```
.REM _
```

IDENTIFICATION

PRODUCT CODE: AC-F001B-MC

PRODUCT NAME: CXKWFB0 GROSS TMNG MOD

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT 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 MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITALS COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

COPYRIGHT (C) 1976,1978 DIGITAL EQUIPMENT CORPORATION

1.0 ABSTRACT

"KWF" IS A GROSS TIMING ANALYSIS MODULE THAT USES THE KW11-L LINE CLOCK OPTION TO PERFORM AN OVERALL TIMING ANALYSIS ON THE 2040 PDP11 CONSOLE PROCESSOR SYSTEM OR A DN2X SECONDARY F.E. IT WILL BE CONFIGURED IN THE EXERCISER TO BE THE FIRST MODULE INITIALIZED BY THE MONITOR AND SINCE IT IS A "NBKMOD" IT WILL RUN ONLY ONE PASS AND NEVER RUN AGAIN UNTIL THE EXERCISER IS RESTARTED. ITS PURPOSE IS TO ESTABLISH CONFIDENCE IN THE HARDWARE AFFECTING OVERALL SYSTEM TIMING. IF THE TIMING IS FOUND TO BE WITHIN A PRE-ESTABLISHED, EMPIRICALLY DETERMINED LIMIT, THE MODULE WILL ENABLE USE OF THE SYSTEM TIMER MODULE. IF NOT IT WILL REPORT THE ERROR AND DISABLE USE OF THE SYSTEM TIMER MODULE.

2.0 REQUIREMENTS

HARDWARE: ANY PDP11/40 OR 11/34A SYSTEM WITH CORE MEMORY AND A KW11-L LINE CLOCK.

STORAGE:: KWF REQUIRES:
1. DECIMAL WORDS: 355
2. OCTAL WORDS: 0543
3. OCTAL BYTES: 1306

3.0 PASS DEFINITION

ONE PASS OF KWF CONSISTS OF A SINGLE PASS THROUGH THE MODULE CODE TO PERFORM THE REQUIRED DIAGNOSIS AND ANALYSIS.

4.0 EXECUTION TIME

ONE PASS OF KWF RUNNING ON AN 11/40 CPU TAKES APPROXIMATELY ONE MINUTE.

NOTE: THE TIMING INFORMATION IN THE "ENDPAS" MESSAGE FOR THE "KWF" MODULE IS MEANINGLESS SINCE THE TIMER MODULE DOES NOT GET STARTED UNTIL "KWF" REPORTS ITS END OF PASS.

5.0 CONFIGURATION PARAMETERS

DEFAULT PARAMETERS:

DVA: 177546 VCT: 100 BR1: 6 BR2: 0 DVC: 1 SR1: 000037

REQUIRED PARAMETERS:

FOR 50 CYCLE SYSTEMS THE TIMING PARAMETER IN SR1 SHOULD BE SET TO 000030. (APPROX)

FOR CPU'S OTHER THAN THE KD11-A, THE VALUE OF SR1 WILL HAVE TO BE EMPIRACALLY DETERMINED BY RUNNING THIS MODULE ALONE AND NOTING THE INFORMATION PROVIDED IN THE ERROR PRINTOUT.

FOR AN 1134A 60 CYCLE SYSTEM THE TIMING PARAMETER IN SR1 SHOULD BE SET TO 000031.

6.0 DEVICE OPTION SET-UP

NONE REQUIRED

7.0 MODULE OPERATION

A. VERIFY THAT THE KW11-L MONITOR BIT CAN BE SET BY THE POWER SUPPLY SIGNAL. IF A FAILURE IS DETECTED, REPORT THE ERROR AND DROP THE MODULE.

B. VERIFY THAT THE KW11-L CAN GENERATE AN INTERRUPT TO THE PROPER VECTOR WHEN ENABLED. IF A FAILURE IS DETECTED REPORT THE ERROR AND DROP THE MODULE.

C. PERFORM THE TIMING ANALISIS:

1. SYNC THE LINE CLOCK
2. COUNT THE NO. OF ITERATIONS THROUGH A MONITOR "BREAK" LOOP WHILE WAITING FOR AN INTERRUPT.
3. STORE THE COUNT IN A TABLE
4. REPEAT (1) THRU (3) SIXTEEN TIMES TO STORE 16 COUNTS IN THE TABLE
5. SUM ALL ENTRIES IN THE TABLE AND DIVIDE BY 16 TO GET THE AVERAGE.
6. CHECK THAT THE AVERAGE IDS WITHIN LIMITS
7. IF WITHIN TOLERANCE, ENABLE SYSTEM TIMER TO BE ABLE TO RUN AND REPORT END OF PASS.
8. IF NOT WITHIN TOLERANCE, REPORT THE ERROR AND DROP THE MODULE. (SYSTEM TIMER NOT ENABLED TO RUN)

8.0 OPERATOR OPTIONS

"SR1" AND THE CONTENTS OF LOCATION "TOL" MAY HAVE TO BE MODIFIED FOR UNUSAL CASES TO COMPENSATE FOR A PARTICULAR SYSTEM INSTALLATION. [SR1] IS THE AVERAGE TIMER COUNT AND IF FOUND TO BE 000000 THE MODULE ASSUMES A DEFAULT OF "000037" (KD11-A CPU). IF NOT 000000, THE MODULE TAKES THE CONTENTS OF "SR1" AS THE AVERAGE TIME VALUE. THE LOCATION TAGGED "TOL" IS THE ALLOWABLE TOLERANCE AND IS PROGRAM LOADED AS A "3".

9.0 NON-STANDARD PRINTOUTS

```
IF A TIMING ANALYSIS ERROR IS DETECTED IT IS REPORTED
VIA AN EXTENDED ERROR PRINTOUT OF FOUR OCTAL NUMBERS:

WWWWWW SSSSSS HHHHHH LLLLLL

WHERE:  WWWWWW  IS THE WAS AVERAGE
        SSSSSS  IS WHAT THE AVERAGE SHOULD HAVE BEEN
        HHHHHH  IS THE HIGH LIMIT
        LLLLLL  IS THE LOW LIMIT
```

```
000000'  NBKMOD  <KWFB >,177546,100,6,,,1,162
000000'          MODULE  41000,KWFB ,177546,100,6,,,1,162
                 .TITLE  KWFB DEC/X11 SYSTEM EXERCISER MODULE
          ;      DDXCOM  VERSION  6      23-MAY-78
                                         .LIST    BIN
                                         ;*****************************************************************
000000'                                  BEGIN:
000000'  053513  041106      040         MODNAM: .ASCII  /KWFB / ;MODULE NAME.
000005'      000                         XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006'  177546                          ADDR:   177546+0                ;1ST DEVICE ADDR.
000010'  000100                          VECTOR: 100+0                   ;1ST DEVICE VECTOR.
000012'      300                         BR1:    .BYTE   PRTY6+0         ;1ST BR LEVEL.
000013'      000                         BR2:    .BYTE   PRTY+0          ;2ND BR LEVEL.
000014'  000001                          DVID1:  +1                      ;DEVICE INDICATOR 1.
000016'  000000                          SR1:    OPEN                    ;SWITCH REGISTER 1
000020'  000000                          SR2:    OPEN                    ;SWITCH REGISTER 2
000022'  000000                          SR3:    OPEN                    ;SWITCH REGISTER 3
000024'  000000                          SR4:    OPEN                    ;SWITCH REGISTER 4
                                         ;*****************************************************************
000026'  041000                          STAT:   41000                   ;STATUS WORD.
000030'  000224'                         INIT:   START                   ;MODULE START ADDR.
000032'  000224'                         SPOINT: MODSP                   ;MODULE STACK POINTER.
000034'  000000                          PASCNT: 0                       ;PASS COUNTER.
000036'  000001                          ICONT:  1                       ;# OF ITERATIONS PER PASS=1
000040'  000000                          ICOUNT: 0                       ;LOC TO COUNT ITERATIONS
000042'  000000                          SOFCNT: 0                       ;LOC TO SAVE TOTAL SOFT ERRORS
000044'  000000                          HRDCNT: 0                       ;LOC TO SAVE TOTAL HARD ERRORS
000046'  000000                          SOFPAS: 0                       ;LOC TO SAVE SOFT ERRORS PER PASS
000050'  000000                          HRDPAS: 0                       ;LOC TO SAVE HARD ERRORS PER PASS
000052'  000000                          SYSCNT: 0                       ;# OF SYS ERRORS ACCUMULATED
000054'  000000                          RANNUM: 0                       ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056'                                  CONFIG:                         ;RESERVED FOR MONITOR USE
000056'  000000                          RES1:   0                       ;RESERVED FOR MONITOR USE
000060'  000000                          RES2:   0                       ;RESERVED FOR MONITOR USE
000062'  000000                          SVR0:   OPEN                    ;LOC TO SAVE R0.
000064'  000000                          SVR1:   OPEN                    ;LOC TO SAVE R1.
000066'  000000                          SVR2:   OPEN                    ;LOC TO SAVE R2.
000070'  000000                          SVR3:   OPEN                    ;LOC TO SAVE R3.
000072'  000000                          SVR4:   OPEN                    ;LOC TO SAVE R4.
000074'  000000                          SVR5:   OPEN                    ;LOC TO SAVE R5.
000076'  000000                          SVR6:   OPEN                    ;LOC TO SAVE R6.
000100'  000000                          CSRA:   OPEN                    ;ADDR OF CURRENT CSR.
000102'                                  SBADR:                          ;ADDR OF GOOD DATA, OR
000102'  000000                          ACSR:   OPEN                    ;CONTENTS OF CSR.
000104'                                  WASADR:                         ;ADDR OF BAD DATA, OR
000104'  000000                          ASTAT:  OPEN                    ;STATUS REG CONTENTS.
000106'                                  ERRTYP:                         ;TYPE OF ERROR
000106'  000000                          ASB:    OPEN                    ;EXPECTED DATA.
000110'  000000                          AWAS:   OPEN                    ;ACTUAL DATA.
000112'  000322'                         RSTRT:  RESTRT                  ;RESTART ADDRESS AFTER END OF PASS
000114'  000000                          WDTO:   OPEN                    ;WORDS TO MEMORY PER ITERATION
000116'  000000                          WDFR:   OPEN                    ;WORDS FROM MEMORY PER ITERATION
000120'  000000                          INTR:   OPEN                    ;# OF INTERRUPTS PER ITERATION
000122'  000162                          IDNUM:  162                     ;MODULE IDENTIFICATION NUMBER=162
         000040                                  .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
                                                 .NLIST
```



```
                                         .WORD   0
                                         .LIST
                                         .ENDR
000224'                                  MODSP:
                                         ;*****************************************************************
                                         .GLOBL  CLOCK,CLOCKL,CLOCKP,LCLEAR,PCLEAR       ;MONITOR AND TIMER MODULE LOCATIONS

000224'  132767  000001  000000G START:  BITB    #BIT0,CLOCK             ;USING A KW11-L TIMER ?
000232'  001006                          BNE     1$                      ;BR IF YES
000234'  132767  000002  000000G         BITB    #BIT1,CLOCK             ;USING THE KW11-P TIMER ?
000242'  001010                          BNE     2$                      ;BR IF YES
000244'  104410  000000'                 ENDS,BEGIN                      ;
000250'  016700  000000G         1$:     MOV     CLOCKL,R0               ;GET POINTER TO TIMING INFORMATION
000254'  012767  000000G 000734          MOV     #LCLEAR,TCLEAR          ;SET PROPER JSR ADDR
000262'  000405                          BR      3$                      ;CONTINUE
000264'  016700  000000G         2$:     MOV     CLOCKP,R0               ;GET POINTER TO TIMING INFORMATION
000270'  012767  000000G 000720          MOV     #PCLEAR,TCLEAR          ;SET PROPER JSR ADDR
000276'  005720                  3$:     TST     (R0)+                   ;POINT TO LOCATION CONTAINING "TIME"
000300'  005010                          CLR     (R0)                    ;INIT RUN TIME = 000000
000302'  004777  000710                  JSR     PC,@TCLEAR              ; CLEAR MODULE PASSTIME TABLE IN CLOCK MODULE
000306'  005767  177504                  TST     SR1                     ;DOES SR1 SPECIFY A TIME ?
000312'  001003                          BNE     RESTRT                  ;BR IF IT DOES
000314'  012767  000037  177474          MOV     #37,SR1                 ;DEFAULT TO 37(9) - KD11-A CPU

000322'  016767  177460  177550  RESTRT: MOV     ADDR,CSRA               ;SAVE KW11-L ADDRESS
000330'  016700  177454                  MOV     VECTOR,R0               ;GET VECTOR ADDRESS
000334'  012720  001124'                 MOV     #KWINT,(R0)+            ;STEER KW INTR'S TO KWINT
000340'  116710  177446                  MOVB    BR1,(R0)
000344'  005077  177436                  CLR     @ADDR

                                         ;VERIFY THAT KW MONITOR BIT (BIT07) CAN BE SET BY POWER SUPPLY SIGNAL
                                         ;-----------------------------------------------------------------

000350'  005005                  KWDT1:  CLR     R5                      ;INIT TIMER
000352'                          1$:
000352'  104407  000000'                 BREAK$,BEGIN                    ;TEMPORARY RETURN TO MONITOR....
000356'  104407  000000'                 BREAK$,BEGIN                    ;THEN CONTINUE AT NEXT INSTRUCTION.
000362'  005305                          DEC     R5                      ;COUNT TIMER
000364'  001372                          BNE     1$                      ;BR UNTIL TIMER = 0
000366'  022777  000200  177412          CMP     #200,@ADDR              ;DID MONITOR BIT SET?
000374'  001415                          BEQ     KWDT2                   ;BR IF YES
000376'  017767  177404  177476          MOV     @ADDR,ACSR              ;SAVE CONTENTS OF LKCSR
000404'  012767  000025  177474          MOV     #25,ERRTYP              ;
                                         ;*****************************************************************
000412'  104405  000000' 000000          ;ERDER$,BEGIN,NULL              ;KW11-L MONITOR BIT WON'T SET
                                         ;*****************************************************************
000420'  105067  000000G                 CLRB    CLOCK                   ;DISABLE ALL TIMING CHECKS
000424'  104410  000000'                 ENDS,BEGIN                      ;

                                         ;VERIFY THAT KW11-L CAN GENERATE AN INTERRUPT PROPERLY
                                         ;-----------------------------------------------------

000430'  005067  000566          KWDT2:  CLR     KWFLG1                  ;CLEAR SOFTWARE INTR. FLAG
000434'  005005                          CLR     R5                      ;INIT TIMER
000436'  052777  000100  177342          BIS     #100,@ADDR              ;SET LKCSR INTR ENAB
```

```
000444' 042777  000200  177334          BIC     #200,@ADDR          ;CLEAR MONITOR BIT
000452' 005767  000544          1$:     TST     KWFLG1              ;DID KW11-L INTR OCCUR
000456' 100425                          BMI     2$                  ;BR IF YES
000460' 104407  000000'                 BREAK$,BEGIN                ;TEMPORARY RETURN TO MONITOR....
000464' 104407  000000'                 BREAK$,BEGIN                ;THEN CONTINUE AT NEXT INSTRUCTION.
000470' 005305                          DEC     R5                  ;COUNT TIMER
000472' 001367                          BNE     1$                  ;BR IF NO TIMEOUT
000474' 017767  177306  177400          MOV     @ADDR,ACSR          ;SAVE CONTENTS OF LKCSR
000502' 005077  177300                  CLR     @ADDR               ;CLEAR INTR ENAB
000506' 012767  000023  177372          MOV     #23,ERRTYP
                                        ;*******************************************************************
000514' 104405  000000' 000000          HRDERS,BEGIN,NULL           ;KW11-L FAILED TO INTR ON TIME
                                        ;*******************************************************************
000522' 105067  000000G                 CLRB    CLOCK               ;DISABLE ALL TIMING CHECKS
000526' 104410  000000'                 ENDS,BEGIN
000532' 032777  000100  177246  2$:     BIT     #100,@ADDR          ;DID INTR SERVICE CLEAR BIT 06?
000540' 001417                          BEQ     KWDT3               ;BR IF YES
000542' 017767  177240  177332          MOV     @ADDR,ACSR          ;SAVE CONTENTS OF LKCSR
000550' 005077  177232                  CLR     @ADDR               ;CLEAR LKCSR
000554' 012767  000025  177324          MOV     #25,ERRTYP
                                        ;*******************************************************************
000562' 104405  000000' 000000          HRDERS,BEGIN,NULL           ;KW11 INTR SERVICE FAILED TO CLR BIT06
                                        ;*******************************************************************
000570' 105067  000000G                 CLRB    CLOCK               ;DISABLE ALL TIMING CHECKS
000574' 104410  000000'                 ENDS,BEGIN

                                ;GROSS TIMING ANALYSIS ROUTINE - PERFORMS GROSS CPU/MEMORY TIMING ANALYSIS
                                ;-------------------------------------------------------------------------
                                ;USING THE KW11-L
                                ;----------------

000600' 004767  000332          KWDT3:  JSR     PC,CLRTAB           ;GO CLEAR LOOP COUNTERS TABLE
000604' 012701  001246'                 MOV     #CTRTAB,R1          ;R1 POINTS TO LOOP COUNTERS TABLE
000610' 005067  000406          KWSYNC: CLR     KWFLG1              ;INIT SOFTWARE INTR. FLAG
000614' 005000                          CLR     R0                  ;INIT TIMER
000616' 052777  000100  177162          BIS     #100,@ADDR          ;ENABLE INTRS
000624' 005767  000372          1$:     TST     KWFLG1              ;INTR OCCUR YET?
000630' 100425                          BMI     2$                  ;BR IF YES
000632' 104407  000000'                 BREAK$,BEGIN                ;TEMPORARY RETURN TO MONITOR....
000636' 104407  000000'                 BREAK$,BEGIN                ;THEN CONTINUE AT NEXT INSTRUCTION.
000642' 005300                          DEC     R0                  ;COUNT TIMER
000644' 001367                          BNE     1$                  ;BR IF NO TIMEOUT
000646' 017767  177134  177226          MOV     @ADDR,ACSR          ;SAVE CONTENTS OF LKCSR
000654' 005077  177126                  CLR     @ADDR               ;CLEAR OUT LKCSR
000660' 012767  000023  177220          MOV     #23,ERRTYP
                                        ;*******************************************************************
000666' 104405  000000' 000000          HRDERS,BEGIN,NULL           ;KW11-L TIMEOUT
                                        ;*******************************************************************
000674' 105067  000000G                 CLRB    CLOCK               ;DISABLE ALL TIMING CHECKS
000700' 104410  000000'                 ENDS,BEGIN
000704' 005067  000312          2$:     CLR     KWFLG1              ;INIT SOFTWARE INTR FLAG
000710' 005000                          CLR     R0                  ;INIT LOOP COUNTER
000712' 052777  000100  177066          BIS     #100,@ADDR          ;ENABLE KW11-L INTRS
000720' 005767  000276          3$:     TST     KWFLG1              ;INTR OCCUR YET?
000724' 100425                          BMI     4$                  ;BR IF YES
000726' 104407  000000'                 BREAK$,BEGIN                ;TEMPORARY RETURN TO MONITOR....
```



```
000732' 104407  000000'                 BREAK$,BEGIN                ;THEN CONTINUE AT NEXT INSTRUCTION.
000736' 005200                          INC     R0                  ;COUNT ONE PASS THRU BREAK LOOP
000740' 001367                          BNE     3$                  ;BR IF NO TIMEOUT
000742' 017767  177040  177132          MOV     @ADDR,ACSR          ;SAVE CONTENTS OF LKCSR
000750' 005077  177032                  CLR     @ADDR               ;CLEAR OUT LKCSR
000754' 012767  000023  177124          MOV     #23,ERRTYP
                                        ;*******************************************************************
000762' 104405  000000' 000000          HRDERS,BEGIN,NULL           ;KW11-L TIMEOUT
                                        ;*******************************************************************
000770' 105067  000000G                 CLRB    CLOCK               ;DISABLE ALL TIMING CHECKS
000774' 104410  000000'                 ENDS,BEGIN
001000' 010021                  4$:     MOV     R0,(R1)+            ;SAVE LOOP COUNT IN TABLE
001002' 022701  001306'                 CMP     #CTRTAB+40,R1       ;END OF TABLE ??
001006' 001300                          BNE     KWSYNC              ;GO DO IT AGAIN
001010' 004767  000140                  JSR     PC,TIMAVG           ;GO COMPUTE TABLE AVERAGE
001014' 010003                          MOV     R0,R3               ;R3 CONTAINS AVERAGE COUNT
001016' 016701  176774                  MOV     SR1,R1              ;GET WHAT AVERAGE COUNT SHOULD BE
001022' 010102                          MOV     R1,R2               ;PUT IT IN R2 TOO
001024' 066701  000170                  ADD     TOL,R1              ;R1 CONTAINS HIGH LIMIT COUNT
001030' 166702  000164                  SUB     TOL,R2              ;R2 CONTAINS LOW LIMIT COUNT
001034' 160301                          SUB     R3,R1               ;LESS THAN HIGH LIMIT?
001036' 103406                          BCS     5$                  ;BR IF NOT
001040' 160203                          SUB     R2,R3               ;GREATER THAN LOW LIMIT?
001042' 103404                          BCS     5$                  ;BR IF NOT
001044' 104413  000000'                 ENDIT$,BEGIN                ;SIGNAL END OF ITERATION.
                                                                    ;MONITOR SHALL TEST END OF PASS
001050' 000167  177246                  JMP     RESTRT
001054' 105067  000000G         5$:     CLRB    CLOCK               ;DISABLE USING THE TIMER MODULE
001060' 060001                          ADD     R0,R1               ;RESTORE HIGH LIMIT
001062' 012704  001236'                 MOV     #TIMES,R4           ;POINT TO SAVE AREA
001066' 010024                          MOV     R0,(R4)+            ;SAVE WAS COUNTER AVERAGE
001070' 016724  176722                  MOV     SR1,(R4)+           ;SAVE SHOULD BE COUNTER AVERAGE
001074' 010124                          MOV     R1,(R4)+            ;SAVE HIGH LIMIT
001076' 010224                          MOV     R2,(R4)+            ;SAVE LOW LIMIT
001100' 017767  176702  176774          MOV     @ADDR,ACSR          ;SAVE CONTENTS OF LKCSR
001106' 005067  176774                  CLR     ERRTYP
                                        ;*******************************************************************
001112' 104405  000000' 001224'         HRDERS,BEGIN,KWMSG          ;PRINT COUNTER INFORMATION
                                        ;*******************************************************************
001120' 104410  000000'                 ENDS,BEGIN

                                ;GENERAL PURPOSE UTILITY ROUTINES
                                ;*********************************

                                ;INTERRUPT SERVICE ROUTINE
                                ;-------------------------

001124' 005077  176656          KWINT:  CLR     @ADDR               ;CLEAR OUT LKCSR
001130' 005167  000066                  COM     KWFLG1              ;SET SOFTWARE INTR. FLAG
001134' 000002                          RTI                         ;RETURN TO MAINLINE

                                ;ROUTINE TO CLEAR LOOP COUNTER TABLE
                                ;-----------------------------------

001136' 012701  001246'         CLRTAB: MOV     #CTRTAB,R1          ;POINT TO FIRST TABLE ENTRY
001142' 005021                  1$:     CLR     (R1)+               ;CLEAR ONE WORD
```

```
001144' 022701  001306'          CMP     #CTRTAB+40,R1   ;DONE 16 WORDS?
001150' 001374                   BNE     1$              ;BR IF NOT
001152' 000207                   RTS     PC              ;RETURN TO TIMING ANALYSIS ROUTINE

                                 ;ROUTINE TO COMPUTE AVERAGE LOOP COUNT
                                 ;-----------------------------------

001154' 012701  001246'          TIMAVG: MOV     #CTRTAB,R1      ;R1 POINTS TO 1ST TABLE ENTRY
001160' 005002                           CLR     R2              ;INIT 32 BIT ACCUMULATOR
001162' 005003                           CLR     R3
001164' 062102                   1$:     ADD     (R1)+,R2        ;ADD TABLE ENTRY TO LOW ACCUM.
001166' 005503                           ADC     R3              ;ADD IN ANY CARRIES TO HI ACCUM.
001170' 022701  001306'                  CMP     #CTRTAB+40,R1   ;SUMMED ALL 16 ENTRIES?
001174' 001373                           BNE     1$              ;BR IF NOT
001176' 005000                           CLR     R0              ;INIT R0 TO ACCUMULATE AVERAGE
001200' 162702  000020           2$:     SUB     #16.,R2         ;SUBTRACT CONSTANT 16(10)
001204' 005603                           SBC     R3              ;BORROW IF REQUIRED
001206' 103402                           BCS     3$              ;BR IF OVER SUBTRACT - (WE'RE DONE)
001210' 005200                           INC     R0              ;COUNT NO. OF TIMES 16 WENT IN
001212' 000772                           BR      2$              ;GO SUBTRACT ANOTHER 16(10)
001214' 000207                   3$:     RTS     PC              ;RETURN - AVERAGE IN R0

                                 ;CONSTANTS, VARIABLES, AND MISCELLANEOUS DATA AREAS
                                 ;-----------------------------------------------

001216' 000000                   TCLEAR: 0                       ;HOLDS SUBROUTINE ADDR.
001220' 000003                   TOL:    3                       ;TOLERANCE FOR LOOP COUNTER AVG.
001222' 000000                   KWFLG1: 0                       ;SOFTWARE INTERRUPT FLAG
001224' 001236'                  KWMSG:  TIMES                   ;POINTS TO WAS AVERAGE
001226' 001240'                          TIMES+2                 ;POINTS TO SHOULD BE AVERAGE
001230' 001242'                          TIMES+4                 ;POINTS TO HIGH LIMIT
001232' 001244'                          TIMES+6                 ;POINTS TO LOW LIMIT
001234' 177777                           177777                  ;TABLE TERMINATOR

001236' 000004                   TIMES:  .BLKW   4.              ;RESERVE FOUR WORDS FOR ERROR INFO.

                                 ;16. WORD TABLE TO STORE SIXTEEN LOOP COUNTS
                                 ;--------------------------------------------

001246' 000020                   CTRTAB: .BLKW   16.

        000001                   .END
```



```
ACSR     000102R          214#    269*    290*    300*    325*    342*    373*
ADDR     000006R          180#    252     256*    267     269     282*    283*    290     291*    298     300     301*    318*
                          325     326*    335*    342     343*    373     386*
ADDR22=  001000           232#
ASB      000106R          218#
ASTAT    000104R          216#
AWAS     000110R          219#
BEGIN    000000R          177#    239     263     264     272     275     286     287     294     297     304     307     321
                          322     329     332     338     339     346     349     363     376     378
BIT0   = 000001           232#    235
BIT1   = 000002           232#    237
BIT10  = 002000           232#
BIT11  = 004000           232#
BIT12  = 010000           232#
BIT13  = 020000           232#
BIT14  = 040000           232#
BIT15  = 100000           232#
BIT2   = 000004           232#
BIT3   = 000010           232#
BIT4   = 000020           232#
BIT5   = 000040           232#
BIT6   = 000100           232#
BIT7   = 000200           232#
BIT8   = 000400           232#
BIT9   = 001000           232#
BREAK$=  104407           232#    263     264     286     287     321     322     338     339
BR1      000012R          182#    255
BR2      000013R          183#
BTOD$  = 104421           232#
CDATA$=  104412           232#
CLOCK  = ****** G         233#    235     237     274*    296*    306*    331*    348*    366*
CLOCKL=  ****** G         233#    240
CLOCKP=  ****** G         233#    243
CLRTAB   001136R          314     393#
CONFIG   000056R          202#
CSRA     000100R          212#    252*
CTRTAB   001246R          315     351     393     395     402     407     434#
DATCK$=  104411           232#
DATER$=  104404           232#
DVID1    000014R          184#
ENDIT$=  104413           232#    363
END$   = 104410           232#    239     275     297     307     332     349     378
ERRTYP   000106R          217#    270*    292*    302*    327*    344*    374*
EXIT$  = 104400           232#
GETPA$=  104415           232#
GWBUF$=  104414           232#
HRDCNT   000044R          197#
HRDER$=  104405           232#    272     294     304     329     346     376
HRDPAS   000050R          199#
ICONT    000036R          194#
ICOUNT   000040R          195#
IDNUM    000122R          224#
INIT     000030R          191#
INTR     000120R          223#
KWDT1    000350R          261#
KWDT2    000430R          268     280#
```

```
KWDT3   000600R          299     314#
KWFLG1  001224R          280*    284     316*    319     333*    336     387*    422#
KWINT   001124R          254     386#
KWMSG   001224R          376     423#
KWSYNC  000610R          316#    352
LCLEAR= ****** G         233#    241
MAP22$= 104416           232#
MODNAM  000000R          178#
MODSP   000224R          192     230#
MSGW$ = 104403           232#
MSGS$ = 104402           232#
MSG$  = 104401           232#
NULL  = 000000           232#    272     294     304     329     346
OPEN  = 000000           179     195     186     197     188     205     206     207     208     209     210     211     212
                         214     216     218     219     221     222     223     232#
OTOAS = 104420           232#
PASCNT  000054R          193#
PCLEAR= ****** G         233#    244
PIRQS = 000004           232#
POPSP = 005726           232#
POPSP2= 022626           232#
PRTV  = 000000           183     232#
PRTV0 = 000000           232#
PRTV1 = 000040           232#
PRTV2 = 000100           232#
PRTV3 = 000140           232#
PRTV4 = 000200           232#
PRTV5 = 000240           232#
PRTV6 = 000300           182     232#
PRTV7 = 000340           232#
PS    = 177776           232#
PSW   = 177776           232#
PUSH  = 005746           232#
PUSH2 = 024646           232#
RANDS = 104417           232#
RANNUM  000054R          201#
RESTRT  000322R          220     249     252#    365
RES1    000056R          203#
RES2    000060R          204#
RSTRT   000112R          220#
SBADR   000102R          213#
SOFCNT  000042R          196#
SOFERS= 104406           232#
SOFPAS  000046R          198#
SPOINT  000032R          192#
SPSIZ = 000040             1#    225
SR1     000016R          185#    248     250*    355     370
SR2     000020R          186#
SR3     000022R          187#
SR4     000024R          188#
START   000224R          191     235#
STAT    000026R          190#
SVR0    000062R          205#
SVR1    000064R          206#
SVR2    000066R          207#
SVR3    000070R          208#
```



```
SVR4    000072R          209#
SVR5    000074R          210#
SVR6    000076R          211#
SYSCNT  000052R          200#
TCLEAR  001216R          241*    244*    247     420#
TIMAVG  001154R          353     402#
TIMES   001236R          368     423     424     425     426     429#
TOL     001220R          357     358     421#
TRPDFD= 000022           232#
VECTOR  000010R          181#    253
WASADR  000104R          215#
WDFR    000116R          222#
WDTO    000114R          221#
XFLAG   000005R          179#
.     = 001306R          429#    434#

. ABS.  000000     000
        001306     001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XKWFB0,XKWFB0/SOL/CRF:SYM=DDXCOM,XKWFB0
 RUN-TIME: 1 1 .2 SECONDS
 RUN-TIME RATIO: 26/3=7.7
 CORE USED:  7K  (13 PAGES)
```

```
XX      XX    KK      KK    WW      WW    FFFFFFFFFF    BBBBBBBB        000000
XX      XX    KK      KK    WW      WW    FFFFFFFFFF    BBBBBBBB        000000
XX      XX    KK      KK    WW      WW    FF            BB      BB    00      00
XX      XX    KK      KK    WW      WW    FF            BB      BB    00      00
  XX  XX      KK    KK      WW      WW    FF            BB      BB    00    0000
  XX  XX      KK    KK      WW      WW    FF            BB      BB    00    0000
    XX        KKKKKK        WW      WW    FFFFFFFF      BBBBBBBB      00  00  00
    XX        KKKKKK        WW      WW    FFFFFFFF      BBBBBBBB      00  00  00
  XX  XX      KK    KK      WW  WW  WW    FF            BB      BB    0000    00
  XX  XX      KK    KK      WW  WW  WW    FF            BB      BB    0000    00
XX      XX    KK      KK    WWWW  WWWW    FF            BB      BB    00      00
XX      XX    KK      KK    WWWW  WWWW    FF            BB      BB    00      00
XX      XX    KK      KK    WW      WW    FF            BBBBBBBB        000000
XX      XX    KK      KK    WW      WW    FF            BBBBBBBB        000000

  SSSSSSSS    EEEEEEEEEE      QQQQQQ
  SSSSSSSS    EEEEEEEEEE      QQQQQQ
SS            EE            QQ      QQ
SS            EE            QQ      QQ
SS            EE            QQ      QQ
SS            EE            QQ      QQ
  SSSSSS      EEEEEEEE      QQ      QQ
  SSSSSS      EEEEEEEE      QQ      QQ
        SS    EE            QQ  QQ  QQ
        SS    EE            QQ  QQ  QQ
        SS    EE            QQ    QQ
        SS    EE            QQ    QQ
SSSSSSSS      EEEEEEEEEE      QQQQ  QQ
SSSSSSSS      EEEEEEEEEE      QQQQ  QQ

*************************************************************************
* FOR REMOTE SITE PROBLEMS CONTACT THE I.P.C. NETWORK TEAM AT EX 5955...
*************************************************************************
 *  *  *  *  *    L I S T I N G   P R O B L E M S ?    *  *  *  *  *  *
* If you have problems/questions with this listing, please              *
*  A. Check the applicable items.                                        *
*      ___ Do not need this listing, please discontinue                  *
*      ___ Interested in receiving listing on microfiche                 *
*      ___ Interested in receiving listing on 8 1/2 X 11 Xerox paper     *
*      ___ Address is wrong or incomplete (Enter correctly below)        *
*      ___ Have other problem/question; not sure whom to contact         *
*  B. Identify yourself and the listing.          Phone:                *
*      Name:                              Location (mail stop):          *
*                                                                        *
*      CC #:        Listing name:                   Listing #:           *
*                                                                        *
*  C. Mail entire page to: Corporate Information Services  PK3-2/F34    *
 *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *

LPTSPL version 102(2263)/3(61) running on LPT061
 *START* User BEAUSOLEIL, [400,20433]  Job  XKWFB0 Seg. 7423 Date 23-Oct-78 17:12:16 Monitor IPC-F 603 [6P8] *START*
/TO:ML21-4:BEAUSO -- Distribution to ML21-4, slot 132
File: CUFF:XKWFB0.SEQ<057>[400,20433] Created: 12-Oct-78 16:48:00 Printed: 23-Oct-78 17:13:14
QUEUE Switches:  /PRINT:ARROW /FILE:ASCII /COPIES:2 /SPACING:1 /LIMIT:138 /FORMS:UNRDK
```