
Subject: Converting characters to real

Posted by [Kelly Dean](#) on Mon, 09 Nov 1998 08:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

I am working on some IDL to code to read in GMS imagery in the McIDAS format. There is one section that I am having difficult understanding - the orbital element section. I know how to use the numbers in the navigation model, but how do I extract the values from this section?

From what I can understand, the nav section is a character string and the character values are converted into a bit, then converted into a real. There is some FORTRAN code at the bottom that maybe of some help.

I would appreciate any suggestions anyone may have to help me to use IDL to convert this character string into a real number.

Kelly

=====FORTRAN Code =====

```
      SUBROUTINE DECODE_OA_BLOCK( COBAT, FORM )
C*****
C  ORBIT AND ATTITUDE DATA PROCESSING ROUTINE
C*****
C
      implicit none
      INTEGER MAP(672,4)
      integer i,j
      CHARACTER COBAT*3200, FORM*5
      REAL  R4DMY,RESLIN(4),RESELM(4),RLIC(4),RELMFC(4),SENSSU(4)
      REAL  VMIS(3),ELMIS(3,3),RLINE(4),RELMNT(4)
      real  SUBLAT,SUBLON
      DOUBLE PRECISION  R8DMY,DSPIN,DTIMS,ATIT(10,10),
&                      ORBT1(35,8)
C
      COMMON /MMAP1/MAP
      COMMON /NAV1/SUBLAT, SUBLON
C
      EQUIVALENCE (MAP( 5,1),DTIMS),  (MAP( 7,1),RESLIN(1))
      EQUIVALENCE (MAP(11,1),RESELM(1)),(MAP(15,1),RLIC(1))
      EQUIVALENCE (MAP(19,1),RELMFC(1)),(MAP(27,1),SENSSU(1))
      EQUIVALENCE (MAP(31,1),RLINE(1)), (MAP(35,1),RELMNT(1))
      EQUIVALENCE (MAP(39,1),VMIS(1)), (MAP(42,1),ELMIS)
      EQUIVALENCE (MAP(131,1),DSPIN)
      EQUIVALENCE (MAP(13,3),ORBT1(1,1)),(MAP(13,2),ATIT(1,1))
C
```

```

C=====
=====

```

```

C
  DO 1000 I=1,4
    DO 1100 J=1,672
      MAP(J,I) = 0
1100  CONTINUE
1000 CONTINUE
C

```

```

  CALL SV0100( 6, 8, COBAT( 1: 6), R4DMY , DTIMS )
  CALL SV0100( 4, 8, COBAT( 7: 10), RESLIN(1), R8DMY )
  CALL SV0100( 4, 8, COBAT( 11: 14), RESLIN(2), R8DMY )
  CALL SV0100( 4, 8, COBAT( 11: 14), RESLIN(3), R8DMY )
  CALL SV0100( 4, 8, COBAT( 11: 14), RESLIN(4), R8DMY )
  CALL SV0100( 4,10, COBAT( 15: 18), RESELM(1), R8DMY )
  CALL SV0100( 4,10, COBAT( 19: 22), RESELM(2), R8DMY )
  CALL SV0100( 4,10, COBAT( 19: 22), RESELM(3), R8DMY )
  CALL SV0100( 4,10, COBAT( 19: 22), RESELM(4), R8DMY )
  CALL SV0100( 4, 4, COBAT( 23: 26), RLIC(1) , R8DMY )
  CALL SV0100( 4, 4, COBAT( 27: 30), RLIC(2) , R8DMY )

```

....

....

....

....

.... if the rest of the code is needed, let me know.

```

SUBROUTINE SV0100( IWORD, IPOS, C, R4DAT, R8DAT)

```

```

C*****

```

```

C  TYPE CONVERT ROUTINE ( R-TYPE )

```

```

C*****

```

```

C
  implicit none
  INTEGER IWORD,IPOS,IDATA1
  CHARACTER C(*)*1
  REAL R4DAT
  DOUBLE PRECISION R8DAT

```

```

C
C=====

```

```

C
  R4DAT = 0.0
  R8DAT = 0.D0
  IF(IWORD.EQ.4) THEN
    IDATA1 = ICHAR( C(1)(1:1) )/128
    R8DAT = DFLOAT( MOD(ICHAR(C(1)(1:1)),128))*2.D0**(8*3)+
*      DFLOAT( ICHAR(C(2)(1:1)) )*2.D0**(8*2)+
*      DFLOAT( ICHAR(C(3)(1:1)) )*2.D0**(8*1)+

```

```

*      DFLOAT( ICHAR(C(4)(1:1)) )
R8DAT = R8DAT/10.D0**IPOS
IF(IDATA1.EQ.1) R8DAT = -R8DAT
R4DAT = SNGL(R8DAT)
ELSEIF(IWORD.EQ.6) THEN
  IDATA1 = ICHAR( C(1)(1:1) )/128
  R8DAT = DFLOAT( MOD(ICHAR(C(1)(1:1)),128))*2.D0**(8*5)+
*      DFLOAT( ICHAR(C(2)(1:1)) )*2.D0**(8*4)+
*      DFLOAT( ICHAR(C(3)(1:1)) )*2.D0**(8*3)+
*      DFLOAT( ICHAR(C(4)(1:1)) )*2.D0**(8*2)+
*      DFLOAT( ICHAR(C(5)(1:1)) )*2.D0**(8*1)+
*      DFLOAT( ICHAR(C(6)(1:1)) )
  R8DAT = R8DAT/10.D0**IPOS
  IF(IDATA1.EQ.1) R8DAT =-R8DAT
  R4DAT = SNGL(R8DAT)
ENDIF
RETURN
END

```
