PRINT			
1111	002532	011453			CALL			
1112	002534	005767	010330		TST	NESPT		;TEST IF HEADER HAS BEEN TYPED
1113	002536	001002			BNE	REPT5		;BRANCH IF YES
1114	002538	104000			PRINT			
1115	002550	011732			NES19			;TEXT 'CH. HIGH AVG. LOW'

1116	002552	104000		REPT5:	PRINT		
1117	002554	011453			CRLF		;CARRIAGE RETURN, LINE FEED
1118	002556	016702	010270		MOV	KSTOR4,R2	;MOV. CH.
1119	002558	104006			BINDEC		;CONVERT TO DECIMAL AND PRINT
1120	002560	104007			SPACE		
1121	002562	104010			PRTCT		;PRINT LOW VALUE
1122	002564	013100			LOW		
1123	002566	104007			SPACE		
1124	002568	104010			PRTCT		;PRINT AVERAGE VALUE
1125	002570	013114			AVERAGE		
1126	002572	104007			SPACE		
1127	002574	104010			PRTCT		;PRINT HIGH VALUE
1128	002576	013114			HIGH		
1129	002578	005767	010262		TST	NESPR	
1130	002580	001002			BNE	REPT6	
1131	002582	104000			PRINT		
1132	002584	011764			NES20		;PRINT 'COUNT SPREAD' HEADER
1133	002586	005767	000007	010246	FIS	#7,NESPR	;INHIBIT OTHER HEADERS
1134	002588	005767	002000	010310	REP	#2000,AVGCNT	;TEST IF ALL COUNTS WERE AT AVG.
1135	002590	001411			REQ	REPT7	;BRANCH TO NEXT CH. IF YES.
1136	002592	104000			PRINT		
1137	002594	011453			CRLF		
1138	002596	012704	013130		REP	BORLOW,R4	
1139	002598	012704			REP	(R4)+,R2	
1140	002600	104006			REP		;TYPE OUT COUNT SPREAD
1141	002602	022704	013162		REP	EXSPR01,R4	;TEST FOR DONE
1142	002604	001373			REP	REPT6A	;BRANCH IF NO AND TYPE NEXT COUNT
1143	002606	005767	010150		REP	INITAL	
1144	002608	005767	010146		REP	FINAL	
1145	002610	005767	010156		REP	KSTOR4	;INCREMENT 'CH.'
1146	002612	005767	010142	010150	REP	FINAL2,KSTOR4	;TESTED ALL CH.(S)?
1147	002614	005767			REP	REPT3	;BRANCH IF NO AND TEST NEXT CH.
1148	002616	005767			REP	REPT2A	;OTHERWISE RESET AND REPEAT

```

1149
1150
1151 *****
1152 GAIN ACCURACY TEST
1153 *****
1154
1155 THE GAIN ACCURACY TEST REQUESTS FOR A DECIMAL CH. TO BE TYPED IN VIA
1156 THE OPERATOR. IT THEN REQUESTS '16' SPECIFIED VOLTAGES TO BE APPLIED TO
1157 THAT CHANNEL. AFTER SUPPLY THE REQUESTED VOLTAGE, THE OPERATOR TYPES A
1158 SPACE ON THE TELETYPE AND A SERIES OF '1024' CONVERSIONS ARE THEN TAKEN
1159 AT SPECIFIED GAIN SETTINGS AND AVERAGED OUT. IF THE AVERAGED VALUE
1160 IS FOUND TO BE MORE THAN + OR -1 COUNT FROM THE INPUT VOLTAGES TRUE
1161 VALUE, IT IS CONSIDERED IN ERROR AND THE ERROR VOLTAGE, THE EXPECTED VALUE
1162 AND THE GAIN ARE TYPED OUT. AFTER COMPUTING INPUT VOLTAGE AVERAGES, A
1163 TABLE OF ALL CONVERSION RESULTS ARE TYPED OUT.
1164
1165 002706 012767 002720 010064 GAIN: MOV      %GT,AVECTR      ;SET UP '1A' RETURN ADDRESS
1166 002714 104000                PRINT
1167 002716 011123                MES3
1168 002720 005067 010110        GT:      CLR      INITIAL      ;TEXT GAIN ACCURACY TEST
1169 002724 012767 002734 010050 GTD:     MOV      %GTO+2,PVECTR    ;SET UP '1P' VECTOR ADDRESS
1170 002732 104017                GETCHA    ;REQUEST CH.
1171 002734 052767 110000 010072 BIS      %110000,INITAL    ;SEQ. DMA, INITIAL
1172 002742 005067 010126        CLR      MESPR1    ;CLR PRINT INHIBIT SWITCH
1173
1174 ;TEST +5.0V X G1
1175
1176 002746 104000        GT1:     PRINT
1177 002750 012115        MES23
1178 002752 104012        WAITGN
1179 002754 000000        G1
1180 002756 013172        POSS00
1181 002760 013216        GFSOX1
1182 002762 005767 010002        TST      ADSIGN
1183 002766 001406        BEQ      GT2
1184
1185 ;TEXT '+5.00V'
1186 ;CALL THE GAIN 'WAIT' HANDLER
1187 ;GAIN X1
1188 ;SAVE VALUE
1189 ;BRANCH TO NEXT TEST IF UNIPOLAR.
1190
1191 ;TEST -5.0V X G1
1192
1193 002770 104000        PRINT
1194 002772 012124        MES24
1195 002774 104012        WAITGN
1196 002776 000000        G1
1197 002780 013204        NEG500
1198 002782 013266        GFSOX1
1199
1200 ;TEXT 'SWITCH VOLTAGE NEG.'
1201 ;GAIN X1
1202 ;SHOULD=-5.0V
1203 ;SAVE VALUE
1204
1205 ;TEST +2.5V X G1
1206
1207 003004 104000        GT2:     PRINT
1208 003006 012141        MES25
1209 003010 104012        WAITGN
1210 003012 000000        G1
1211 003014 013174        POSS50
1212 003016 013220        GFSOX1
1213
1214 ;TEXT '+2.5V'
1215 ;GAIN X1
1216 ;SAVE VALUE

```


1254			;TEST -1.25V X G2	
1255	003140	104013	TAKEGN	
1256	003142	020000	G2	;GAIN X2
1257	003144	013106	NEG250	
1258	003146	013302	GM12X2	;SAVE VALUE
1259				
1260			;TEST -1.25V X G4	
1261	003150	104013	TAKEGN	
1262	003152	040000	G4	;GAIN X4
1263	003154	013204	NEG500	;SHOULD = 5.0V
1264	003156	013312	GM12X4	;SAVE VALUE
1265				
1266			;TEST +0.625V X G1	
1267	003160	104000	GT4: PRINT	
1268	003162	012157	MES27	;TEXT '+0.625V'
1269	003164	104012	WAITGN	
1270	003166	000000	G1	
1271	003170	010000	POS625	;SHOULD = +0.625V
1272	003172	013124	GP62X1	;SAVE VALUE
1273				
1274			;TEST +0.625V X G2	
1275	003174	104013	TAKEGN	
1276	003176	020000	G2	;GAIN X2
1277	003178	013176	POS125	;SHOULD = +1.25V
1278	003180	013234	GP62X2	;SAVE IT
1279				
1280			;TEST +0.625V X G4	
1281	003204	104013	TAKEGN	
1282	003206	040000	G4	;GAIN X4
1283	003208	013174	POS250	;SHOULD = +2.5V
1284	003210	013244	GP62X4	;SAVE IT
1285				
1286			;TEST +0.625V X G8	
1287	003214	104013	TAKEGN	
1288	003216	060000	G8	;GAIN X8
1289	003218	013172	POS500	;SHOULD = +5.00V
1290	003220	010054	GP62X8	;SAVE IT
1291	003222	000000	TST	
1292	003224	000000	BEQ	ADSIGN
1293	003226	001422	GT5	;BRANCH IF UNIPOLAR
1294				
1295			;TEST -0.625V X G1	
1296	003232	104000	PRINT	
1297	003234	012124	MES24	;SWITCH VOLTAGE NEG.
1298	003236	104012	WAITGN	
1299	003238	000000	G1	;GAIN X1
1300	003240	013212	NEG625	;SHOULD = -0.625V
1301	003242	013274	GP62X1	
1302				
1303			;TEST -0.625V X G2	
1304	003246	104013	TAKEGN	
1305	003248	020000	G2	;GAIN X2
1306	003250	013210	NEG125	;SHOULD = -1.25V
1307	003252	013304	GP62X2	;SAVE IT

007540

J03

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 36
 DZADHA.CMB

1308			;TEST -0.625V X G4	
1309	003256	104013	TAKEGN	
1310	003258	040000	G4	;GAIN X4
1311	003262	013206	NEG250	;SHOULD = -2.5V
1312	003264	013314	GM62X4	
1313				
1314			;TEST -0.625V X G8	
1315	003266	104013	TAKEGN	
1316	003270	060000	G8	;GAIN X8
1317	003272	013204	NEG500	;SHOULD = -5.00V
1318	003274	013324	GM62X8	
1319				
1320			;TEST +0.3125V X G1	
1321	003276	104000	GT5: PRINT	
1322	003300	012167	MES28	;TEXT '+0.3125V'
1323	003302	104012	WAITGN	
1324	003304	000100	G1	;GAIN X1
1325	003306	013112	POS312	;SHOULD = +0.3125V
1326	003310	013226	GP31X1	;SAVE IT
1327				
1328			;TEST +0.3125V X G2	
1329	003312	104013	TAKEGN	
1330	003314	020000	G2	;GAIN X2
1331	003316	013200	POS625	;SHOULD = +0.625V
1332	003320	013236	GP31X2	
1333				
1334			;TEST +0.3125V X G4	
1335	003322	104013	TAKEGN	
1336	003324	040000	G4	;GAIN X4
1337	003326	013176	POS125	;SHOULD = +1.25V
1338	003330	013246	GP31X4	
1339				
1340			;TEST +0.3125V X G8	
1341	003332	104013	TAKEGN	
1342	003334	060000	G8	;GAIN X8
1343	003336	013174	POS250	;SHOULD = +2.50V
1344	003340	013256	GP31X8	
1345	003342	005767	TST	
1346	003346	001422	BE0	;BRANCH IS UNIPOLAR
1347				
1348			;TEST -0.3125V X G1	
1349	003350	104000	PRINT	
1350	003352	012124	MES24	;TEXT 'SWITCH NEG.'
1351	003354	104012	WAITGN	
1352	003356	000000	G1	;GAIN X1
1353	003360	013214	NEG312	;SHOULD = -0.3125V
1354	003362	013276	GP31X1	
1355				
1356			;TEST -0.3125V X G2	
1357	003364	104013	TAKEGN	
1358	003366	020000	G2	;GAIN X2
1359	003370	013212	NEG625	;SHOULD = -0.625V
1360	003372	013306	GM31X2	
1361				

007422

ADSIGN
GT6

K03

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 37
 DZADHA.CMB

1362			;TEST -0.3125V X G4	
1363	003374	104013	TAKEGN	
1364	003376	040000	G4	;GAIN X4
1365	003400	013210	NEG125	;SHOULD = -1.25V
1366	003402	013316	GM31X4	
1367				
1368			;TEST -0.3125V X G8	
1369	003404	104013	TAKEGN	
1370	003406	060000	G8	;GAIN X8
1371	003410	013206	NEG250	;SHOULD = -2.50V
1372	003412	013326	GM31X8	
1373				
1374			;TEST +0.1563V X G2	
1375	003414	104000	GT6: PRINT	
1376	003416	012200	MES29	;TEXT '+0.1563V'
1377	003420	104012	WAITGN	
1378	003422	020000	G2	;GAIN X2
1379	003424	013212	POS312	;SHOULD = +0.3125V
1380	003426	013240	GP15X2	
1381				
1382			;TEST +0.1563V X G4	
1383	003430	104013	TAKEGN	
1384	003432	040000	G4	;GAIN X4
1385	003434	013200	POS625	;SHOULD = +0.625V
1386	003436	013250	GP15X4	
1387				
1388			;TEST +0.1563V X G8	
1389	003440	104013	TAKEGN	
1390	003442	060000	G8	;GAIN X8
1391	003444	013176	POS125	;SHOULD = +1.25V
1392	003446	013210	GP15X8	
1393	003450	005767	TST	ADSIGN
1394	003454	001416	BC0	GT7
1395				;BRANCH IF UNIPOLAR
1396				
1397			;TEST -0.1563V X G2	
1398	003456	104000	PRINT	
1399	003460	012124	MES24	;TEXT 'SWITCH NEG.'
1400	003462	104012	WAITGN	
1401	003464	020000	G2	;GAIN 'X2'
1402	003466	013214	NEG312	;SHOULD = -0.3125V
1403	003470	013310	GM15X2	
1404				
1405			;TEST -0.1563V X G4	
1406	003472	104013	TAKEGN	
1407	003474	040000	G4	;GAIN X4
1408	003476	013212	NEG625	;SHOULD = -0.625V
1409	003500	013320	GM15X4	
1410				
1411			;TEST -0.1563V X G8	
1412	003502	104013	TAKEGN	
1413	003504	060000	G8	;GAIN X8
1414	003506	013210	NEG125	;SHOULD = -1.25V
1415	003510	013330	GM15X8	

007314

1416				;TEST +0.0781V X G4	
1417	003512	104000		GT7: PRINT	
1418	003514	012211		MES30	;TEXT '+0.0781V'
1419	003516	104012		WAITGN	
1420	003520	040000		G4	;GAIN X4
1421	003522	013202		POS312	;SHOULD = +0.3125V
1422	003524	013252		GP07X4	
1423					
1424					
1425	003526	104013		;TEST +0.0781V X B	
1426	003530	060000		TAKEGN	
1427	003532	013200		G8	;GAIN X8
1428	003534	013262		POS625	;SHOULD = +0.625V
1429	003536	005767	007226	GP07X8	
1430	003542	001412		TST	ADSIGN
1431				BEQ	GT8
1432					
1433	003544	104000		;TEST -0.0781V X G4	
1434	003546	012124		PRINT	
1435	003550	104012		MES24	;TEXT 'SWITCH NEG.'
1436	003552	040000		WAITGN	
1437	003554	013214		G4	;GAIN X4
1438	003556	013222		NEG312	;SHOULD = -0.3125V
1439				GM07X4	
1440					
1441	003560	104013		;TEST -0.0781V X G8	
1442	003562	060000		TAKEGN	
1443	003564	013212		G8	;GAIN X8
1444	003566	013332		NEG625	;SHOULD = -0.625V
1445				GM07X8	
1446					
1447	003570	104000		;TEST +0.0390V X G8	
1448	003572	012122		GT8: PRINT	
1449	003574	104012		MES31	;TEXT '+0.0390V'
1450	003576	060000		WAITGN	
1451	003577	013202		G8	;GAIN X8
1452	003578	013254		POS312	;SHOULD = +0.3125V
1453	003579	005767	007160	GP03X8	
1454	003580	001406		TST	ADSIGN
1455				BEQ	GT9
1456					
1457	003612	104000		;TEST -0.0390V X G8	
1458	003614	012124		PRINT	
1459	003616	104012		MES24	;TEXT 'SWITCH NEG.'
1460	003620	060000		WAITGN	
1461	003622	013214		G8	
1462	003624	013334		NEG312	;SHOULD = -0.3125V
				GM03X8	

```

1463
1464
1465
1466 003626 012767 013216 003302 GT9: MOV #GP50X1 AVGTAB ;SET UP GAIN TABLE
1467 003634 012767 000002 007202 MOV #2,KSTOR1
1468 003642 104000 PRINT
1469 003644 012233 MES32 ;TYPE TABLE 'HEADER'
1470 003646 012767 000005 007150 GT10: MOV #5,ICOUNT ;SET UP PRINT ROUTINE
1471 003654 104000 PRINT
1472 003656 012370 MES34 ;TYPE GAIN X1 VALUES
1473 003660 104014 PRTAVG ;TYPE OUT AVERAGES X1
1474 003662 104000 PRINT
1475 003664 012400 MES35 ;TYPE GAIN X2
1476 003666 012767 000001 007126 MOV #1,COUNT
1477 003674 104015 SIXDSH ;TYPE DASHES
1478 003676 104014 PRTAVG ;TYPE OUT AVERAGES X2
1479 003700 104000 PRINT
1480 003702 012410 MES36
1481 003704 012767 000002 007110 MOV #2,COUNT
1482 003712 104015 SIXDSH ;TYPE DASHES
1483 003714 104014 PRTAVG ;TYPE OUT AVERAGES X4
1484 003716 104000 PRINT
1485 003720 012420 MES37
1486 003722 012767 000003 007072 MOV #3,COUNT
1487 003730 104015 SIXDSH
1488 003732 104014 PRTAVG ;TYPE OUT AVERAGES X8
1489 003734 005767 007030 TST ADSIGN
1490 003740 001002 BNE GT11 ;IF UNIPOLAR, TYPE OUT NEG. COUNTS
1491 003742 000167 176764 JMP GTO ;OTHERWISE RESTART GAIN TEST
1492 003746 005367 007072 GT11: DEC KSTOR1
1493 003752 001002 BNE .+6
1494 003754 000167 176752 JMP GTO
1495 003760 104000 PRINT
1496 003762 011453 CRLF
1497 003764 000733 ER GT10
1498 003766 104011 XWATCH: TTYIN ;WAIT FOR 'CR' BEFORE CONTINUING
1499 003770 005067 007076 CLR RETSWH ;CLR SOFTWARE SW.
1500 003774 042767 060000 007032 BIC #6000,INITAL ;CLR GAIN BITS
1501 004002 057667 000000 007024 BIS @2(SP),INITAL ;GET SPECIFIED GAIN AND SET IT UP.
1502 004010 017667 000000 007026 MOV @2(SP),KSTOR1 ;SAVE GAIN
1503 004016 062716 000002 ADD #2,(SP) ;SET UP THE ADDRESS OF TRUE VOLTAGE
1504 004022 017667 000000 000226 MOV @2(SP),PRTADR ;SAVE ADDRESS OF TRUE VOLTAGE
1505 004030 062716 000002 ADD #2,(SP) ;SET UP STORAGE ADDRESS FOR VOLTAGE
1506 004034 017667 000000 007004 MOV @2(SP),KSTOR2 ;SAVE ADDRESS
1507 004042 062716 000002 ADD #2,(SP) ;SET UP STACK TO EXIT
1508 004046 104016 GLOOP: TSTTKS ;TEST FOR KEYBOARD FLAG
1509 004050 012777 176000 175146 MOV #-2000,ADWCR ;SET UP TO TAKE '1024' CONVERSIONS
1510 004056 004767 002012 JSR PC,ADCHVT ;TAKE THE CONVERSIONS
1511 004062 104004 CMPUTE ;COMPUTE THE AVERAGE
1512 004064 016777 007024 006774 MOV AVERAGE,@KSTOR2 ;SAVE THE AVERAGE VALUE
1513 004072 104005 CATORIZ ;CATORIZE THE COUNT SPREAD
1514 004074 026777 007014 000154 CMP AVERAGE,@PRTADR ;IS AVERAGE =TO KNOWN VALUE?
1515 004102 001414 BEQ GANEXT ;EXIT IF EQUAL
1516 004104 026777 007006 000144 CMP AVERAGE,@PRTADR ;IS AVERAGE =TO KNOWN VALUE +1?
1517 004112 001410 BEQ GANEXT ;EXIT IF EQUAL
1518 004114 027767 000136 006770 CMP @PRTADR,AVERM1 ;IS AVERAGE =TO KNOWN VALUE -1?

```

Address	Hex	Label	Operation	Comment
1519	004122	001404	BEO	GANEXT
1520	004124	032777	BIT	#SW14,JSWR
1521	004132	001406	BEO	GERR1
1522	004134	005767	006732	GANEXT: TST
1523	004140	001402	BEO	RETSWH
1524	004142	104000	PRINT	.+6
1525	004144	011453	CRLF	
1526	004146	000002	RTI	
1527	004150	032777	020000	175034 GERR1: BIT
1528	004156	001333	BNE	#SW13,JSWR
1529	004160	005767	006710	GLOOP
1530	004164	001005	TST	MESPRT
1531	004166	052767	000001	006700 BNE
1532	004174	104000	BIS	GERR2
1533	004176	012340	PRINT	#1,MESPRT
1534	004200	005767	006640	GERR2: TST
1535	004204	001003	BNE	KSTOR1
1536	004206	104000	PRINT	GERR3
1537	004210	012340	MES33	
1538	004212	001003	BR	
1539	004214	022		

```

1561 ;*****
1562 ;RECOVERY TEST
1563 ;*****
1564
1565 ;THIS TEST IS DESIGNED TO TEST RECOVERY OF THE A/D CONVERTER VIA ACCEPT-
1566 ;ING TWO (2) CHANNEL AND GAIN INPUTS FROM THE TELETYPE AND THEN TAKE A
1567 ;SERIES OF EIGHT CONVERSIONS ON EACH CHANNEL AND TYPING OUT THE CON-
1568 ;VERSION VALUES OF THE 2ND CHANNEL IN THE ORDER THEY WERE TAKEN.
1569
1570 004276 012767 004310 006474 RECVY: MOV #RECVY1,AVECTR ;SET UP THE '1A' RETURN ADDRESS
1571 004304 104000 PRINT
1572 004306 011540 P SR ;TEXT 'RECOVERY TEST'
1573 004310 012767 000010 006506 RECVY1: MOV #10,ICOUNT ;SET UP TO PRINT '8' VALUES
1574 004316 005067 006512 CLR INITIAL ;CLR CH. TEMP. STORAGE
1575 004322 005067 006512 CLR INITL2 ;CLR 2ND CH. STORAGE
1576 004326 012767 004354 006446 MOV #RECVY2,PVECTR
1577 004334 052767 010000 006472 BIS #10000,INITAL ;SELECT RANDOM,DMA
1578 004342 052767 010000 006470 BIS #10000,INITL2
1579 004350 104017 GETCHA ;REQUEST CHANNELS
1580 004352 104003 GAININ ;GET THE GAIN SETTINGS
1581 004354 012767 000020 006440 RECVY2: MOV #20,COUNT ;TAKE '16' CONVERSIONS
1582 004362 012701 013336 MOV #RANBUF,R1 ;SET UP RANDOM TABLE
1583 004366 012702 000010 MOV #10,R2 ;SAVE 8 VALUES
1584 004372 016721 006436 RECVY3: MOV INITIAL,(R1)+
1585 004376 016721 006436 DEC R2
1586 004400 001374 BNE RECVY3
1587 004402 012702 000015 MOV #15,R2
1588 004406 016721 006426 RECVY4: MOV INITL2,(R1)+
1589 004412 005302 DEC R2
1590 004414 100374 BPL RECVY4
1591 004416 104016 RECVY5: TSTTKS ;CHECK FOR KEYBOARD FLAG
1592 004420 012777 177760 174576 MOV #20,RANMCR ;SET M.C FOR '16' CONVERSIONS
1593 004426 004767 001442 JSR PC,ADCVT ;TAKE THE CONVERSIONS
1594 004432 032777 020000 174552 BIT #SW13,ASMR ;TEST THE PRINT INHIBIT SW
1595 0
```

```

1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618 004470 012777 000340 174502 INCTST: MOV      8340, 2PSW
1619 004476 016706 006274          MOV      STACK, SP
1620 004482 000000 000000          CLR      SFLAG          ;CLEAR PRINT FLAG
1621 004488 016707 000000          MOV      812A, PVECTR          ;SET UP A CONTROL P START
1622 004494 012767 004544 006256 INCTSB: MOV      812TSA, AVECTR          ;SET UP A RESTART ADDRESS
1623 004500 005767 006256          TST      INCFLG          ;TEST IF CORE AVAIL
1624 004506 001104          BNE      IS
1625 004512 016707 000000          PRINT   MES42
1626 004518 016707 000000          J/P      MONITR          ;REPORT INSUFFICIENT CORE
1627 004524 000167 175026          IS:      PRINT   MES44          ;RETURN TO MONITOR
1628 004530 012767 000000          INCTSA: PRINT   MES10          ;TEXT INSUFFICIENT CORE
1629 004536 016707 000000          TTYIN   INBUF, PROC          ;SYNC I OR E
1630 004542 011453 000000          CRLF
1631 004548 016767 001112 006236          MOV      INBUF, PROC          ;1 CHARACTER
1632 004554 011453 000000          PRINT   CRLF          ;SAVE IN TEMP STORE
1633 004560 016707 000000          IS:      PRINT, MES45          ;ASK FOR TEST
1634 004566 016707 000000          TTYIN   INBUF, ITEST          ;RECIEVE 1 CHAR.
1635 004572 016707 000000          MOV      8123, ITEST          ;SAVE TTY CHAR
1636 004578 016707 000000          CMP      8115, ITEST          ;IS IT AN S?
1637 004584 016707 000000          BNE      IS
1638 004590 016707 000000          BNE      IS
1639 004596 016707 000000          BNE      IS
1640 004602 016707 000000          BNE      IS
1641 004608 016707 000000          BNE      IS
1642 004614 016707 000000          BNE      IS
1643 004620 016707 000000          BNE      IS
1644 004626 016707 000000          BNE      IS
1645 004632 016707 000000          BNE      IS
1646 004638 016707 000000          BNE      IS
1647 004644 016707 000000          BNE      IS
1648 004650 016707 000000          BNE      IS
1649 004656 016707 000000          BNE      IS
1650 004662 016707 000000          BNE      IS
1651 004668 016707 000000          BNE      IS
1652 004674 016707 000000          BNE      IS
1653 004680 016707 000000          BNE      IS
1654 004686 016707 000000          BNE      IS
1655 004692 016707 000000          BNE      IS
1656 004698 016707 000000          BNE      IS
1657 004704 016707 000000          BNE      IS
1658 004710 016707 000000          BNE      IS
1659 004716 016707 000000          BNE      IS

```


E04

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 44
DZADHA.CMB

1716	005324	104000	012561		PRINT,MES46		:PRINT HEADER
1717	005330	005367	005356		DEC 50FLAG		:SET FLAG
1718	005334	016701	005550	15:	MOV OFFSET,R1		:POINT TO TABLE
1719	005340	104016			TSTTKS		
1720	005346	005711		25:	TST (R1)		:ANY INCS?
1721	005352	001005			BNE 45		
1722	005358	020167	005534	55:	CHP R1,MEMSIZ		:AT TOP OF TABLE YET?
1723	005364	001420			BEQ 35		:IF YES GET OUT
1724	005370	005721			TST (R1)+		:UPDATE POINTER
1725	005376	000771			BR 25		:KEEP CYCLEING
1726	005382	010167	005542	45:	MOV R1,HOLD		:LOAD VALUE FOR PRINTING
1727	005388	104010			PRTOCT		:PRINT THE LOCATION INCREMENTED
1728	005394	013026			HOLD		
1729	005400	012767	000003 000110		MOV #3,SPACE		:PRINT 3 SPACES
1730	005406	104007			SPACE		:PRINT THEM
1731	005412	104016			TSTTKS		:TEST TTY STATUS
1732	005418	011102			MOV (R1),R2		:GET DATA AND RESTORE R1
1733	005424	104006			BINDEC		:CONVERT DATA IN R2 AND PRINT
1734	005430	104000	011453		PRINT,CRLF		
1735	005436	000755			BR 55		:CYCLE TILL DONE
1736	005442	104000	011453	35:	PRINT,CRLF		:DOE THIS LOOP
1737	005448	104016			TSTTKS		:TEST TTY STATUS
1738	005454	000167	177544		JMP QUANT		:KEEP RUNNING
1739				:			
1740				:			
1741				:			

F04

 REF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 45
 02RDNA.CMB

```

1739 ;SUBROUTINE TO 'CLR' CORE BEFORE USING THE 'INC. MEM. MODE'
1740
1741 005326 005767 005452 CLRCOR: TST INCFLG ;MEMORY AVAILABLE?
1742 005328 001410 BEQ EXCORE ;IF NOT EXIT
1743 005330 016701 005450 CLRCR1: MOV OFFSET,R1 ;START CLEARING CORE AT OFFSET
1744 005332 005021 (R1)+ ;CLR CORE
1745 005334 020167 005440 CYP R1,MSIZ ;DONE
1746 005336 001374 BNE CLRCR1 ;BRANCH IF NOT
1747 005338 005077 005432 CLR ;CLR LAST LOCATION
1748 005340 000207 EXCORE: RTS PC ;EXIT
1749
1750 ;SUBROUTINE TO ISSUE N SPACES
1751 ;N IS ONE PLUS THE CORE COUNTED IN SPACEX
1752 ;SPACEX IS CLEARED WITHIN THE SUBROUTINE, SO THAT A CALL ON
1753 ;SPACE WITHOUT LOADING SPACEX ISSUES ONLY ONE SPACE
1754
1755 005356 105777 173624 XSPACE: TSTB JTPS ;WAIT FOR TTY READY
1756 005358 100375 L -4
1757 005360 012777 000240 173616 MOV J240,JTPB ;OUTPUT A SPACE
1758 005362 005367 000010 DEC SPACEX ;DECR INT COUNT
1759 005364 005367 000002 BGT XSPACE ;LOOP IF NOT DONE
1760 005366 005067 CLR SPACEX ;LINECOUNT TO 270
1761 005368 000002 RTI ;RETURN
1762 005370 000000 SPACEX: 0
1763
1764 ;KEYBOARD SERVICE ROUTINE
1765
1766 005410 104020 XTTYIN: SAVREG
1767 005412 012704 MOV @INBUF,R4 ;SETUP CHARACTER BUFFER
1768 005414 005067 CLR CHCNT ;CLEAR CHARACTER COUNTER
1769 005416 005067 CLR IF JF+2
1770 005418 105777 173550 INPUTA: TSTB JTKS ;CHARACTER READY?
1771 005420 100375 BPL JUTA ;NO, WAIT IT OUT
1772 005422 017701 173544 MOV JTKB,R1 ;SAVE CHARACTER
1773 005424 042701 000200 BIC @200,R1 ;STRIP PARITY BIT
1774 005426 120127 000060 CMPB R1,#0 ;IS IT A SPECIAL CHARACTER
1775 005428 103420 BMI SPCHR ;YES, TEST IT
1776 005430 123701 000137 CMPB @137,R1
1777 005432 011415 BMI SCHR
1778 005434 011424 INPUTB: MOV R1,(R4)+ ;SAVE CHARACTER
1779 005436 005332 INC CHCNT ;INCREMENT THE CHARACTER COUNT.
1780 005438 000007 005324 CMPB @7,CHCNT
1781 005440 011472 BMI SPCHR ;TYPE '?' IF TOO MANY CHAR.
1782 005442 101777 173504 OUTPUTA: TSTB JTPS ;ECHO CHARACTER
1783 005444 101375 BPL OUTPUTA
1784 005446 110177 173500 MOVB R1,JTPB
1785 005448 000746 BR INPUTA ;WAIT FOR NEXT CHARACTER

```


K04

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 50
DZADMA.CMB

;SUBROUTINE TO INPUT A 'GAIN FROM THE TELETYPE

```

XGAINA: PRINT
        MES18
        DECOCT
        MOV #INITAL,R1
        MOV #BCDTAB,R2
        JSR PC,XGAINB
        TST (R2)+
        TST (R2)
        BEQ EXGAIN
        MOV #INITL2,R1
        JSR PC,XGAINB
        EXGAIN: RTI
XGAINB: BIC #60000,(R1)
        CMP #1,(R2)
        BNE XGAIN2
        RTS PC
XGAIN2: CMP #2,(R2)
        BNE XGAIN4
        BIS #2,(R1)
        RTS PC
XGAIN4: CMP #4,(R2)
        BNE XGAIN3
        BIS #4,(R1)
        RTS PC
XGAINB: CMP #10,(R2)
        BNE NOGAIN
        BIS #8,(R1)
        RTS PC
NOGAIN: POP1SP
        BR XGAINA

```

;TEXT 'GAIN?'
;CONVERT TO OCTAL;SET UP BCDTAB+2
;WAS A SECOND GAIN ENTERED?
;NO EXIT
;SET UP SECOND GAIN;CLR GAIN BITS
;TEST FOR '1'
;IF NOT '1' TEST FOR '2'

;ILLEGAL ENTRY, TRY AGAIN

;RESET STACK
;ACCEPT NEW GAIN

```

;*****
;S  OUTLINE ENTERED ON AN ILLEGAL TRAP. THE ROUTINE REPORTS WHERE IT
;T  LD 'FROM' AND WHERE IT TRAP 'TO'.
;*****

```

```

ERTRAP: MOV (SP),TEMP1
        POP2SP
        MOV (SP),TEMP2
        PRINT
        MES40
        S 3
        PNT OCT
        TEMP1
        PRINT
        MES41
        S 3
        PNT OCT
        TEMP2

```

```

;SAVE LOCATION WHERE IT TRAPPED 'TO'
;SAVE WHERE IT TRAPPED FROM.
;TEXT 'ILLEGAL TRAP TO'
;TYPE 'PC' TRAPPED TO
;TEXT 'FROM'
;TYPE WHERE IT TRAPPED FROM

```

```

1962
1963
1964 006450 104000
1965 006452 011720
1966 006454 104001
1967 006456 012701 013034
1968 006458 012702 006064
1969 006460 004767 000020
1970 006472 007722
1971 006474 005712
1972 006476 001404
1973 006478 012701 013040
1974 006480 004757 000002
1975 006482 000002
1976
1977 006512 007711 060000
1978 006514 002712 000001
1979 006516 001001
1980 006518 000207
1981
1982 006536 022712 000002
1983 006538 001003
1984 006540 002711 020000
1985 006542 000007
1986
1987 006542 022712 000004
1988 006544 001003
1989 006546 002711 040000
1990 006548 000207
1991
1992 006556 022712 000010
1993 006558 001003
1994 006560 052711 060000
1995 006562 000207
1996
1997 006572 005726
1998 006574 000725
1999
2000
2001
2002
2003
2004
2005 006576 011667 004252
2006 006578 022626
2007 006580 011667 004246
2008 006582 104000
2009 006584 012451
2010 006586 162767 000004 004232
2011 006588 104010
2012 006590 013054
2013 006592 104000
2014 006594 012473
2015 006596 162767 000002 004216
2016 006598 104010
2017 006600 013056

```

2018	006644	000167	172716	JMP	MONITR	:RETURN TO MONITOR
2019				:*****		
2020				:SUBROUTINE TO SAVE 'R1-R5' ON STACK		
2021				:*****		
2022						
2023	006650	012667	004206	XSAVRG:	MOV	(SP)+,SAVEPC
2024	006651	012667	004204		MOV	(SP)+,SAVEPSW
2025	006660	012667	004202		MOV	(SP)+,SAVE2PC
2026	006664	012667	004200		MOV	(SP)+,SAVE2SW
2027	006670	010146			MOV	R1,-(SP)
2028	006672	010246			MOV	R2,-(SP)
2029	006674	010346			MOV	R3,-(SP)
2030	006676	010446			MOV	R4,-(SP)
2031	006700	010546			MOV	R5,-(SP)
2032	006702	016746	004162		MOV	SAVE2SW,-(SP)
2033	006706	016746	004154		MOV	SAVE2PC,-(SP)
2034	006712	016746	004146		MOV	SAVEPSW,-(SP)
2035	006716	016746	004140		MOV	SAVEPC,-(SP)
2036	006722	000002			RTI	
2037						
2038				:*****		
2039				:SUBROUTINE TO RESTORE 'R1-R5' FROM THE STACK		
2040				:*****		
2041						
2042	006724	012667	004132	XGETRG:	MOV	(SP)+,SAVEPC
2043	006730	012667	004130		MOV	(SP)+,SAVEPSW
2044	006734	012667	004128		MOV	(SP)+,SAVE2PC
2045	006740	012667	004124		MOV	(SP)+,SAVE2SW
2046	006744	012605			MOV	(SP)+,SAVE2PC
2047	006746	012604			MOV	(SP)+,SAVE2SW
2048	006750	012603			MOV	(SP)+,SAVE2PC
2049	006754	012602			MOV	(SP)+,SAVE2SW
2050	006758	016746	004106		MOV	SAVE2SW,-(SP)
2051	006762	016746	004100		MOV	SAVE2PC,-(SP)
2052	006766	016746	004072		MOV	SAVEPSW,-(SP)
2053	006772	016746	004064		MOV	SAVEPC,-(SP)
2054	006776	000002			RTI	

;COMMAND DECODER, ENTERED VIA '1P'
;ROUTINE ALLWS THE USER TO CHANGE A SINGLE PARAMETER (CHANNEL, GAIN
;COUNT SPREAD OR SYNC TYPE) WITHOUT CHANGING ALL PARAMETERS.

DCODER: PRINT
ASTRIC
TTYIN

MOV #INBUF, R1
CMP #103, (R1)
BEQ PARMC
CMP #107, (R1)
BEQ PARMG
CMP #123, (R1)
BEQ PARMS

PARMC: TST
IF #IF+2
BNE PARMCS

PARMG: GAININ

PARMS: TST
INBUF+2
BEQ
JMP
+6
DPVECTR

PARMCS: PRINT
MES16
DECOCT
MOV BCDTAB, KSTOR3
BR DCODER

;SUBROUTINE TO TYPE OUT '5' AVERAGES FOR THE GAIN TEST HISTOGRAM.

XPRTAB: MOV ICOUNT, TEMP3
XPTA1: PRT OCT
AVGTAB: GP5OX1
ADD #2, AVGTAB
MOV #2, SPACEX
SPACE
DEC TEMP3
BNE XPTA1
RTI

;TYPE '*' TO INDICATE READY
;WAIT FOR INPUT
;SET UP TO TEST CHAR.
;WAS 'C' TYPED?
;BRANCH IF YES AND DECODE 'C' OR 'CS'
;WAS 'G' TYPED?
;BRANCH IF YES AND DECODE 'GAIN'
;WAS 'S' TYPED
;BRANCH IF YES AND DECODE 'SYNC' OR 'ST'
;ILLEGAL CALL
;TYPE '?'
;RESTART
;TEST FOR 2 CHARACTER INPUT
;BRANCH IF YES AND DECODE COUNT SPREAD
;OTHERWISE REQUEST CH.
;WAIT NEXT INSTRUCTION
;REQUEST GAIN
;WAIT NEXT INSTRUCTION
;TEST FOR 2 CHARACTER INPUT
;BRANCH IF NO
;OTHERWISE ASSUME 'ST' & EXIT

;TEXT 'SYNC?'
;WAIT INPUT
;SAVE INPUT

;TEXT 'COUNT SPREAD'
;CONVERT TO OCTAL
;SAVE IT

;PRINT OCTAL VALUE OF GAIN AVERAGE

;UPDATE GAIN TABLE

;TYPE '2' SPACES

;IF NOT DONE, PRINT NEXT AVG.

2056
2057
2058
2059
2060 007000 104000
2061 007002 011455
2062 007004 104011
2063 007006 012701 005670
2064 007012 022711 000103
2065 007016 001411
2066 007020 002711 000107
2067 007024 001413
2068 007028 002711 000123
2069 007032 104000
2070 007036 011453
2071 007040 000757
2072 007042 005767 176624
2073 007046 001020
2074 007050 104017
2075 007052 000752
2076 007054 104003
2077 007056 000750
2078 007060 005767 176606
2079 007064 001402
2080 007066 000177 003710
2081 007072 104000
2082 007074 012441
2083 007076 104011
2084 007100 016767 176564 003710
2085 007106 000734
2086 007110 104000
2087 007112 011701
2088 007114 104001
2089 007116 016767 176742 003724
2090 007124 000725
2091
2092
2093
2094
2095 007126 016767 003672 003724
2096 007134 104010
2097 007136 013216
2098 007140 062767 000002 177770
2099 007146 012767 000002 176232
2100 007154 104007
2101 007156 005367 003676
2102 007162 001364
2103 007164 000002

2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150

007166	011646	
007170	162716	000002
007174	017616	000000
007200	005716	
007202	001001	
007204	000000	
007206	000016	
007210	042716	177001
007214	062716	007226
007220	017616	000000
007224	000136	
007226	007306	
007230	005710	
007232	010246	
007234	000450	
007236	007604	
007240	007726	
007242	007404	
007244	005356	
007246	010576	
007250	005410	
007252	003766	
007254	003770	
007256	007126	
007260	007272	
007262	010702	
007264	006336	
007266	000650	
007270	006724	
007272	104000	
007274	012430	
007276	005367	003520
007302	001373	
007304	000002	

; EMT DISPATCH SERVICE ROUTINE
; ARGUMENT OF EMT IS EXTRACTED AND USED AS OFFSET TO OBTAIN POINTER
; TO THE SELECTED SUBROUTINE.

EMTSRV:	MOV	(SP), -(SP)	; GET PC FOR TO RETURN
	SUB	#2, (SP)	; PC OF EMT
	MOV	2(SP), (SP)	; GET EMT
	TST	(SP)	; IS EMT VALID?
	BNE	EMTOK	
	HALT		; INVALID EMT
EMTOK:	ASL	(SP)	; MULTIPLY EMT ARG BY '2'
	BIC	#177001, (SP)	; CLEAR UNWANTED BITS
	ADD	#EMTTAB, (SP)	; POINTER TO SUBROUTINE ADDRESS
	MOV	2(SP), (SP)	; SUBROUTINE ADDRESS
	JMP	2(SP)+	; GO TO SUBROUTINE

; EMT DISPATCH TABLE

EMTTAB:	TYPNES	; MESSAGE PRINT ROUTINE
	BCDPRN	; DECIMAL TO BINARY CONVERSION ROUTINE
	XRC EM	; SUBROUTINE TO READ & CATEGORIZE INC. MEM. VALUES
	XGAINA	; REQUEST A 'GAIN' FROM THE TTY.
	CMPT	; SUBROUTINE TO COMPUTE THE AVG
	CATORZ	; SUBROUTINE TO COMPUTE 'COUNT SPREAD'
	DECPRT	; SUBROUTINE TO CONVERT OCT TO DEC + PRINT
	XSPACE	; SUBROUTINE TO TYPE SPACES
	OCTPRT	; OCTAL PRINT ROUTINE
	XTTYIN	; TELEPRINTER SERVICE ROUTINE
	XWATGN	; GAIN TEST CONVERSION ROUTINE
	XWATGN+2	; GAIN TEST CONVERSION ROUTINE
	XPRTAV	; SUBROUTINE TO PRINT OUT THE GAIN AVERAGES
	DASH6	; SUBROUTINE TO TYPE OUT '6' DASHES
	TKSFLG	; SUBROUTINE TO TEST FOR KEYBOARD FLAG
	XCHAIN	; SUBROUTINE TO DECODE A CHANNEL FROM TTY
	XSAVRG	; SUBROUTINE TO SAVE REG'S ON THE STACK
	XGETRG	; SUBROUTINE TO GET REG'S FROM THE STACK

; SUBROUTINE TO TYPE OUT 'N' SETS OF SIX DASHES

DASH6:	PRINT	
	MES38	
	DEC	COUNT
	BNE	DASH6
	RTI	

2207 007564 000000
2208 007566 000000
2209 007570 000240
2210 007572 007574
2211 007574 001750
2212 007576 000144
2213 007600 000012
2214 007602 000001

DIGIT: 0
DIGCNT: 0
ZERO: 240
DECPNT: +2
1000.
100.
10.
1.

; COMPUTE THE RESULTS OF '1024' CONVERSIONS AS HIGH, LOW AND AVERAGE

2215
2216
2217
2218 007604 012701 001777
2219 007610 005000
2220 007612 012704 013434
2221 007616 012403
2222 007620 010367 003252
2223 007624 010367 003250
2224 007630 066703 003160
2225 007634 012402
2226 007636 020267 003234
2227 007642 003402
2228 007644 010267 003226
2229 007650 020267 003224
2230 007654 003002
2231 007656 010267 003216
2232 007662 066702 003126
2233 007666 012203
2234 007670 012203
2235 007672 012301
2236 007674 001357
2237 007676 012701 000012
2238 007702 006200
2239 007704 005003
2240 007706 005301
2241 007710 001374
2242 007712 005503
2243 007714 166703 003074
2244 007720 010367 003170
2245 007724 000002

CMPT: MOV #1777, R1 ; SET UP TO COMPARE '1023' NUMBERS
CLR R0 ; CLR HI OF ER DIVIDEND
MOV #ADBUFF, R4 ; SET UP DATA BUFFER ADDRESS
MOV (R4)+, R3 ; STORE 1ST VALUE AS AVERAGE
MOV R3, HIGH ; HIGH
MOV R3, LOW ; & LOW
ADD #ADSIZE, R3 ; ADD OFFSET TO AVERAGE
GETDAT: MOV (R4)+, R2
CMP R2, HIGH ; IS NEW NO. GREATER THAN OLD NO.
BLE TSLO ; BRANCH IF NOT GREATER
MOV R2, HIGH ; OTHERWISE SAVE AS NEW HIGH
TSLO: CMP R2, LOW
BGT TAGA
MOV R2, LOW ; OTHERWISE SAVE AS NEW LOW
TAGA: ADD #ADSIZE, R2 ; ADD OFFSET TO MAKE ALL NO. POS.
ADD R2, R3 ; ADD LOW ORDER
ADD R0, R3 ; ADD CARRY TO HI ORDER
DEC R1
BNE GETDAT ; 1024 ADDITIONS?
MOV #12, R1 ; YES, DIVIDE/1024
AVGDAT: ASR R0 ; SHIFT CARRY BIT INTO LO ORDER
ROR R3
DEC R1
BNE AVGDAT
ADC R3 ; DONE?
SUB #ADSIZE, R3 ; YES, ADD REMAINDER TO LO ORDER
MOV R3, AVRAVE ; SUBTRACT OFFSET TO OBTAIN REAL AVERAGE
RTI ; SAVE AS AVERAGE

;SUBROUTINE TO CALCULATE THE PLUS & MINUS 5 COUNT LIMITS FROM AN AVERAGE

```

CATORZ: MOV #5, R1
        MOV AVERAGE, R2
        ADD AOSIZE, R2
        MOV AVERAGE, R3
        INC R2
        MOV R2, (R3)
        SUB AOSIZE, (R3)+
        DEC R1
        BNE FILE1
;SET UP TABLE OF AVG. -1 TO -5
        MOV #5, R1
        MOV AVERAGE, R2
        ADD AOSIZE, R2
        MOV AVERAGE, R3
        DEC R2
        MOV R2, -(R3)
        SUB AOSIZE, (R3)
        DEC R1
        BNE FILE2
;MOV AVG. TO WORK AREA.
;SET UP DISTRIBUTION TABLE NEG.
A=1-1
SAVE 'A-1'
RESTORE ORIGINAL NO. -1
SAVED '5' COUNTS?
BRANCH IF NO

```

;CATEGORIZE THE COUNT SPREAD AS '+6 & -6' COUNTS FROM THE AVERAGE

```

CATR1: MOV BORLOW, R3
        CLR (R3)+
        CMP BORHIGH+2, R3
        BNE CATR1
        MOV #2001, R3
        MOV #0, R4
        DEC R3
        CATRS
        (R0)+, R4
        CMP #5, R4
        BNE CATR1
        CMP R4, AVERMS
        BNE CATR1
        CLR R1
        MOV AVERAGE, R2
        CMP (R2)+, R4
        BEQ CATR4
        INC R1
        CMP #13, R1
        BNE CATR3
        HALT
CATR4: RSL R1
        INC MINUSS(R1)
        BR CATR2
OVRHI: INC ORHIGH
        BR CATR2
OVRLO: INC ORLOW
        BR CATR2

```

;FATAL ERROR MR. BOUSE!
;MULTIPLY 'OFFSET' X2

```

007726 012701 000005
007727 016702 003156
007728 066702 003052
007729 012703 013116
007730 005202
007731 010213
007732 166723 003036
007733 005301
007734 001372
007735 012701 000005
007736 016702 003122
007737 066702 003016
007738 012703 013114
010002 005302
010004 010243
010006 166713 003002
010012 005301
010014 001372
010016 012703 013130
010022 005023
010024 022703 013162
010030 001374
010032 012703 002001
010036 012700 013434
010042 005303
010044 001433
010046 012004
010050 026704 003052
010054 100421
010056 020467 003020
010062 100421
010064 001701
010066 012702 013102
010072 022204
010074 001405
010076 005201
010100 022701 000013
010104 001372
010106 000000
010110 001701
010112 001751 013132
010116 000751
010120 001757 003034
010124 000746
010126 005267 002776
010132 000743

```

;ADD THE COUNTS AND SAVE TOTAL IN SPREADS OF '1-4'

```

CATRS:  MOV    AVG CNT,XSPRD1
        ADD     PLUS1,XSPRD1
        ADD     MINUS1,XSPRD1      ;=TO NO. COUNTS AT SPREAD OF '1'
        MOV     XSPRD1,XSPRD2
        ADD     PLUS2,XSPRD2
        ADD     MINUS2,XSPRD2      ;=TO NO. COUNTS AT SPREAD OF '2'
        MOV     XSPRD2,XSPRD3
        ADD     PLUS3,XSPRD3
        ADD     MINUS3,XSPRD3      ;=TO NO. COUNTS AT SPREAD OF '3'
        MOV     XSPRD3,XSPRD4
        ADD     PLUS4,XSPRD4
        ADD     MINUS4,XSPRD4      ;=TO NO. COUNTS AT SPREAD OF '4'
        RTI      ;EXIT

```

;SUBROUTINE TO COMPUTE THE HIGH, AVERAGE AND LOW VALUES THAT WERE STORED
;IN MEMORY VIA THE INCREMENT MEMORY MODE

```

XRDMEN:  MOV     OFFSET,R0      ;START AT 4K OR 8K FOR 13 BITS++++++
        CLR      R1
        CLR      R2
        CLR      R3
        CLR      HIGH
        CLR      LOW
        CLR      SOFLAG
        MOV     (R0),R4 ;MOVE THE COUNTS TO TEMP (R4)
        DEC     R4
        BMI     NXADDR
        MOV     R1,HIGH
        ADD     R1,R3
        ADC     R2
        TST     SOFLAG
        BNE     RDMEN2
        MOV     R1,LOW
        DEC     SOFLAG
        MOV     RDMEN2
        ;INITIAL HEATER INDICATOR
        ;CHECK IF TEMP
        ;CHECK IF EMPTY
        ;SAVE AS HIGH
        ;ADD VALUES WEIGHT TO LOW ORDER
        ;ADD CARRY TO HIGH ORDER
        ;CHECK IF LOW VALUE HAS BEEN SAVED
        ;OTHERWISE LOOP AGAIN
        ;SAVE 1ST ENTRY AS LOW VALUE
        ;SET FLAG TO INDICATE THAT LOW VALUE IS FILLED

NXADDR:  INC     R1
        CLP     R0,MENSIZ
        BEQ     IS
        TST     (R0)+
        BNE     RDMEN1
        MOV     R12,R4
        MOV     R2
        MOV     R3
        DEC     R4
        BNE     RDMEN
        MOV     R3,AVRAGE
        TST     RDSIGN
        BEQ     IS
        MOV     RDSIZE,R4
        CLC
        RSL     R4
        ;INCREMENT VALUE
        ;DONE?
        ;UPDATE POINTER
        ;SET UP TP DIVIDE BY 1024
        ;SHIFT CARRY INTO LOW ORDER
        ;CHECK IF NO.
        ;YIELD AND 7 POINTER TO LOW ORDER
        ;SHIFT CARRY TO TOTAL VALUE
        ;IF BI POLAR SHIFT FLAG, INVERTING THE SIGN
        ;BITS TO COMPUTE FOR LINEAR MEMORY
        ;ADJUST TO DETERMINE IF SIGN IS SET
        ;1 LEFT FOR BEGINING FF SIGN BIT

```


H05

MESSAGES

TITLE: .BYTE
.ASCII 'XADF11 PART II, ANALOG DIAGNOSTIC TEST, 23/8/75'

.ASCII '(MAINDEC-11-DZADH-A)@'

MES2: .ASCII 'XA/D LENGTH? @'

MES3: .ASCII 'X"GAIN ACCURACY TEST". SUPPLY THE FOLLOWING VOLTAGES X'

.ASCII " TO SELECTED CH., TYPE 'CR' TO START TEST.@"

MES4: .ASCII "XTYPE LETTER ' ' TO RUN DESIRED TEST:X"

.ASCII "'C'ALIBRATION, 'R'EPEATIBILITY, 'G'AIN, R'E'COVERY,"

.ASCII "'I'NCREMENT MEMORYX@"

011430	046440	046505	051117	
011436	022531	100		
011441	136	022503	100	CNTRLC: .ASCII 'ICX2'
011445	136	040101		CNTRLA: .ASCII 'IA2'
011450	050136	100		CNTRLP: .ASCII 'IP2'
011453	045	100		CRLF: .ASCII 'X2'
011455	045	040052		ASTRIC: .ASCII 'X22'
011460	027045	100		DOT: .ASCII 'X.2'
011463	077	040040		QMARK: .ASCII '? 2'
011466	047111	027103	046440	MES6: .ASCII 'INC. MEM. (Y OR N)? 2'
011474	047111	027103	046440	
011502	047440	020122	046450	
011510	020077	100	046516	
011513	045	041442	046101	MES7: .ASCII 'X"CALIBRATION TEST"X2'
011520	047111	047111	046524	
011526	047111	047111	051505	
011534	047111	047111		
011540	047111	047111	047503	MES8: .ASCII 'X"RECOVERY TEST"2'
011546	047111	047111	052040	
011551	051505	021124	100	
011561	045	044103	020056	MES9: .ASCII 'XCH. 2'
011567	100			
011567	045	042447	054047	MES10: .ASCII 'X'E'XT. OR 'I'NT. SYNC? 2'
011574	047111	047440	020122	
011580	047440	047047	027124	
011586	051440	047131	037503	
011592	040040			
011620	051105	047522	020122	MES12: .ASCII 'ERROR BIT SET!X2'
011626	046505	020124	042523	
011634	020624	040045		
011640	021045	042522	042520	MES13: .ASCII 'X"REPEATIBILITY TEST"2'
011646	052101	041111	046111	
011654	052111	020131	042524	
011662	052123	040042		
011666	022445	044103	024056	MES14: .ASCII 'XXCH.(S)? 2'
011674	024523	020077	100	
011701	103	052517	052116	MES16: .ASCII 'COUNT SPREAD' 2'
011706	051440	051120	040506	

J05

FORM 11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 62
 L.C.

011714	037504	040040		
011720	040507	047111	051450	MES18: .ASCII 'GAIN(S)? 2'
011726	037451	040040		
011732	020045	041440	027110	MES19: .ASCII '% CH. LO AV HI2'
011740	020040	045040	020117	
011746	020040	020040	053101	
011754	020040	020040	044040	
011762	040111			
011764	020045	020040	047514	MES20: .ASCII '% LO -5 -4 -3 -2 -1 AV'
011772	020040	026440	020065	
011780	020040	032053	020040	
011786	020040	020040	020040	
011794	031053	020040	026440	
011802	020061	020040	053101	
011810	020040	020061	020040	.ASCII' +1 +2 +3 +4 +5 HI2'
011818	020040	020040	026440	
011826	031053	020040	020040	
011834	020063	020040	032053	
011842	020040	025440	020065	
011850	020040	044510	100	
012045	045	047520	042527	MES21: .ASCII 'XPOWER FAILURE 2'
012053	020122	040506	046111	
012100	051125	020106	100	
012105	045	041445	027110	MES22: .ASCII 'XXCH.? 2'
012112	020077	100		
012115	053	027065	030060	MES23: .ASCII '+5.00V2'
012122	040126			
012124	053523	052111	044103	MES24: .ASCII 'SWITCH NEG.!2'
012132	047040	043505	020456	
012140	100			
012141	053	027062	030065	MES25: .ASCII '+2.50V2'
012146	040126			
012150	030453	031056	053065	MES26: .ASCII '+1.25V2'
012156	100			
012157	053	027060	031066	MES27: .ASCII '+0.625V2'
012164	053625	100		
012167	053	027060	030463	MES28: .ASCII '+0.3125V2'
012174	032462	040126		
012200	030753	030456	033065	MES29: .ASCII '+0.1563V2'
012206	053633	100		
012211	053	027060	033460	MES30: .ASCII '+0.0781V2'
012216	030470	040126		
012222	030053	030056	034463	MES31: .ASCII '+0.0390V2'

K05

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 63
DZADHA.CMB

2609	012230	053060	100		
2610					
2611	012233	045	040507	047111	MES32: .ASCII '%GAIN 5.0000 2.5000 1.2500 0.6250 '
2612	012240	020040	027065	030060	
2613	012246	030060	020040	027062	
2614	012254	030065	030060	020040	
2615	012262	027061	032462	030060	
2616	012270	020040	027060	031066	
2617	012276	030065	020040		
2618	012302	027060	030463	032462	.ASCII '0.3125 0.1563 0.0781 0.3906'
2619	012310	020040	027060	032461	
2620	012316	031466	020040	027060	
2621	012324	033460	030470	020040	
2622	012332	027060	034463	040060	
2623					
2624	012340	043445	044501	020116	MES33: .ASCII '%GAIN VOLTAGE AVERAGE%'
2625	012346	053040	046117	040524	
2626	012354	042507	040440	042526	
2627	012362	040522	042507	040045	
2628					
2629	012370	030445	020040	020040	MES34: .ASCII '%1 0'
2630	012376	040040			
2631					
2632	012400	031045	020040	020040	MES35: .ASCII '%2 0'
2633	012406	040040			
2634					
2635	012410	032045	020040	020040	MES36: .ASCII '%4 0'
2636	012416	040040			
2637					
2638	012420	034045	020040	020040	MES37: .ASCII '%8 0'
2639	012426	040040			
2640					
2641	012430	026455	026455	026455	MES38: .ASCII '----- 0'
2642	012436	020040	100		
2643	012441	045	054523	041516	MES39: .ASCII '%SYNC? 0'
2644	012446	020077	100		
2645	012451	045	046111	042514	MES40: .ASCII '%ILLEGAL TRAP TO 0;
2646	012456	045	041114	041124	
2647	012461	050101	050040	020117	
2648	012466	100			
2649	012471	045	051106	046517	MES41: .ASCII '% FROM 0;
2650	012476	040040			
2651	012481	045	051516	043125	MES42: .ASCII '%INSUFFICIENT MEMORY0'
2652	012486	045	047105	047515	
2653	012491	045	047503	042522	MES43: .ASCII '%CORE MEMORY NOT INCREMENTED=0'
2654	012496	046440	047516	041117	
2655	012501	047111	047105	041124	
2656	012506	047105	047105	041124	
2657					
2658	012511	047111	051103		MES44: .ASCII '%INCREMENT MEMORY TEST0'
2659	012516	047105	051124		
2660	012521	047515	051522		
2661	012526	051505	040124		

L05

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 64
DZADHA.CMB

2665	012614	047503	053116	051105	MES45: .ASCII /CONVERSIONS 'S'INGLE OR 'M'ULTIPLE 2/
2666	012622	044524	047117	020123	
2667	012630	051447	044447	043516	
2668	012636	042514	047440	020122	
2669	012644	046447	052447	052114	
2670	012652	050111	042514	020040	
2671	012660	100			
2672	012661	045	047503	042522	MES46: .ASCII '%CORE LOC.INCREMENTED,# OF CONVERSIONS IN LOCATION%%2'
2673	012666	046040	041517	044456	
2674	012674	041516	042522	042515	
2675	012702	052116	042105	021454	
2676	012710	047440	020106	047503	
2677	012716	053116	051106	044524	
2678	012724	047117	020123	047111	
2679	012732	046040	041517	052101	
2680	012740	047511	022516	040045	
2681	012746	046445	046505	051117	MES47: .ASCII '%MEMORY OFFSET = 2'
2682	012754	020131	043117	051506	
2683	012762	052105	036440	040040	
2684					
2685					

2742	013140	000000
2743	013142	000000
2744	013144	000000
2745	013146	000000
2746	013150	000000
2747	013152	000000
2748	013154	000000
2749	013156	000000
2750	013160	000000
2751	013162	000000
2752	013164	000000
2753	013166	000000
2754	013170	000000
2755	013172	000000
2756	013174	000000
2757	013176	000000
2758	013200	000000
2759	013202	000000
2760	013204	000000
2761	013206	000000
2762	013210	000000
2763	013212	000000
2764	013214	000000
2765	013216	000000
2766	013220	000000
2767	013222	000000
2768	013224	000000
2769	013226	000000
2770	013230	000000
2771	013232	000000
2772	013234	000000
2773	013236	000000
2774	013240	000000
2775	013242	000000
2776	013244	000000
2777	013246	000000
2778	013250	000000
2779	013252	000000
2780	013254	000000
2781	013256	000000
2782	013260	000000
2783	013262	000000
2784	013264	000000
2785	013266	000000
2786	013270	000000
2787	013272	000000
2788	013274	000000
2789	013276	000000
2790	013300	000000
2791	013302	000000
2792	013304	000000
2793	013306	000000
2794	013310	000000
2795	013312	000000
2796	013314	000000
2797	013316	000000

MINUS2:	0
MINUS1:	0
AVGCNT:	0
PLUS1:	0
PLUS2:	0</

2798	013320	000000	GM15X4: 0
2799	013322	000000	GM07X4: 0
2800	013324	000000	GM62X8: 0
2801	013326	000000	GM31X8: 0
2802	013330	000000	GM15X8: 0
2803	013332	000000	GM07X8: 0
2804	013334	000000	GM03X8: 0
2805			; HERE STARTS A '30' WORD STATUS WORD BUFFER
2806	013336	000000	RANBUF: 0
2807		013434	. = . +60.
2808			
2809			; HERE STARTS THE '512' WORD A/D DATA BUFFER.
2810			ADBUFF: 0
2811	013434	000000	
2812			
2813		001242	.END INIT

C06

 ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 69
 DZADHA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

RODR	001234	871#	1071#	17#	1701#	1895#												
ROBLFF	013434	1023	1031	101	1897	1891	2220	2275	2811#									
ROCHVT	006074	1020	1037	1510	1593	1659	1703	1884#										
ROCR	001216	864#	1897#	1898#														
ROCSR	001220	865#	1646#	1691#	1884#	1905	2430#	2437#										
RODBR	001228	868#																
ROINT	001236	872#	2427#	2439#														
ROVL	001240	873#	2429#	2438	2439#													
ROSIGN	012770	883#	913	1181	1207	1243	1291	1345	1393	1429	1453	1489	1818#	2349				
		2690#																
ROSIZE	013014	932#	935	2224	2232	2243	2250	2254	2260	2264	2351	2700#						
ROSMR	001229	866#	1660	1665	1672	1675	1886#											
ROUCR	001234	867#	1018#	1085#	1509#	1592#	1658#	1702#	1888									
ROUAB	001230	869#	1687#															
ROUAB	001230	870#	1679#	1889#	1890#	1891#												
ROSTRIC	011455	2061	2508#															
ROVECTR	013000	958#	990#	1055#	1164#	1570#	1622#	1797</										

D06

 R0F11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 70
 DZADHA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

CHKCNT	002536	1102	1108#																	
CHKCNT	01373	1771#	1783#	1783	1810	1812#	1843	1845#	1864	2702#										
CLRCOR	01373	1084	1647	1690	1744#															
CLRCOR	01373	1747#	1749																	
CLRCOR	01373	954	1904	2436#																
CLRCOR	01373	2128	2218#																	
CLRCOR	01373	840#	1072	1511																
CLRCOR	01373	1794																		
CLRCOR	01373	976																		
CLRCOR	01373	1802	2304#																	
CLRCOR	01373	1893	1903#																	
CLRCOR	01373	1906	1909#																	
CLRCOR	01373	914	916#																	
CLRCOR	01373	1476#	1478#	1486#	1581#	2147#	2703#													
CLRCOR	01373	1003	1003	1111	1117	1137	1496	1525	1635											

H06

 ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 74
 DZADHA.CMS CROSS REFERENCE TABLE -- USER SYMBOLS

POS125	013176	1229	1277	1337	1391	2757#													
POS250	013174	1199	1235	1283	1343	2756#													
POS312	013202	1325	1379	1421	1451	2753#													
POS330	013172	901	1179	1275	1241	1753	2755#												
POS335	013100	1271	1331	1373	1477	2753#													
PRINT =	104000	836#	871	874	873	940	943	955	960	978	992	994	997	1002					
		1028	1066	1064	1074	1110	1114	1116	1131	1136	1165	1175	1186	1195					
		1211	1225	1247	1267	1295	1321	1349	1375	1397	1417	1433	1447	1457					
		1468	1471	1474	1479	1484	1495	1524	1532	1536	1541	1546	1549	1571					
		1596	1625	1628	1630	1634	1676	1643	1663	1682	1713	1716	1734	1736					
		1793	1791	1824	1828	1872	1907	1937	1943	1964	2008	2013	2060	2070					
		2753	2737	2145															
PROC	013016	1073#	1015	1633#	1651	1695	1920#	1926	2005#	2701#									
PRTAOR	004256	1504#	1514	1516	1518	1552#													
PRTAVG#	104014	848#	1024	1473	1														

		1805	1808	1813#	1816	1820	1822	1841#	1854	1855#	1856#	1857#	1858#	1859#
		1860#	1863	1914	1932#	1967#	1973#	1977#	1984#	1989#	1994#	2027	2050#	2063#
		2064	2066	2068	2218#	2235#	2237#	2240#	2248#	2255#	2258#	2265#	2283#	2297#
		2288	2291#	2292#	2319#	2328	2329	2333	2337#	2374#	2377#	2380#	2383	2388
R2	=X000002	2390#	2399#	2413#										
		537#	873#	893#	902	907	931#	932	1100#	1101	1118#	1139#	1583#	1585#
		1587#	159#	1598#	1732#	1842#	1854#	1858	1859	1915	1931#	1968#	1970	1971
		1978	1982	1987	1992	2008	2049#	2157#	2161	2165	2167	2173	2184#	2186#
		2225#	2226	2228	2229	2231	2232#	2233	2249#	2250#	2253#	2253	2259#	2260#
R3	=X000003	2262#	2263	2284#	2285	2320#	2330#	2343#	2405#	2408#	2409#	2411		
		538#	891#	896#	902#	903	904#	907#	908#	909	910#	1839#	1863#	1916
		1930#	2029	2048#	2221#	2222	2223	2224#	2233#	2239#	2242#	2243#	2244	2251#
		2253#	2254#	2261#	2263#	2264#	2270#	2271#	2272	2274#	2275#	2321#	2329#	2344#
		2347#	2348	2371#	2372#									

K06

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 77
DZADHA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

. = 013436

562#	563	565	567	569	571	573	575	577	579	581	583	585
587	589	591	593	595	597	599	601	603	605	607	609	611
613	615	617	619	621	623	625	627	629	631	633	635	637
639	641	643	645	647	649	651	653	655	657	659	661	663
665	667	669	671	673	675	677	679	681	683	685	687	689
691	693	695	697	699	701	703	705	707	709	711	713	715
717	719	721	723	725	727	729	731	733	735	737	739	741
743	745	747	749	751	753	755	757	759	761	763	765	767
769	771	773	775	777	779	781	783	785	787	789	791	793
795	797	799	801	803	805	807	809	811	813	815	817	819
821	823	825	827	829	831	833	835	837	839	841	843	845
847	849	851	853	855	857	859	861	863	865	867	869	871
873	875	877	879	881	883	885	887	889	891	893	895	897
899	901	903	905	907	909	911	913	915	917	919	921	923
925	927	929	931	933	935	937	939	941	943	945	947	949
951	953	955	957	959	961	963	965	967	969	971	973	975
977	979	981	983	985	987	989	991	993	995	997	999	1001
1003	1005	1007	1009	1011	1013	1015	1017					

MO6

ADF11A PART II, ANALOG DIAGNOSTIC TEST MACY11 27(732) 26-OCT-76 16:42 PAGE 80
DZADHA.CMB CROSS REFERENCE TABLE -- MACRO NAMES

.SSUPR	18
.STRAP	18
.STYPB	18
.STYPD	18
.STYPE	18
.STYPO	18
.S4OCA	18
.1170	18

MOV8	1083	1085	1098	1099	1100	1118	1138	1139	1164	1168	1466	1467	1470	1476	1481
NEG	1486	1503	1504	1506	1509	1512	1570	1573	1576	1581	1582	1583	1584	1587	1588
RESET	1598	1598	1601	1618	1619	1621	1622	1633	1638	1645	1646	1648	1654	1656	1657
FOR	1658	1660	1665	1672	1675	1681	1691	1692	1698	1700	1701	1702	1718	1726	1729
FOR	1733	1737	1760	1770	1775	1781	1813	1818	1838	1839	1853	1854	1863	1885	1886
RTI	1887	1888	1897	1898	1913	1914	1915	1916	1917	1918	1919	1920	1921	1925	1926
RTS	1932	1933	1937	1938	1931	1932	1933	1950	1952	1967	1968	1973	2005	2007	2023
SUB	2023	2023	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2042	2043	2044
SW8	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2063	2085	2090	2095	2099
TST	2110	2112	2119	2157	2169	2172	2179	2180	2181	2182	2199	2201	2205	2218	2220
TSTB	2223	2223	2223	2225	2228	2231	2237	2244	2248	2249	2251	2253	2258	2259	2261
WAIT	2270	2270	2274	2275	2278	2284	2301	2304							

10			...B1	2160	007324	100375	...B5
34			...C1	2216			...C5
			...D1	2255	007756	005301	...D5
91			...E1	2307	010200	016767	...E5
			...F1	2363	010446	040467	...F5
151			...G1	2419			...G5
			...H1	2450	011052	027463	...H5
211			...I1	2506	011453	045	...I5
			...J1	2562	011762	040111	...J5
271			...K1	2618	012302	027060	...K5
			...L1	2674	012674	041516	...L5
329			...M1	2695	013002	001722	...M5
			...N1	2751	013162	000000	...N5
389			...B2	2807	013336	000000	...B6
440			...C2				...C6
459			...D2	CNTRLP	011450		...D6
482			...E2	GM31X1	013276		...E6
538		000003	...F2	INITA	001476		...F6
569	000014	000016	...G2	MES40	012451		...G6
625	000174	000176	...H2				...H6
681	000354	000356	...I2				...I6
737	000534	000536	...J2	SW15	= 100000		...J6
793	000714	000716	...K2				...K6
843		104007	...L2	REPORT	1#	CROSS RE	...L6
883	001260	005067	...M2				...M6
937	001540	006301	...N2		1847	1957	...N6
990	001730	012767	...B3		2110	2112	...B7
			...C3	**END**	USER	DAVIES, TOM	...C7
1049			...D3				