EXAMPLES

FILEID**LABIOSTAT

LABIOSTAT
FOR

```
!File:  LABIOSTAT.FOR
        Version 'V04-000'

!**************************************************************************
!*                                                                        *
!*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                              *
!*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.               *
!*   ALL RIGHTS RESERVED.                                                 *
!*                                                                        *
!*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
!*   ONLY IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
!*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
!*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
!*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
!*   TRANSFERRED.                                                         *
!*                                                                        *
!*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
!*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
!*   CORPORATION.                                                         *
!*                                                                        *
!*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
!*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.              *
!*                                                                        *
!**************************************************************************
!
        Program LABIO_STATUS
! This is a utility routine for the LABIO  system. It displays
! the status of all 16 channels of the A/D. It assumes that
! the terminal is a VT52 or an equivalent, e.g VT100 in VT52 mode.
! The display is update once every 1-9 seconds. Default is
! one second. There are 5 commands associated with the program
!
        C - display status of 16 channels
        P - display status by process PID
        H - display help frame (timeouts after 1 min.)
        E - Exit to VMS DCL
        Digit(1-9) Change cycle time.
!
! The key pad can also be used to enter commands. The special function
! Keys on the VT52 or VT100 correspond to the first 4 commands (3 on VT52).
!
! Typing ANY key will cause a display refresh.


        Include 'LABCHNDEF.FOR'

        Character*10 STATUS(4)
        Character*8 XTIME
        Character*9 XDATE
        Parameter COMMAND_MAX = 4
        Character*1 COMMAND,COMMAND_TABLE(COMMAND_MAX,2),ESCAPE,TERMINATOR
        Character*63 COMMAND_DEV

        External SS$_NOTRAN,SS$_NORMAL,SS$_PARTESCAPE
        External IO$M_CVTLOW,IO$M_NOECHO,IO$M_TIMED,IO$_READVBLK,IO$M_PURGE
```

C 13

```
        Logical SUCCESS,SYS$QIOW,SYS$ASSIGN
        Integer CHANNEL,DISPLAY_FLAG,OLD_DISPLAY,COMMAND_CHAN
        Integer DEF_TIME_OUT,TIME_OUT
        Byte    ERASE_SCREEN(2),HOME(2),ERASE_LINE(2),VT52_MODE(7)
        Integer*2 IO_STATUS(4),CHAR_COUNT
        Equivalence (ESCAPE,HOME),(CHAR_COUNT,IO_STATUS(2))

! VT52 control ESCAPE Sequences

        Data HOME,ERASE_SCREEN,ERASE_LINE
       1    /'33'O,'H','33'O,'J','33'O,'K'/

! VT100 control ESCAPE sequences
! This ESC seq places a VT100 in VT52 mode

        Data VT52_MODE/'33'O,'[','?','2','l','33'O,']'/

        Data STATUS/'Unknown ','Inactive',' Active ','  '/
        Data COMMAND_TABLE/'C','P','E','H','P','Q','S','R'/
        Data DISPLAY_FLAG,ERASE_FLAG /1,.TRUE./
        Data DEF_TIME_OUT /1/

! Map to the GLOBAL DATA section created by the I/O program

        Call LABIO_INIT(0)

! Place VT100's in VT52 mode

        Type 500, VT52_MODE

! Initialize Command input channel
! We will read the command via a QIOW with a 1 sec timeout
! Commands are single character, to simplify matters we will
! read with no echo and convert lower to upper case.

        Call SYS$ASSIGN( 'TT',COMMAND_CHAN,,)
        QIO_READ = %Loc(IO$M_NOECHO) + %Loc(IO$M_CVTLOW) + %Loc(IO$M_TIMED)
       1 + %Loc(IO$_READVBLK)
        TT_PURGE = %Loc(IO$M_PURGE)
        Go To 25                      ! Display Something

! Get a command from the user, but only wait a short time (TIME_OUT)
! so we can update the screen. The input buffer is purged if a command
! was decode on the last read. (Prevents unnecessary erase loops)

20      DISPLAY_FLAG = OLD_DISPLAY  !Default is last display
        TIME_OUT = DEF_TIME_OUT     !Default time out

21      TABLE_INDEX = 1             !Assume no escape sequence

22      Call SYS$QIOW(,%Val(COMMAND_CHAN),%Val(QIO_READ+PURGE),
       1 IO_STATUS,,,%Ref(COMMAND),%Val(1),%Val(TIME_OUT),,,,)
        PURGE = 0
```

```
! If escape seq., set command table pointer to second table and
! get character following escape.
        TERMINATOR = Char( IO_STATUS(3) )
        If( TERMINATOR .ne. ESCAPE ) Go To 23
        TABLE_INDEX = 2
        Go To 22            !Get char following escape

23      If( CHAR_COUNT .ne. 0) Then      ! Char count not 0
! Check for char 1-9
        If( COMMAND .ge. '0' .and. COMMAND .le. '9' ) Then
            DEF_TIME_OUT = Ichar ( COMMAND )  - Ichar( '0' )
! Not 1-9 try a command.
        Else
            ERASE_FLAG = .true.           ! Screen erase
            Do 24 I = 1,COMMAND_MAX
            If( COMMAND .eq. COMMAND_TABLE(I,TABLE_INDEX)) DISPLAY_FLAG = I
24          Continue
        End If
        PURGE = TT_PURGE               !Purge the input buffer next time
        End If
!
! Get date and time, then dispatch to display routine
!
25      Call DATE (XDATE)
        Call TIME (XTIME)

        Go to (50,60,99,40) DISPLAY_FLAG
!
! Refresh the screen (Erase and Redisplay)
!
30      DISPLAY_FLAG = OLD_DISPLAY      !Redisplay last display
        ERASE_FLAG = .true.
        Go To 25
!
! Display the HELP frame, set the temporary time-out to 1 minute
!
40      Type 600, HOME,ERASE_SCREEN      !Display the help frame
        TIME_OUT = 60                    !Give the user 1 minute to read it
        DISPLAY_FLAG = OLD_DISPLAY       !When it times out, default old
        ERASE_FLAG = .true.
        Go To 21
!
! Generate the Status Line for each A/D channel
!
50      If ( ERASE_FLAG ) Type 300, HOME,ERASE_SCREEN
        Type 100, HOME,XTIME,XDATE
        CHANNEL_COUNT = 0
        Do 51 CHANNEL = 1,MAX_AD_CHANNEL
        If( AD_BLOCK(2,CHANNEL) .ne. 0 ) Then    !If allocated, display info
          Type 200,CHANNEL, STATUS(AD_BLOCK(1,CHANNEL)+i),
        1 (AD_BLOCK(J,CHANNEL), J = 2,6 )
          CHANNEL_COUNT = CHANNEL_COUNT + 1
        Else                             !If not allocated, say so
          Type 900, CHANNEL,'<Unused>',ERASE_LINE
        End If
51      Continue
```

```
        PID_COUNT = 0
        Do 52 PID_INDEX = 1, MAX_PID
        PID = CONNECT_BLOCK(PID_INDEX,1)
        If ( PID .ne. 0 ) PID_COUNT = PID_COUNT + 1
52      Continue

        Type 400,ERASE_LINE, PID_COUNT,CHANNEL_COUNT
        OLD_DISPLAY = DISPLAY_FLAG
        ERASE_FLAG = .false.
        Go to 20
!
! Status display via process (PID)
!
60      If ( ERASE_FLAG ) Type 300, HOME,ERASE_SCREEN
        Type 100, HOME,XTIME,XDATE
        PID_COUNT = 0                    ! Number of connected processess
        CHANNEL_COUNT = 0               ! Number of allocated channels
        Do 61 PID_INDEX = 1, MAX_PID
        PID = CONNECT_BLOCK(PID_INDEX,1)
        If ( PID .ne. 0 ) Then
        PID_COUNT = PID_COUNT + 1
        OLD_COUNT = CHANNEL_COUNT
          Do 62 CHANNEL = 1, MAX_AD_CHANNEL
          If( AD_BLOCK( 2,CHANNEL) .eq. PID ) Then  !If right PID, display info
            Type 200, CHANNEL, STATUS(AD_BLOCK(1,CHANNEL)+1),
     1      (AD_BLOCK(J,CHANNEL), J = 2,6 )
            CHANNEL_COUNT = CHANNEL_COUNT + 1
          End IF
62        Continue
        If (OLD_COUNT .eq. CHANNEL_COUNT ) Type 800, '<None>',PID,ERASE_LINE
        End IF
61      Continue
        Type 400,ERASE_LINE,PID_COUNT,CHANNEL_COUNT,ERASE_SCREEN
        OLD_DISPLAY = DISPLAY_FLAG
        ERASE_FLAG = .false.
        Go to 20
!
! Exit
!
99      Call Exit
!
! Format Statments
!
100     Format(1X,2A1,'           Lab IO Status as of ',A,' ',A//
     1' Channel    Status         PID     Tics/Sample  Buffer Size
     1       Buffers  '/)

200     Format(I5,5x,A8,Z10,4I12)

300     Format(' ',4A1)

400     Format(' '2A1/' Totals: ',I2,' Processes connected   ',I2,' Channels
     1 allocated'/'                     <Type an H for help>'2A1$)
```

```
500      Format(' '7A1)

600      Format(' '4A1/
        1' The following commands are available:'//
        1'     VT100   VT52      any'/
        1'     ------  ----      ---'/
        1'     PF1     red       C     Channel Display'/
        1'     PF2     blue      P     Process Display'/
        1'     PF3     grey      H     Help Display'/
        1'     PF4     n/a       E     Exit'//
        1' To change display time, type a digit 0-9 for the desired time'//)
700      Format(A)

800      Format(' ',A6,11X,Z10,2A1)

900      Format(I5,5x,A8,2A1)

        End
![End of File]
```

XALINK
MAR

LABIOLINK
COM

DRMASTER
FOR

LPATEST
FOR

LABIOPEAK
FOR

LABIOSTRT
COM

XAMESSAGE
MAR

XATEST
FOR

LABIOCOM
FOR

LBRDEMO
COM

LABMBXDEF
FOR

LABIOSAMP
FOR

MAILCOMPRESS
COM

LABCHNDEF
FOR

CONNECT
COM

LABIOCON
FOR

LBRDEMO
FOR

PEAK
FOR

DRCOPYBLD
COM

XIDRIVER
MAR

LABIOSEC
FOR

DRSLAVE
FOR

LABIOACQ
FOR

LABIOCOMP
COM

LABIOSTAT
FOR

TESTLABIO
FOR