```
      INTEGER FUNCTION GET_GRID_INDEX( LAT, LON, GRID_INDEX )

c-----------------------------------------------------------------------
c!F77
c
c!DESCRIPTION:
c    For a given latitude and longitude, compute the grid index value
c    in a global equal area grid.
c    The grid is defined as follows:
c    (1) Assume equatorial and polar circumference of Earth is 40000 km.
c    (2) Divide the latitude range from -90 to 90 into
c        800 bands spaced at 25 km (20000 km / 25 km = 800).
c    (3) Divide the longitude range at the equator from 0 to 360 into
c        1600 bands spaced at 25 km (40000 km / 25 km = 1600).
c    (4) For each latitude band north or south of the equator,
c        divide the longitude range from 0 to 360 into
c        [ 1600 * cosine( latitude ) ] bands.
c
c!INPUT PARAMETERS:
c    LAT               Latitude (degrees, -90S to +90N)
c    LON               Longitude (degrees, -180W to 180E, Greenwich=0)
c
c!OUTPUT PARAMETERS:
c    GET_GRID_INDEX    Return status
c                       0 => Success
c                      -1 => Latitude not in range -90 to 90
c                      -2 => Longitude not in range -180 to 180
c                      -3 => Grid index not in range 1 to 814880
c    GRID_INDEX        Index value in global grid (1-814880)
c
c!REVISION HISTORY:
c
c!TEAM-UNIQUE HEADER:
c    Developed by the MODIS Group, CIMSS/SSEC, UW-Madison.
c
c!DESIGN NOTES:
c    Original version by Richard.Frey@ssec.wisc.edu
c
c!END
c-----------------------------------------------------------------------

      implicit none

c ... Input arguments

      real lat, lon

c ... Output arguments

      integer grid_index

c ... Parameters

      real grdsiz
```

```
      parameter ( grdsiz = 25.0 )
      real ce
      parameter ( ce = 40000.0 )
      real beg_lat
      parameter ( beg_lat = 90.0 )
      real beg_lon
      parameter ( beg_lon = 0.0 )

c ... Constants

      integer ne, num_bands, beg_ltindx
      real lat_inc

c ... Local variables

      integer beg_lnindx, lat_band, lon_bin, npboxes
      real alon

      integer num_boxes( 800 )
      integer box_index

      real lon_inc( 800 )
      integer lon_index

c ... Data statements

      data ( num_boxes( box_index ), box_index = 1, 114 ) /
     &        3,       12,       28,       50,       78,      113,
     &      154,      201,      254,      314,      380,      452,
     &      531,      616,      707,      804,      908,     1018,
     &     1134,     1256,     1385,     1520,     1661,     1808,
     &     1962,     2122,     2288,     2460,     2639,     2824,
     &     3015,     3212,     3416,     3626,     3842,     4064,
     &     4293,     4528,     4769,     5016,     5269,     5529,
     &     5795,     6067,     6345,     6629,     6920,     7217,
     &     7520,     7829,     8144,     8465,     8793,     9127,
     &     9467,     9813,    10165,    10523,    10887,    11257,
     &    11634,    12017,    12406,    12801,    13202,    13609,
     &    14022,    14441,    14866,    15297,    15734,    16177,
     &    16626,    17081,    17542,    18009,    18482,    18961,
     &    19446,    19937,    20434,    20937,    21446,    21961,
     &    22482,    23009,    23542,    24081,    24626,    25177,
     &    25734,    26297,    26865,    27439,    28019,    28605,
     &    29197,    29795,    30399,    31008,    31623,    32244,
     &    32871,    33504,    34142,    34786,    35436,    36092,
     &    36753,    37420,    38093,    38771,    39455,    40145 /

      data ( num_boxes( box_index ), box_index = 115, 228 ) /
     &    40840,    41541,    42248,    42960,    43678,    44402,
     &    45131,    45866,    46606,    47352,    48103,    48860,
     &    49623,    50391,    51165,    51944,    52729,    53519,
     &    54314,    55115,    55921,    56733,    57550,    58373,
     &    59201,    60034,    60873,    61717,    62566,    63421,
     &    64281,    65146,    66017,    66893,    67774,    68660,
```

```
&    69552,    70449,    71351,    72258,    73170,    74087,
&    75010,    75938,    76871,    77809,    78752,    79700,
&    80653,    81611,    82574,    83542,    84515,    85493,
&    86476,    87464,    88457,    89455,    90458,    91466,
&    92479,    93496,    94518,    95545,    96577,    97614,
&    98656,    99702,   100753,   101809,   102869,   103934,
&   105004,   106078,   107157,   108241,   109329,   110422,
&   111520,   112622,   113729,   114840,   115956,   117076,
&   118201,   119330,   120464,   121602,   122744,   123891,
&   125042,   126198,   127358,   128522,   129690,   130863,
&   132040,   133221,   134407,   135597,   136791,   137989,
&   139191,   140397,   141608,   142823,   144042,   145265,
&   146492,   147723,   148958,   150197,   151440,   152687 /

 data ( num_boxes( box_index ), box_index = 229, 342 ) /
&   153938,   155193,   156451,   157713,   158979,   160249,
&   161523,   162801,   164082,   165367,   166656,   167949,
&   169245,   170545,   171849,   173156,   174467,   175781,
&   177099,   178421,   179746,   181075,   182407,   183743,
&   185082,   186424,   187770,   189119,   190472,   191828,
&   193187,   194550,   195916,   197285,   198657,   200033,
&   201412,   202794,   204179,   205567,   206958,   208352,
&   209750,   211151,   212555,   213962,   215372,   216785,
&   218200,   219618,   221039,   222463,   223890,   225320,
&   226753,   228188,   229626,   231067,   232511,   233957,
&   235406,   236858,   238312,   239769,   241229,   242691,
&   244156,   245623,   247093,   248565,   250040,   251517,
&   252996,   254478,   255962,   257448,   258937,   260428,
&   261921,   263417,   264915,   266415,   267917,   269421,
&   270927,   272436,   273947,   275460,   276975,   278492,
&   280011,   281532,   283055,   284580,   286106,   287634,
&   289164,   290696,   292230,   293766,   295303,   296842,
&   298383,   299925,   301469,   303015,   304562,   306111,
&   307662,   309214,   310768,   312323,   313880,   315438 /

 data ( num_boxes( box_index ), box_index = 343, 456 ) /
&   316997,   318558,   320120,   321683,   323248,   324814,
&   326381,   327950,   329520,   331091,   332663,   334236,
&   335811,   337387,   338964,   340542,   342121,   343701,
&   345282,   346864,   348447,   350031,   351615,   353200,
&   354786,   356373,   357961,   359550,   361139,   362729,
&   364320,   365911,   367503,   369096,   370689,   372283,
&   373877,   375472,   377067,   378663,   380259,   381856,
&   383453,   385050,   386648,   388246,   389844,   391443,
&   393042,   394641,   396240,   397839,   399439,   401039,
&   402639,   404239,   405839,   407439,   409039,   410639,
&   412239,   413839,   415439,   417039,   418638,   420237,
&   421836,   423435,   425034,   426632,   428230,   429828,
&   431425,   433022,   434619,   436215,   437811,   439406,
&   441001,   442595,   444189,   445782,   447375,   448967,
&   450558,   452149,   453739,   455328,   456917,   458505,
&   460092,   461678,   463263,   464847,   466431,   468014,
&   469596,   471177,   472757,   474336,   475914,   477491,
&   479067,   480642,   482215,   483787,   485358,   486928,
```

```
&   488497,   490064,   491630,   493195,   494758,   496320 /

 data ( num_boxes( box_index ), box_index = 457, 570 ) /
&   497881,   499440,   500998,   502555,   504110,   505664,
&   507216,   508767,   510316,   511863,   513409,   514953,
&   516495,   518036,   519575,   521112,   522648,   524182,
&   525714,   527244,   528772,   530298,   531823,   533346,
&   534867,   536386,   537903,   539418,   540931,   542442,
&   543951,   545457,   546961,   548463,   549963,   551461,
&   552957,   554450,   555941,   557430,   558916,   560400,
&   561882,   563361,   564838,   566313,   567785,   569255,
&   570722,   572187,   573649,   575109,   576566,   578020,
&   579472,   580921,   582367,   583811,   585252,   586690,
&   588125,   589558,   590988,   592415,   593839,   595260,
&   596678,   598093,   599506,   600916,   602323,   603727,
&   605128,   606526,   607920,   609311,   610699,   612084,
&   613466,   614845,   616221,   617593,   618962,   620328,
&   621691,   623050,   624406,   625759,   627108,   628454,
&   629796,   631135,   632471,   633803,   635132,   636457,
&   637779,   639097,   640411,   641722,   643029,   644333,
&   645633,   646929,   648222,   649511,   650796,   652077,
&   653355,   654629,   655899,   657165,   658427,   659685 /

 data ( num_boxes( box_index ), box_index = 571, 684 ) /
&   660940,   662191,   663438,   664681,   665920,   667155,
&   668386,   669613,   670836,   672055,   673270,   674481,
&   675687,   676889,   678087,   679281,   680471,   681657,
&   682838,   684015,   685188,   686357,   687521,   688681,
&   689837,   690988,   692135,   693277,   694415,   695549,
&   696678,   697803,   698923,   700039,   701150,   702257,
&   703359,   704457,   705550,   706638,   707722,   708801,
&   709875,   710945,   712010,   713070,   714126,   715177,
&   716223,   717265,   718302,   719334,   720361,   721383,
&   722400,   723413,   724421,   725424,   726422,   727415,
&   728403,   729386,   730364,   731337,   732305,   733268,
&   734226,   735179,   736127,   737070,   738008,   738941,
&   739869,   740792,   741709,   742621,   743528,   744430,
&   745327,   746219,   747105,   747986,   748862,   749733,
&   750598,   751458,   752313,   753162,   754006,   754845,
&   755678,   756506,   757329,   758146,   758958,   759764,
&   760565,   761360,   762150,   762935,   763714,   764488,
&   765256,   766019,   766776,   767527,   768273,   769013,
&   769748,   770477,   771201,   771919,   772631,   773338 /

 data ( num_boxes( box_index ), box_index = 685, 798 ) /
&   774039,   774734,   775424,   776108,   776786,   777459,
&   778126,   778787,   779443,   780093,   780737,   781375,
&   782008,   782635,   783256,   783871,   784480,   785084,
&   785682,   786274,   786860,   787440,   788014,   788583,
&   789146,   789703,   790254,   790799,   791338,   791871,
&   792398,   792919,   793434,   793943,   794446,   794943,
&   795434,   795919,   796398,   796871,   797338,   797799,
&   798254,   798703,   799146,   799583,   800014,   800439,
&   800858,   801271,   801678,   802079,   802474,   802863,
```

```
&   803246,   803623,   803993,   804357,   804715,   805067,
&   805413,   805753,   806087,   806415,   806736,   807051,
&   807360,   807663,   807960,   808251,   808535,   808813,
&   809085,   809351,   809611,   809864,   810111,   810352,
&   810587,   810816,   811038,   811254,   811464,   811668,
&   811865,   812056,   812241,   812420,   812592,   812758,
&   812918,   813072,   813219,   813360,   813495,   813624,
&   813746,   813862,   813972,   814076,   814173,   814264,
&   814349,   814428,   814500,   814566,   814626,   814679,
&   814726,   814767,   814802,   814830,   814852,   814868 /

 data ( num_boxes( box_index ), box_index = 799, 800 ) /
&   814877,   814880/

 data ( lon_inc( lon_index ), lon_index = 1, 114 ) /
&120.00000, 40.00000, 22.50000, 16.36364, 12.85714, 10.28571,
&  8.78049,  7.65957,  6.79245,  6.00000,  5.45455,  5.00000,
&  4.55696,  4.23529,  3.95604,  3.71134,  3.46154,  3.27273,
&  3.10345,  2.95082,  2.79070,  2.66667,  2.55319,  2.44898,
&  2.33766,  2.25000,  2.16867,  2.09302,  2.01117,  1.94595,
&  1.88482,  1.82741,  1.76471,  1.71429,  1.66667,  1.62162,
&  1.57205,  1.53191,  1.49378,  1.45749,  1.42292,  1.38462,
&  1.35338,  1.32353,  1.29496,  1.26761,  1.23711,  1.21212,
&  1.18812,  1.16505,  1.14286,  1.12150,  1.09756,  1.07784,
&  1.05882,  1.04046,  1.02273,  1.00559,   .98901,   .97297,
&   .95491,   .93995,   .92545,   .91139,   .89776,   .88452,
&   .87167,   .85919,   .84706,   .83527,   .82380,   .81264,
&   .80178,   .79121,   .78091,   .77088,   .76110,   .75157,
&   .74227,   .73320,   .72435,   .71571,   .70727,   .69903,
&   .69098,   .68311,   .67542,   .66790,   .66055,   .65336,
&   .64632,   .63943,   .63380,   .62718,   .62069,   .61433,
&   .60811,   .60201,   .59603,   .59113,   .58537,   .57971,
&   .57416,   .56872,   .56426,   .55901,   .55385,   .54878,
&   .54463,   .53973,   .53492,   .53097,   .52632,   .52174 /

 data ( lon_inc( lon_index ), lon_index = 115, 228 ) /
&   .51799,   .51355,   .50919,   .50562,   .50139,   .49724,
&   .49383,   .48980,   .48649,   .48257,   .47936,   .47556,
&   .47182,   .46875,   .46512,   .46213,   .45860,   .45570,
&   .45283,   .44944,   .44665,   .44335,   .44064,   .43742,
&   .43478,   .43217,   .42908,   .42654,   .42403,   .42105,
&   .41860,   .41618,   .41332,   .41096,   .40863,   .40632,
&   .40359,   .40134,   .39911,   .39691,   .39474,   .39258,
&   .39003,   .38793,   .38585,   .38380,   .38176,   .37975,
&   .37775,   .37578,   .37383,   .37190,   .36999,   .36810,
&   .36623,   .36437,   .36254,   .36072,   .35892,   .35714,
&   .35538,   .35398,   .35225,   .35054,   .34884,   .34716,
&   .34549,   .34417,   .34253,   .34091,   .33962,   .33803,
&   .33645,   .33520,   .33364,   .33210,   .33088,   .32937,
&   .32787,   .32668,   .32520,   .32403,   .32258,   .32143,
&   .32000,   .31887,   .31746,   .31634,   .31524,   .31386,
&   .31277,   .31142,   .31034,   .30928,   .30822,   .30691,
&   .30586,   .30483,   .30354,   .30252,   .30151,   .30050,
&   .29950,   .29851,   .29727,   .29630,   .29532,   .29436,
```

```
&   .29340,   .29245,   .29150,   .29056,   .28962,   .28869 /

 data ( lon_inc( lon_index ), lon_index = 229, 342 ) /
&   .28777,   .28685,   .28617,   .28526,   .28436,   .28346,
&   .28257,   .28169,   .28103,   .28016,   .27929,   .27842,
&   .27778,   .27692,   .27607,   .27544,   .27460,   .27397,
&   .27314,   .27231,   .27170,   .27088,   .27027,   .26946,
&   .26886,   .26826,   .26746,   .26686,   .26608,   .26549,
&   .26490,   .26412,   .26354,   .26297,   .26239,   .26163,
&   .26106,   .26049,   .25993,   .25937,   .25881,   .25825,
&   .25751,   .25696,   .25641,   .25586,   .25532,   .25478,
&   .25442,   .25388,   .25334,   .25281,   .25228,   .25175,
&   .25122,   .25087,   .25035,   .24983,   .24931,   .24896,
&   .24845,   .24793,   .24759,   .24708,   .24658,   .24624,
&   .24573,   .24540,   .24490,   .24457,   .24407,   .24374,
&   .24341,   .24291,   .24259,   .24226,   .24177,   .24145,
&   .24113,   .24064,   .24032,   .24000,   .23968,   .23936,
&   .23904,   .23857,   .23825,   .23794,   .23762,   .23731,
&   .23700,   .23669,   .23638,   .23607,   .23591,   .23560,
&   .23529,   .23499,   .23468,   .23438,   .23422,   .23392,
&   .23361,   .23346,   .23316,   .23286,   .23271,   .23241,
&   .23211,   .23196,   .23166,   .23151,   .23121,   .23107 /

 data ( lon_inc( lon_index ), lon_index = 343, 456 ) /
&   .23092,   .23062,   .23047,   .23033,   .23003,   .22989,
&   .22974,   .22945,   .22930,   .22915,   .22901,   .22886,
&   .22857,   .22843,   .22828,   .22814,   .22799,   .22785,
&   .22770,   .22756,   .22742,   .22727,   .22727,   .22713,
&   .22699,   .22684,   .22670,   .22656,   .22656,   .22642,
&   .22627,   .22627,   .22613,   .22599,   .22599,   .22585,
&   .22585,   .22571,   .22571,   .22556,   .22556,   .22542,
&   .22542,   .22542,   .22528,   .22528,   .22528,   .22514,
&   .22514,   .22514,   .22514,   .22514,   .22500,   .22500,
&   .22500,   .22500,   .22500,   .22500,   .22500,   .22500,
&   .22500,   .22500,   .22500,   .22500,   .22514,   .22514,
&   .22514,   .22514,   .22514,   .22528,   .22528,   .22528,
&   .22542,   .22542,   .22542,   .22556,   .22556,   .22571,
&   .22571,   .22585,   .22585,   .22599,   .22599,   .22613,
&   .22627,   .22627,   .22642,   .22656,   .22656,   .22670,
&   .22684,   .22699,   .22713,   .22727,   .22727,   .22742,
&   .22756,   .22770,   .22785,   .22799,   .22814,   .22828,
&   .22843,   .22857,   .22886,   .22901,   .22915,   .22930,
&   .22945,   .22974,   .22989,   .23003,   .23033,   .23047 /

 data ( lon_inc( lon_index ), lon_index = 457, 570 ) /
&   .23062,   .23092,   .23107,   .23121,   .23151,   .23166,
&   .23196,   .23211,   .23241,   .23271,   .23286,   .23316,
&   .23346,   .23361,   .23392,   .23422,   .23438,   .23468,
&   .23499,   .23529,   .23560,   .23591,   .23607,   .23638,
&   .23669,   .23700,   .23731,   .23762,   .23794,   .23825,
&   .23857,   .23904,   .23936,   .23968,   .24000,   .24032,
&   .24064,   .24113,   .24145,   .24177,   .24226,   .24259,
&   .24291,   .24341,   .24374,   .24407,   .24457,   .24490,
&   .24540,   .24573,   .24624,   .24658,   .24708,   .24759,
```

```
&   .24793,   .24845,   .24896,   .24931,   .24983,   .25035,
&   .25087,   .25122,   .25175,   .25228,   .25281,   .25334,
&   .25388,   .25442,   .25478,   .25532,   .25586,   .25641,
&   .25696,   .25751,   .25825,   .25881,   .25937,   .25993,
&   .26049,   .26106,   .26163,   .26239,   .26297,   .26354,
&   .26412,   .26490,   .26549,   .26608,   .26686,   .26746,
&   .26826,   .26886,   .26946,   .27027,   .27088,   .27170,
&   .27231,   .27314,   .27397,   .27460,   .27544,   .27607,
&   .27692,   .27778,   .27842,   .27929,   .28016,   .28103,
&   .28169,   .28257,   .28346,   .28436,   .28526,   .28617 /

 data ( lon_inc( lon_index ), lon_index = 571, 684 ) /
&   .28685,   .28777,   .28869,   .28962,   .29056,   .29150,
&   .29245,   .29340,   .29436,   .29532,   .29630,   .29727,
&   .29851,   .29950,   .30050,   .30151,   .30252,   .30354,
&   .30483,   .30586,   .30691,   .30796,   .30928,   .31034,
&   .31142,   .31277,   .31386,   .31524,   .31634,   .31746,
&   .31887,   .32000,   .32143,   .32258,   .32403,   .32520,
&   .32668,   .32787,   .32937,   .33088,   .33210,   .33364,
&   .33520,   .33645,   .33803,   .33962,   .34091,   .34253,
&   .34417,   .34549,   .34716,   .34884,   .35054,   .35225,
&   .35398,   .35538,   .35714,   .35892,   .36072,   .36254,
&   .36437,   .36623,   .36810,   .36999,   .37190,   .37383,
&   .37578,   .37775,   .37975,   .38176,   .38380,   .38585,
&   .38793,   .39003,   .39258,   .39474,   .39691,   .39911,
&   .40134,   .40359,   .40632,   .40863,   .41096,   .41332,
&   .41618,   .41860,   .42105,   .42403,   .42654,   .42908,
&   .43217,   .43478,   .43742,   .44064,   .44335,   .44665,
&   .44944,   .45283,   .45570,   .45860,   .46213,   .46512,
&   .46875,   .47182,   .47556,   .47936,   .48257,   .48649,
&   .48980,   .49383,   .49724,   .50139,   .50562,   .50919 /

 data ( lon_inc( lon_index ), lon_index = 685, 798 ) /
&   .51355,   .51799,   .52174,   .52632,   .53097,   .53492,
&   .53973,   .54463,   .54878,   .55385,   .55901,   .56426,
&   .56872,   .57416,   .57971,   .58537,   .59113,   .59603,
&   .60201,   .60811,   .61433,   .62069,   .62718,   .63269,
&   .63943,   .64632,   .65336,   .66055,   .66790,   .67542,
&   .68311,   .69098,   .69903,   .70727,   .71571,   .72435,
&   .73320,   .74227,   .75157,   .76110,   .77088,   .78091,
&   .79121,   .80178,   .81264,   .82380,   .83527,   .84706,
&   .85919,   .87167,   .88452,   .89776,   .91139,   .92545,
&   .93995,   .95491,   .97297,   .98901,  1.00559,  1.02273,
&  1.04046,  1.05882,  1.07784,  1.09756,  1.12150,  1.14286,
&  1.16505,  1.18812,  1.21212,  1.23711,  1.26761,  1.29496,
&  1.32353,  1.35338,  1.38462,  1.42292,  1.45749,  1.49378,
&  1.53191,  1.57205,  1.62162,  1.66667,  1.71429,  1.76471,
&  1.82741,  1.88482,  1.94595,  2.01117,  2.09302,  2.16867,
&  2.25000,  2.33766,  2.44898,  2.55319,  2.66667,  2.79070,
&  2.95082,  3.10345,  3.27273,  3.46154,  3.71134,  3.95604,
&  4.23529,  4.55696,  5.00000,  5.45455,  6.00000,  6.79245,
&  7.65957,  8.78049, 10.28571, 12.85714, 16.36364, 22.50000 /

 data ( lon_inc( lon_index ), lon_index = 799, 800 ) /
```

```
     & 40.00000,120.00000/

c ... Set constants

      ne = nint((ce/grdsiz)*0.5)*2
      num_bands = ne/2
      lat_inc = 180.0/num_bands
      beg_ltindx = int((90.0-beg_lat)/lat_inc)

c ... Check input arguments

      if ( lat .lt. -90.0 .or. lat .gt. 90.0 ) then
        get_grid_index = -1
        return
      endif

      if ( lon .lt. -180.0 .or. lon .gt. 180.0 ) then
        get_grid_index = -2
        return
      endif

c ... Find latitude band

      lat_band =
     &  min( int( (90.0 - lat ) / lat_inc ) + 1 - beg_ltindx , 800 )

c ... Convert longitude (west negative, -180 to 180) to
c ... McIDAS longitude (west positive, -180 to 180)

      alon = -lon

c ... Convert McIDAS longitude (west positive, -180 to 180) to
c ... Greenwich system (Greenwich zero, 0 to 360)

      if ( alon .le. 0.0 ) then
        alon = abs( alon )
      else
        alon = 360.0 - alon
      endif

c     Get longitude bin of the current latitude band

      beg_lnindx = int( beg_lon / lon_inc( lat_band ) )
      lon_bin = int( alon / lon_inc( lat_band ) ) + 1 - beg_lnindx

      if ( lat_band .eq. 1 ) then
        npboxes = 0
      else
        npboxes = num_boxes( lat_band - 1 )
      endif

c ... Compute grid index

      grid_index = npboxes + lon_bin
```

```
c ... Set function return value

      get_grid_index = 0
      if ( grid_index .lt. 0 .or. grid_index .gt. 814880 ) then
        grid_index = -1
        get_grid_index = -3
      endif

      END
```