ADAS Subroutine bfdata

```
C
      SUBROUTINE BFDATA( IUNIT  , NDLEV  , NDTRN , NDMET ,
     &                   TITLED , IZ     , IZ0   , IZ1   , BWNO  ,
     &                   NPL    , BWNOA  , LBSETA, PRTWTA, CPRTA ,
     &                   IL     ,
     &                   IA     , CSTRGA , ISA   , ILA   , XJA   , WA ,
     &                   CPLA   , NPLA   , IPLA  , ZPLA  ,
     &                   CIONP  ,
     &                   NV     , SCEF   ,
     &                   ITRAN  , MAXLEV ,
     &                   TCODE  , I1A    , I2A   , AVAL  , SCOM
     &                 )
C
C-----------------------------------------------------------------------
C
C  ****************** FORTRAN77 SUBROUTINE: BFDATA *********************
C
C  PURPOSE:  TO FETCH DATA FROM INPUT ADF04 DATA SET, INCLUDING
C            MULTIPLE PARENTS ON FREE-ELECTRON AND CHARGE EXCHANGE
C            ON RECOMBINATION, INCLUSION OF EXPLICIT CONTRIBUTIONS BY
C            IONISATION.
C
C            IMPROVEMENT OF AUTOMATIC IONISATION CALC. BY INCLUDING
C            ASSIGNMENT OF FINAL STATE PARENT. KEEPS IONISATION
C            POTENTIAL INFORMATION FROM LEVEL TERMINATOR LINE.
C
C  CALLING PROGRAM: ADAS215
C
C  DATA:
C           THE 'REAL' DATA IN THE FILE IS REPRESENTED IN AN ABBREVIATED
C           FORM WHICH OMITS THE "D" OR "E" EXPONENT SPECIFIER.
C           e.g. 1.23D-06 or 1.23E-06 IS REPRESENTED AS 1.23-06
C                6.75D+07 or 6.75E+07 IS REPRESENTED AS 6.75+07
C
C           THEREFORE THE FORM OF EACH 'REAL' NUMBER IN THE DATA SET IS:
C                          N.NN+NN or N.NN-NN
C
C           THE UNITS USED IN THE DATA FILE ARE TAKEN AS FOLLOWS:
C
C           IONISATION POTENTIAL: WAVE NUMBER (CM-1)
C           INDEX LEVEL ENERGIES: WAVE NUMBER (CM-1)
C           TEMPERATURES        : KELVIN
C           A-VALUES            : SEC-1
C           GAMMA-VALUES        :
C           RATE COEFFT.        : CM3 SEC-1
C
C
C  SUBROUTINE:
C
C  INPUT : (I*4)  IUNIT   = UNIT TO WHICH INPUT FILE IS ALLOCATED
C  INPUT : (I*4)  NDLEV   = MAXIMUM NUMBER OF LEVELS THAT CAN BE READ
C  INPUT : (I*4)  NDTRN   = MAX. NUMBER OF TRANSITIONS THAT CAN BE READ
```

```
C  INPUT : (I*4)  NDMET   = MAX. NUMBER OF METASTABLES ALLOWED
C
C  OUTPUT: (C*3)  TITLED  = ELEMENT SYMBOL.
C  OUTPUT: (I*4)  IZ      =  RECOMBINED ION CHARGE READ
C  OUTPUT: (I*4)  IZ0     =         NUCLEAR CHARGE READ
C  OUTPUT: (I*4)  IZ1     = RECOMBINING ION CHARGE READ
C                           (NOTE: IZ1 SHOULD EQUAL IZ+1)
C  OUTPUT: (R*8)  BWNO    = IONISATION POTENTIAL (CM-1) OF LOWEST PARENT
C  OUTPUT: (I*4)  NPL     = NUMBER OF PARENTS ON FIRST LINE AND USED
C                           IN LEVEL ASSIGNMENTS
C  OUTPUT: (R*8)  BWNOA() = IONISATION POTENTIAL (CM-1) OF PARENTS
C  OUTPUT: (L*4)  LBSETA()= .TRUE.  - PARENT WEIGHT SET FOR BWNOA()
C                           .FALSE. - PARENT WEIGHT NOT SET FOR BWNOA()
C  OUTPUT: (R*8)  PRTWTA()= PARENT WEIGHT FOR BWNOA()
C  OUTPUT: (C*9)  CPRTA() = PARENT NAME IN BRACKETS
C
C  OUTPUT: (I*4)  IL      = INPUT DATA FILE: NUMBER OF ENERGY LEVELS
C
C  OUTPUT: (I*4)  IA()    = ENERGY LEVEL INDEX NUMBER
C  OUTPUT: (C*18) CSTRGA()= NOMENCLATURE/CONFIGURATION FOR LEVEL 'IA()'
C  OUTPUT: (I*4)  ISA()   = MULTIPLICITY FOR LEVEL 'IA()'
C                           NOTE: (ISA-1)/2 = QUANTUM NUMBER (S)
C  OUTPUT: (I*4)  ILA()   = QUANTUM NUMBER (L) FOR LEVEL 'IA()'
C  OUTPUT: (R*8)  XJA()   = QUANTUM NUMBER (J-VALUE) FOR LEVEL 'IA()'
C                           NOTE: (2*XJA)+1 = STATISTICAL WEIGHT
C  OUTPUT: (R*8)  WA()    = ENERGY RELATIVE TO LEVEL 1 (CM-1) FOR LEVEL
C                           'IA()'
C  OUTPUT: (C*1)  CPLA()  = CHAR. SPECIFYING 1ST PARENT FOR LEVEL 'IA()'
C                                INTEGER - PARENT IN BWNOA() LIST
C                                'BLANK' - PARENT BWNOA(1)
C                                  'X'   - DO NOT ASSIGN A PARENT
C  OUTPUT: (I*4)  NPLA()  = NO. OF PARENT/ZETA CONTRIBUTIONS TO IONIS.
C                           OF LEVEL
C  OUTPUT: (I*4)  IPLA(,) = PARENT INDEX FOR CONTRIBUTIONS TO IONIS.
C                           OF LEVEL
C                           1ST DIMENSION: PARENT INDEX
C                           2ND DIMENSION: LEVEL INDEX
C  OUTPUT: (I*4)  ZPLA(,) = EFF. ZETA PARAM. FOR CONTRIBUTIONS TO IONIS.
C                           OF LEVEL
C                           1ST DIMENSION: PARENT INDEX
C                           2ND DIMENSION: LEVEL INDEX
C
C  OUTPUT: (C*92) CIONP   = STRING CONTAINING LEVEL TERMINATOR LINE
C
C  OUTPUT: (I*4)  NV      = INPUT DATA FILE: NUMBER OF GAMMA/TEMPERATURE
C                           PAIRS FOR A GIVEN TRANSITION.
C  OUTPUT: (R*8)  SCEF()  = INPUT DATA FILE: ELECTRON TEMPERATURES (K)
C                           (INITIALLY JUST THE MANTISSA. SEE 'ITPOW()')
C                           (NOTE: TE=TP=TH IS ASSUMED)
C
C  OUTPUT: (I*4)  ITRAN   = INPUT DATA FILE: NUMBER OF TRANSITIONS
C  OUTPUT: (I*4)  MAXLEV  = HIGHEST INDEX LEVEL IN READ TRANSITIONS
C
```

```
C  OUTPUT: (C*1)  TCODE() = TRANSITION: DATA TYPE POINTER:
C                           ' ' => Electron Impact   Transition
C                           'P' => Proton   Impact   Transition
C                           'H' => Charge   Exchange Recombination
C                           'R' => Free     Electron Recombination
C                           'I' => Coll. ionisation from lower stage ion
C  OUTPUT: (I*4)  I1A()   = TRANSITION:
C                            LOWER ENERGY LEVEL INDEX (CASE ' ' & 'P')
C                            SIGNED PARENT NDEX (CASE 'H','R' & 'I')
C  OUTPUT: (I*4)  I2A()   = TRANSITION:
C                            UPPER ENERGY LEVEL INDEX (CASE ' ' & 'P')
C                            CAPTURING LEVEL INDEX (CASE 'H','R' & 'I')
C  OUTPUT: (R*8)  AVAL()  = TRANSITION:
C                            A-VALUE (SEC-1)          (CASE ' ')
C                            NEUTRAL BEAM ENERGY      (CASE 'H')
C                            NOT USED             (CASE 'P','R' & 'I')
C  OUTPUT: (R*8)  SCOM(,) = TRANSITION:
C                            GAMMA VALUES             (CASE ' ' & 'P')
C                            RATE COEFFT.(CM3 SEC-1)(CASE 'H','R' & 'I')
C                           1ST DIMENSION - TEMPERATURE 'SCEF()'
C                           2ND DIMENSION - TRANSITION NUMBER
C
C          (I*4)  NVMAX   = PARAMETER = MAX. NUMBER OF TEMPERATURES
C                                       THAT CAN BE READ IN.
C          (I*4)  MTIED   = PARAMETER = MUST BE GREATER THAN OR EQUAL TO
C                                       THE MAX. NO. OF LEVELS.
C          (R*8)  DZERO   = PARAMETER = MINIMUM VALUE FOR 'AVAL()' AND
C                                       'SCOM()' ARRAYS = 1.0D-30
C
C          (I*4)  I4UNIT  = FUNCTION (SEE ROUTINE SELECTION BELOW)
C          (I*4)  IQS     = X-SECT DATA FORMAT SELECTOR
C                           NOTE: IQS=3 ONLY ALLOWED IN THIS PROGRAM
C          (I*4)  IFAIL   = FAILURE NUMBER FROM B9PARS AND B9PRS1
C          (I*4)  I       = GENERAL USE.
C          (I*4)  IABT    = RETURN CODE FROM 'R(FCTN' (0 => NO ERROR)
C                           OR FROM INTERROGATION OF 'C7'
C          (I*4)  J       = GENERAL USE.
C          (I*4)  J1      = INPUT DATA FILE - SELECTED TRANSITION:
C                            LOWER ENERGY LEVEL INDEX (CASE ' ' & 'P')
C          (I*4)  J2      = INPUT DATA FILE - SELECTED TRANSITION:
C                            UPPER ENERGY LEVEL INDEX (CASE ' ' & 'P')
C                            CAPTURING    LEVEL INDEX (CASE 'H' & 'R')
C          (I*4)  LENCST  = BYTE LENGTH OF STRING CSTRGA()
C          (I*4)  ILINE   = ENERGY LEVEL INDEX FOR CURRENT LINE
C          (I*4)  IRECL   = RECORD LENGTH OF INPUT DATASET (<=128)
C          (I*4)  IAPOW   = EXPONENT OF 'AVALM'
C          (I*4)  IGPOW() = EXPONENT OF 'GAMMA()'
C          (I*4)  ITPOW() = TEMPERATURES - EXPONENT
C                           NOTE: MANTISSA INITIALLY KEPT IN 'SCEF()'
C
C          (R*4)  ZF      = SHOULD BE EQUIVALENT TO 'IZ1'
C
C          (R*8)  AVALM   = INPUT DATA FILE - SELECTED TRANSITION:
```

```
C                            MANTISSA OF:   ('IAPOW' => EXPONENT)
C                             A-VALUE (SEC-1)          (CASE ' ')
C                             NEUTRAL BEAM ENERGY      (CASE 'H')
C                             NOT USED              (CASE 'P','R' & 'I')
C          (R*8)  GAMMA() = INPUT DATA FILE - SELECTED TRANSITION:
C                            MANTISSA OF: ('IGPOW()' => EXPONENT)
C                             GAMMA VALUES             (CASE ' ' & 'P')
C                             RATE COEFFT.(CM3 SEC-1)(CASE 'H','R' & 'I')
C                            DIMENSION => TEMPERATURE 'SCEF()'
C
C          (C*7)  C7      = USED TO PARSE VALUE FOR XJA()
C          (C*7)  CDELIM  = DELIMITERS FOR INPUT OF DATA FROM HEADERS
C          (C*18) C18     = USED TO PARSE VALUE TO CSTRGA()
C          (C*18) C18T    = COPY OF C18 : UNSATISFACTORY METHOD OF
C                            AVOIDING COMPILER REFERENCE ERROR :
C                            DHB 07.04.95
C          (C*80) CLINE   = CURRENT ENERGY LEVEL INDEX PARAMETER LINE
C          (C*75) STRING  = TAIL STRING OF 1ST DATA LINE FOR PARSING
C          (C*44) STRG1   = TAIL STRING OF LEVEL SPEC LINES FOR PARSING
C          (C*128)BUFFER  = GENERAL STRING BUFFER STORAGE
C          (C*3)  CITPOW()= USED TO PARSE VALUES TO ITPOW()
C          (C*5)  CSCEF() = USED TO PARSE VALUES TO SCEF()
C
C          (L*4)  LDATA   = IDENTIFIES  WHETHER  THE END OF AN  INPUT
C                            SECTION IN THE DATA SET HAS BEEN LOCATED.
C                            (.TRUE. => END OF SECTION REACHED)
C          (L*4)  LTCHR   = .TRUE.  => CURRENT 'TCODE()' = 'H' OR 'R'
C                                                           OR 'I'
C                         = .FALSE. => CURRENT 'TCODE()'.NE.'H' OR 'R'
C                                                            OR 'I'
C          (L*4)  LTCPR   = .TRUE.  => CURRENT 'TCODE()' = 'P' OR 'R'
C                                                           OR 'I'
C                         = .FALSE. => CURRENT 'TCODE()'.NE. 'P' OR 'R'
C                                                             OR 'I'
C          (L*4)  LERROR  = .TRUE.  => UNTIED LEVEL FOUND
C                         = .FALSE. => ALL LEVELS TIED
C          (L*4)  LTIED() = .TRUE.  => SPECIFIED LEVEL TIED
C                         = .FALSE. => SPECIFIED LEVEL IS UNTIED
C                            DIMENSION => LEVEL INDEX
C
C
C NOTE:            LTCHR        LTCPR         TCODE()
C                  ------------------------------------
C                  .TRUE.       .TRUE.    =>     'R','I'
C                  .TRUE.       .FALSE.   =>     'H'
C                  .FALSE.      .TRUE.    =>     'P'
C                  .FALSE.      .FALSE.   =>     ' '
C
C        FOR A-VALUES & GAMMA-VALUES ENTRIES LESS THAN 'DZERO' ARE TAKEN
C        AS BEING EQUAL TO DZERO. THIS AFFECTS THE 'AVAL()' AND 'SCOM()'
C        ARRAYS.
C
C ROUTINES:
```

```
C          ROUTINE    SOURCE    BRIEF DESCRIPTION
C          ------------------------------------------------------------
C          I4UNIT     ADAS      FETCH UNIT NUMBER FOR OUTPUT OF MESSAGES
C          R8FCTN     ADAS      CONVERTS FROM CHARACTER TO REAL VARIABLE
C          I4FCTN     ADAS      CONVERTS FROM CHAR. TO INTEGER  VARIABLE
C          XXWORD     ADAS      PARSES A STRING INTO SEPARATE WORDS
C                               FOR ' ()<>{}' DELIMITERS
C
C AUTHOR:  HP SUMMERS   UNIVERSITY OF STRATHCLYDE
C          JA8.08
C          TEL.  0141-553-4196
C
C DATE:    04/06/98
C
C UPDATE:  1.1 DATE: 09/08/98
C MODIFIED: RICHARD MARTIN
C - PUT UNDER SCCS CONTROL.
C
C UPDATE:  1.2 DATE: 17/05/07
C MODIFIED: Allan Whiteford
C - Updated comments as part of subroutine documentation
C                 procedure.
C
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
      CHARACTER*92        CIONP
      CHARACTER           CPLA(NDLEV)
      CHARACTER*9         CPRTA(NDMET)
      CHARACTER*(*)       CSTRGA(NDLEV)
      CHARACTER           TCODE(NDTRN)
      CHARACTER*3         TITLED
      INTEGER             I1A(NDTRN),  I2A(NDTRN),  IA(NDLEV),   IL
      INTEGER             ILA(NDLEV),  IPLA(NDMET,NDLEV)
      INTEGER             ISA(NDLEV),  ITRAN,       IUNIT,       IZ
      INTEGER             IZ0,         IZ1,         MAXLEV,      NDLEV
      INTEGER             NDMET,       NDTRN,       NPL
      INTEGER             NPLA(NDLEV), NV
      LOGICAL             LBSETA(NDMET)
      REAL*8              AVAL(NDTRN), BWNO,        BWNOA(NDMET)
      REAL*8              PRTWTA(NDMET),            SCEF(NVMAX)
      REAL*8              SCOM(NVMAX,NDTRN),        WA(NDLEV)
      REAL*8              XJA(NDLEV),  ZPLA(NDMET,NDLEV)
```