Table A22 List of the file, **wall.plastic.src.**
This is a "fleshed out" version of SUBROUTINE WALL
valid for an elastic-plastic material. Compare with
Table a20 to see what “fleshing out” the STAGS user did.

```
======================================================================
C=Deck    wall
c=purpose User-written WALL subroutine
c=version May 2002
c=This particular version is for an isogrid-stiffened
c=torispherical head optimized by GENOPT.
c=The isogrid stiffeners are internal and smeared.
c=The shell skin is layer 2; the internal isogrid is layer1.
c=The skin thickness and isogrid height vary in the meridional
c=direction only. (Meridional direction=XYs(1) coordinate.)

#include "keydefs.h"

#if   _usage_
*
*     calling sequence: call WALL ( iunit, ielt, kelt, XYZg, XYs,
*                                   zeta,  ecz,  ilin, iplas )
*
*     Input Arguments:
*     ================
*     iunit = unit number
*     ielt  = local element number (in unit iunit)
*     kelt  = element type code
*     XYZg  = {x,y,z} global coordinates
*     XYs   = {s,t} surface coordinates (shell unit, only)
*
*     Output Arguments:
*     =================
*     zeta  = zeta (see M-5 or T-3 for details)
*     ecz   = eccentricity (see M-5 or T-3 for details)
*     ilin  = nonlinearity flag
*     iplas = plasticity flag
*
#endif

************************************************************************
            subroutine WALL ( iunit, ielt, kelt, XYZg, XYs,
     &                        zeta,  ecz,  ilin, iplas )
************************************************************************

      _implicit_none_

      Integer    iunit
      Integer    ielt
```

```
       Integer    kelt
       Integer    ilin
       Integer    iplas
       Real       XYZg(3)
       Real       XYs(2)
       Real       zeta
       Real       ecz

       Integer         maxLAY
       PARAMETER     ( maxLAY = 100 )

       Integer         maxSM
       PARAMETER     ( maxSM  = 6 )

       Integer         nit,  not
       common /nitnot/ nit,  not

       Integer         itaw, kwall, nlay, nlip, nsmrs
       common /WALLX / itaw, kwall, nlay, nlip, nsmrs

       Integer         matL (maxLAY)
       Real            tL   (maxLAY)
       Real            zetL (maxLAY)
       Integer         lsoL (maxLAY)
       Real            e1L  (maxLAY)
       Real            u12L (maxLAY)
       Real            gL   (maxLAY)
       Real            rhoL (maxLAY)
       Real            a1L  (maxLAY)
       Real            e2L  (maxLAY)
       Real            a2L  (maxLAY)
       common /WALL1 / matL, tL,   zetL, lsoL, e1L, u12L,
     &                 gL,   rhoL, a1L,  e2L,  a2L

       Integer         matF, matM
       Real            ttL   (maxLAY)
       Real            xxL   (maxLAY)
       Real            zetwL (maxLAY)
       Real            oL    (maxLAY)
       Real            eF,   uF,   rhoF,  alF
       Real            eM,   uM,   rhoM,  alM
       common /WALL2 / matF, matM,
     &                 ttL,  xxL,  zetwL, oL,
     &                 eF,   uF,   rhoF,  alF,
     &                 eM,   uM,   rhoM,  alM

       Integer         matC, matS
       Real            ct,   cc,   ch,   cd,   cb
```

```
      Real             ts,   phi,  anc
      Real             eC,   uC,   rhoC, alC
      Real             eS,   uS,   rhoS, alS
      common /WALL3 / matC, matS,
     &                 ct,   cc,   ch,   cd,   cb,
     &                 ts,   phi,  anc,
     &                 eC,   uC,   rhoC, alC,
     &                 eS,   uS,   rhoS, alS

      Integer          ta,   mat,  itvs, idumt
      Real             ccc,        cts
      common /WALL4 / ta,   mat,  itvs, idumt,
     &                 ccc(6,6),   cts(2,2)

      Integer          icroSM (maxSM)
      Real             spaSM  (maxSM)
      Real             zetSM  (maxSM)
      Real             xsiSM  (maxSM)
      Real             eczSM  (maxSM)
      common /SMEAR / icroSM, spaSM, zetSM, xsiSM, eczSM
C
      Real SPACNG,EMATL,DNMATL,PHDIFF,XDIFF,RATIO,TDIFF,HDIFF,TATX,HATX
      Real THSKIN,THKSTF,HEIGHT,PHORIG,SARCLT
      Integer I5,NSEG,ISEG,JSEG,I5I,I,IMORE,IMORE1
      COMMON/ISEGX1/PHORIG(100,30),SARCLT(100,30)
      COMMON/ISEGX2/THSKIN(100,30),THKSTF(100,30),HEIGHT(100,30)
      COMMON/ISEGX3/I5(30)
      COMMON/ISEGX4/SPACNG,EMATL,NUMATL,DNMATL
      REAL NUMATL
      CHARACTER*38 WORD1,WORD2,WORD3,WORD4,WORD5,WORD6,WORD7,WORD8
      CHARACTER*2 WORD3B
C23456789012345678901234567890123456789012345678901234567890123456789012
C
      character        filnam*33
      integer          iw,       ios,      itime
      data             iw    / 61 /
      data             itime / -1 /

c-----------------------------------------------------------------------
c     1st time enter, open the wall thickness file (iw)
c     read the data therein
c     and fill common blocks ISEGX1, ISEGX2, ISEGX3
c-----------------------------------------------------------------------
      if ( itime .lt. 0 ) then

        filnam  =  'WALLTHICK.STAGS'

        open ( unit=iw, name=filnam, access='SEQUENTIAL',
```

```
     $         form='FORMATTED', iostat=ios)

        if (ios .ne. 0) then
          write(not,3000) iw, filnam, ios
 3000     format (/,'*****ERROR in routine WALL(---) *****',
     $            /,'tried to open file:  iw = ',I4,'   name = ',A,
     $            /,'error return (iostat) = ',I12,/)
          call exit
        endif
c-----------------------------------------------------------------------
C  Retrieve angle, PHORIG and arc length SARCLT (X-coordinates),
C  shell skin thickness THSKIN, stringer thickness, THKSTF, and
C  stringer height, HEIGHT
C
        WORD1 = '     Number of shell segments (units)='
        WORD2 = '   Isogrid spacing,modulus,nu,density='
        WORD3 = '                 Nodal points in Segment'
        WORD3B= '= '
        WORD4 = '                   Angle (X-coordinate)='
        WORD5 = ' Meridional arc length (X-coordinate)='
        WORD6 = '                   Shell skin thickness='
        WORD7 = '          Stringer (or isogrid)  height='
        WORD8 = '     Stringer (or isogrid)  thickness='
        READ(iw,'(/,A38,I4)') WORD1,NSEG
        READ(iw,'(/,A38,1P,4E14.6)')
     1   WORD2,SPACNG,EMATL,NUMATL,DNMATL
        DO 3 ISEG = 1,NSEG
           READ(iw,'(/,A38,I3,A2,I4)') WORD3,JSEG,WORD3B,I5I
           I5(ISEG) = I5I
           READ(iw,'(/,A38,/(1P5E14.6))') WORD4,(PHORIG(I,ISEG),I=1,I5I)
           READ(iw,'(/,A38,/(1P5E14.6))') WORD5,(SARCLT(I,ISEG),I=1,I5I)
           READ(iw,'(/,A38,/(1P5E14.6))') WORD6,(THSKIN(I,ISEG),I=1,I5I)
           READ(iw,'(/,A38,/(1P5E14.6))') WORD7,(HEIGHT(I,ISEG),I=1,I5I)
           READ(iw,'(/,A38,/(1P5E14.6))') WORD8,(THKSTF(I,ISEG),I=1,I5I)
    3   CONTINUE
C234567890123456789012345678901234567890123456789012345678901234567890
C
c  Test SUBROUTINE WALL (remove the following statements later)
        rewind iw
        WRITE(not,'(/,A38,I4)')
     1  '     Number of shell segments (units)=',NSEG
        WRITE(not,'(/,A38,1P,4E14.6)')
     1  '   Isogrid spacing,modulus,nu,density=',
     1      SPACNG,EMATL,NUMATL,DNMATL
        DO 20 ISEG = 1,NSEG
           I5I = I5(ISEG)
           WRITE(not,'(/,A38,I3,A2,I4)')
     1'               Nodal points in Segment',ISEG,' =',I5I
```

```
            WRITE(not,'(/,A38,/(1P5E14.6))')
     1'                 Angle (X-coordinate)=', (PHORIG(I,ISEG),I=1,I5I)
            WRITE(not,'(/,A38,/(1P5E14.6))')
     1' Meridional arc length (X-coordinate)=', (SARCLT(I,ISEG),I=1,I5I)
            WRITE(not,'(/,A38,/(1P5E14.6))')
     1'                 Shell skin thickness=', (THSKIN(I,ISEG),I=1,I5I)
            WRITE(not,'(/,A38,/(1P5E14.6))')
     1'        Stringer (or isogrid)  height=', (HEIGHT(I,ISEG),I=1,I5I)
            WRITE(not,'(/,A38,/(1P5E14.6))')
     1'     Stringer (or isogrid)  thickness=', (THKSTF(I,ISEG),I=1,I5I)
   20   CONTINUE
C23456789012345678901234567890123456789012345678901234567890123456789012
c
        CLOSE(UNIT=iw)
C
        itime  =  0

      endif
C
c  Fill common block WALLX:
      itaw = 0
      kwall = 1
      nlay  = 2
c     nlip = 0
      nlip = 5
      nsmrs = 0
c
c  Fill "A" output except ecz (which is a function of X)
      zeta = 0.
      ilin = 0
c     iplas = 0
      iplas = 1
c
c  Find thickness, stiffener height at shell coordinate, X:
c  thickness at X = TATX;  stiffener height at X = HATX
      I5I = I5(iunit)
      DO 10 I = 2,I5I
         IF (XYs(1).LT.PHORIG(I,iunit)) THEN
            IMORE = I
            GO TO 11
         ENDIF
   10 CONTINUE
   11 CONTINUE
      IMORE1 = IMORE - 1
      PHDIFF = PHORIG(IMORE,iunit) - PHORIG(IMORE1,iunit)
      XDIFF  = XYs(1) - PHORIG(IMORE1,iunit)
      RATIO  = XDIFF/PHDIFF
      TDIFF  = THSKIN(IMORE,iunit) - THSKIN(IMORE1,iunit)
```

```
      HDIFF  = HEIGHT(IMORE,iunit) - HEIGHT(IMORE1,iunit)
      TATX   = THSKIN(IMORE1,iunit) + RATIO*TDIFF
      HATX   = HEIGHT(IMORE1,iunit) + RATIO*HDIFF
c
c   Find ecz
      ecz = (TATX + HATX)/2. - HATX
c
c   Fill common block WALL1
c     matL(1) = 0
c     matL(2) = 0
      matL(1) = 2
      matL(2) = 1
      tL(2)   = TATX
      tL(1)   = HATX
      zetL(1) = 0.
      zetL(2) = 0.
      lsoL(1) = 0
      lsoL(2) = 0
      e1L(2)  = EMATL
      e1L(1)  = EMATL*THKSTF(1,iunit)/SPACNG
      u12L(2) = NUMATL
      u12L(1) = 1./3.
      gL(2)   = EMATL/(2.*(1.+NUMATL))
      gL(1)   = e1L(1)/(2.*(1.+u12L(1)))
      rhoL(2) = DNMATL
      rhoL(1) = DNMATL*THKSTF(1,iunit)/SPACNG
      a1L(1)  = 0.
      a1L(2)  = 0.
      e2L(1)  = e1L(1)
      e2L(2)  = e1L(2)
      a2L(1)  = 0.
      a2L(2)  = 0.
c
c     WRITE(6,'(/,A,1P,3E12.4)')' TATX,HATX,ecz=',TATX,HATX,ecz
c     WRITE(6,'(/,A,/,1P,6E12.4)')
c    1  ' e1L(1),e1L(2),u12L(1),u12L(2),gL(1),gL(2)=',
c    1    e1L(1),e1L(2),u12L(1),u12L(2),gL(1),gL(2)
c     WRITE(6,'(/,A,/,1P,6E12.4)')
c    1  ' tL(1),tL(2),rhoL(1),rhoL(2),e2L(1),e2L(2)=',
c    1    tL(1),tL(2),rhoL(1),rhoL(2),e2L(1),e2L(2)
c
      return
c
c     write (not,900)
c     stop
c 900 format (//' SUBROUTINE WALL HAS NOT BEEN PROVIDED.')
      end
```

===