KRC planetary surface temperatures: Helplist

Hugh H. Kieffer   File=-/krc/helplist.tex Version 2.2.1 2013aug04

August 5, 2013

Contents

1 Introduction

This document is intended to help the expert user of KRC set up an input file that addresses their goals and will generate the kind of output they desire. It is assumed that the user is familiar with the KRC journal article: H.H. Kieffer, Thermal model for analysis of Mars infrared mapping, J. Geophys. Res. Planets, (2012) [Ref. 1]

The evolution of KRC code is contained in: evolve.txt

A crude diagram of the call architecture is in: flow.txt

2 METHOD

The program is designed to compute surface and subsurface temperatures for a global set of latitudes at a full set of seasons, with enough depth to capture the annual thermal wave, and to compute seasonal condensation mass. For historic reasons, the code has substantial optimization. Although developed for Mars, there are generalities that allow this code set to be used for any solid body with any spin vector, in any orbit (around any star); this is also the source of some of the complexity.

Method is explicit forward finite differences with exponentially increasing layer thickness and binary time increase with depths where allowed by stability. Depth parameter is scaled to the diurnal thermal skin depth. Initially starts at 18 hours with the mean temperature of a perfect conductor. Second degree perturbation is applied at the end (midnight) of the (third) day; this jumps the mean temperature of all layers and the lower boundary to equal the mean surface temperature.

Boundary condition treatment:
Perturbation solution of quartic equation at surface for each iteration; temperature gradient assumed uniform in top interval.
Lower boundary may be insulating or constant-temperature.

Atmospheric Radiation: KRC uses a one-layer atmosphere that is grey in both the solar and infra-red regions. parametric atmosphere. The default atmospheric parameters are based on estimates of Mars' gas and aerosol properties.
Delta-Eddington model for insolation; direct onto sloped surface and diffuse, with possible twilight extension. Atmosphere temperature based on Delta-Eddington solar absorption and IR opacity

Keplerian orbital motion; seasons are at uniform increments of time. Mean orbital elements are pre-calculated for any epoch (all planets and several comets) by the PORB code set.

Units are SI, except for use of days for orbital motion and rotation period

Options:
Different Physical properties below a set layer (IC).
Regional slope
Three ways to handle seasonal global pressure variation

Atmosphere condensation:
Global integral of CO2 frost-gas budget can control surface pressure.
Allows different surface elevation for each latitude zone.
Zonal frost saturation temperature tracks local surface pressure.
Option for cap albedo to depend upon mean daily insolation.

2.1 Convergence Notes

Convergence prediction routine can't jump more than one time constant (TAU=X**2/2) $\tau = x^2/2$ for the total thickness. Therefore, if X(N1) is small, make DDT smaller than usual. If DELJUL is much smaller than (X(N1))**2/2 $X_{N1}^2/2$, then DDT can be as large as 0.3. Otherwise DDT must be about 0 for the prediction routine to work well (it assumes the 3rd derivative to be 0).

3 INPUT FILE

All parameters for KRC are set by a formatted text file. An example is master.inp , which has default values for a 19 latitude set for a run of three martian years, with the last output to disk. Parameter values are listed below their titles, which are in many cases identical to the code name, and last charater of the title is above the last location in the field. Thus, integer values MUST be aligned. Titles with a leading "[" indicate that the value is not used. The recommended procedure is to copy master.inp and edit only the values you wish to change. The number of lines of Latitudes and Elevations must match the value of N4, e.g., 2 lines for N4=11:20, entries beyond the N4 position may be left blank or contain the end of the line. The 7 lines following Elevations are a geometry matrix for Mars orientation and orbit in 2010, and should not be touched; they can be replaced by running PORBMN carefully.

The title lines are skipped, so that you may put comments there carefully

The first input line is always KOLD,KEEP (I*), which sets file usage; these are described in §7. If KOLD=0, then a full set of input values is read.

If and only if there is a third non-zero integer, then will read next card as 6 debug flags, IDB1 to IBD6, which are normally zero. See §10 below
The next (normally second) line is free text where you can outline the purpose of your run.

Change lines may follow immediately after the geometry matrix (see §4) . The end of definition of a "case" is indicated by a "0/" line. Two successive "0/" lines ends the run.

Items with numbers inset 2 spaces below are computed, not input. The source code for 'krccom.inc' indicates which subroutine sets many of the parameters; as the routine name in lowercase just below the parameter name.

```
 - - - - - - - - - - - - - - - - - - - - -
Type 4 Title (20A4) 80 characters of anything to appear at top of each page.

Type 1 Real parameters  (8F10.2) ================================
  Surface Properties
1   ALB    Surface albedo
2   EMIS   Surface emissivity
3   SKRC   Surface thermal inertia [J m^-2 s^-1/2 K^-1] { cal cm * 4.184e4}
4   COND2  Lower material conductivity (IC>0)
5   DENS2  Lower material density (IC>0)
6   PERIOD Length of solar day in days (of 86400 seconds)
7   SPHT   Surface specific heat [J/(kg K)]  {cal/(g K) * 4184.}
8   DENS   Surface density [kg/m^3] {g/cubic cm. *10}
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Atmospheric Properties
9   CABR    IR opacity of dust-free atmosphere of PTOTAL surface pressure
10  AMW     Molecular weight of the atmosphere
11  ABRPHA  unused       [Phase of ABRAMP, degrees relative to midnight]
12  PTOTAL  Global annual mean surface pressure at 0 elev., Pascal[=.01mb]
             If KPREF=2, global average of atmosphere plus cap system.
13  FANON   Mass-fraction of mean atmosphere that is non-condensing
14  TATM    Atm temp for scale-height calculations
-   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -
15  TDEEP   Fixed bottom temperature. Used if IB>=1.
16  SPHT2   Lower material specific heat (IC>0)
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
  Dust  & Slope Properties
17  TAUD    Mean visible opacity of dust, solar wavelengths
18  DUSTA   Single scattering albedo of dust
19  TAURAT  Ratio of thermal to visible opacity of dust
20  TWILI   Twilight extension angle [deg]
21  ARC2    Henyey-Greenstein asymmetry factor
     moon   = eclipse start time in local Hours
22  ARC3    unused     [coeff. for planetary heating]
     moon   = eclipse duration in seconds 0=no eclipse
23  SLOPE   Ground slope, degrees dip. Only pit may slope beyond pole.
24  SLOAZI  Slope azimuth, degrees east from north. <-360 is a pit
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Frost Properties
25  TFROST  Minimum Frost saturation temperature
            may be overridden by local saturation temperature (LVFT)
26  CFROST  Frost latent heat [J/kg] {cal/gm*4184. [ Not used if
27  AFROST  Frost albedo, may be overridden (LVFA) [ TFROST never
28  FEMIS   Frost emissivity                       [ reached
29  AF1     constant term in linear relation of albedo to solar flux
30  AF2     linear term in relation of albedo to solar flux units=1/flux
              Afrost = AF1 + AF2 * <cos incidence> SOLCON / DAU^2
31  FROEXT  Frost required for unity scattering attenuation coeff. [kg/m^2]
            the greater of this and 0.01 is always used.
32  fd32    unused
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Thermal Solution Parameters
33  RLAY    Layer thickness ratio
34  FLAY    First layer thickness (in skin depths)
35  CONVF   Safety factor for classical numerical convergence
              0 for no binary time division of lower layers
             >0.8 for binary time division. Larger is more conservative
36  DEPTH   Total model depth (scaled) (overrides FLAY if not 0.)
37  DRSET   Perturbation factor in jump convergence. If = 0., then
              all layers reset to same average as surface layer. Else,
             does quadratic curve between surface and bottom averages
38  DDT     Convergence limit of temperature RMS 2nd differences
39  GGT     Surface boundary condition iteration test on temperature
40  DTMAX   Convergence test: RMS layer T changes in a day
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Orbit Geometry & Constants
41  DJUL    Starting Julian date of run -2451545 (J2000.0) (N5>0)
42  DELJUL  Increment between seasons in Julian days (if N5>1)
43  SDEC    Solar declination in degrees. (if Not LPROB)
44  DAU     Distance from Sun in astronomical units (if Not LPROB)
45  SUBS    Aerocentric longitude of Sun, in degrees. For printout
             only. Computed from date unless N5=0(for printout only)
46  SOLCON  Solar constant Applied Optics 1977 v.16, p.2693: 1367.9 W/m^2
             1366.2 Based on figure in Frohlich, Observations of
             irradiance variations, Space Sci. Rev.,94,15-24,2000
47  GRAV    Surface gravity.  MKS-units
48  AtmCp   Specific heat at constant pressure of the atmosphere [J/kg/K]
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Temperature dependent conductivity.  Ignored unless LKOFT set.
49  ConUp0  Constant coef for upper material
50  ConUp1  Linear   in k=c0+c1x+c2x^2+c3x^3 where x=(T-220)*0.01
51  ConUp2  Quadratic    "
```

```
52  ConUp3  Cubic coeff. "
53  ConLo0  Constant coef for lower material
54  ConLo1  Linear      as for ConUp above
55  ConLo2  Quadratic    "
56  ConLo3  Cubic coeff. "
  Temperature dependent specific heat.  Ignored unless LKOFT set.
57  SphUp0  Constant coef for upper material
58  SphUp1  Linear   in k=c0+c1x+c2x^2+c3x^3 where x=(T-220)*0.01
59  SphUp2  Quadratic    "
60  SphUp3  Cubic coeff. "
61  SphLo0  Constant coef for lower material
62  SphLo1  Linear      as for   SphUp above
63  SphLo2  Quadratic    "
64  SphLo3  Cubic coeff. "
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Computed REAL*4 values
65  HUGE    = 3.3E38   nearly largest  REAL*4 value
66  TINY    = 2.0E-38  nearly smallest REAL*4 value
67  EXPMIN  = 86.80  neg exponent that would almost cause underflow
68  FSPARE  Spare
69  FLOST   Atm frost 'lost' in the atm. in last day at current lat./season
70  RGAS    = 8.3145  ideal gas constant  (MKS=J/mol/K)
71  TATMIN  Atmosphere saturation temperature
72  PRES    Local surface pressure at current season
73  OPACITY Solar opacity for current elevation and season
74  TAUIR   current thermal opacity at the zenith
75  TAUEFF  effective current thermal opacity
76  TATMJ   One-layer atmosphere temperature
77  SKYFAC  fraction of upper hemisphere that is sky
78  TFNOW   frost condensation temperature at current latitude
79  AFNOW   frost albedo  at current latitude
80  PZREF   Current surface pressure at 0 elevation, [Pascal]
81  SUMF    Global average columnar mass of frost [MKS]
82  TEQUIL  Equilibrium temperature (no diurnal variation)
83  TBLOW   Numerical limit (Blowup) temperature
84  HOURO   Output Hour requested for "one-point" model
85  SCALEH  Atmospheric scale height
86  BETA    Atmospheric IR absorption
87  DJU5    Current Julian date (offset 2440000 ala PORB convention)
88  DAM     Half length of daylight in degrees
89  EFROST  Frost on the ground at current latitude [kg/m^2] {g/cm^2 * 10.}
90  DLAT    Current latitude
91  COND    Top material Thermal conductivity (for printout only)
92  DIFFU   Top material Thermal diffusivity (for printout only)
93  SCALE   Top material Diurnal skin depth (for printout only)
94  PIVAL   pi
95  SIGSB   Stephan-Boltzman constant (set in KRC)
96  RADC    Degrees/radian

Type 2 Integer Parameters (8I10) ===================================

1   N1      # layers (including fake first layer) (lim MAXN1)
2   N2      # 'times' per day (lim MAXN2). Must be an even number,
             should be a multiple of N24 and NMHA.
3   N3      Maximum # days to iterate for solution (lim MAXN3)
             This can be 1, but then must use  DELJUL ~= PERIOD
```

```
            If N3 lt 3, first day starts on midnight. else at 18H
4   N4     # latitudes (lim MAXN4=19). Global integrations done for N4>8
5   N5     # 'seasons' total for this run. If 0, then DAU and SDEC will be
            used as entered for a single season.
6   N24    # 'hours' per day stored, should be divisior of  N2 (lim MAXNH)
7   IB     Bottom control: 0=insulating, 1=constant temperature
            2=start all layers =TDEEP & constant temperature
8   IC     First layer (remember that 1 is air) of changed properties.
            if 3 to N1-2.   > N1-2 (e.g., 999) =homogeneous
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
9   NRSET  # days before reset of lower layers first season;  >N3=no reset
10  NMHA   # 'hour angles' per day for printout (no limit)
11  NRUN   Run #; appears in some printout. Initalized as 0 and
            auto-increment whenever disk file opened. May be modified
12  JDISK  Season count that disk output is to begin. 0=none
13  IDOWN  Season at which to read change cards
14  I14    Index in FD of flexible print
15  I15     ""
16  KPREF  Mean global pressure control. 0=constant
             1= follows Viking Lander curve  2=reduced by global frost, but
             then N4 must be >8, and latitudes must be monotonic increasing
             and must include both polar regions (no warning for your failure)
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
17  K4OUT  Disk output control: See details in DISK BINARY FILES section
            Three modes of direct access Fortran files;  one case per file.
                 -=KRCCOM(once), then TSF & TPF;
                 0=KRCCOM,LATCOM each season
              1:49=KRCCOM,DAYCOM for the last latitude; each season
             Modes of bin5 file for multiple cases
              51=(Hours, 2 min/max,  lat, seasons, cases)
              52=(hours,  7 items, lat, seasons, cases)
              54=[many seasons, 5 items,lats, cases]
              55=[many seasons,9 items, cases]
              56=[packed T hour and depth, latitude,season,case]
18  JBARE  J5 season count at end of which to set frost amount to 0. 0=never
19  NMOD   Spacing of season for notification. minimum of 1
20  IDISK2 Last season to disk for which to print notice
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Computed I*4 values
21  KOLD   Season index for reading starting conditions
22  KVALB  Flag: to use seasonal surface albedo ALB
23  KVTAU  Flag: 1:TAUD=SEASTAU(SUBS)  2:CLIMTAU opacities for dust and ice
24  ID24(4) spare
28  NFD    Number of real items read in
39  NID    Number of integer items read in
30  NLD    Number of logical items read in
31  N1M1   Temperature vrs depth printout limit (N1-1)
32  NLW    Temperature vrs depth printout increment
33  JJO    Index of starting time of first day
34  KKK    Total # separately timed layers
35  N1PIB  N1+IB Used to control reset of lowest layer
36  NCASE  Count of input parameter sets in one run
37  J2     Index of current time of day
38  J3     Index of current day of iteration
39  J4     Index of current latitude
40  J5     Index of current "season"
```

```
Type 3 Logical Parameters (10L7) ====================================

1   LP1     Print program description. TPRINT(1)
2   LP2     Print all parameters and change cards (2)
3   LP3     Print hourly conditions on last day (3)
4   LP4     Print daily convergence summary (4)
5   LP5     Print latitude summary (5)
6   LP6     Print TMIN and TMAX versus latitude and layer (6)
7   LPGLOB  Print global parameters each season
8   LVFA    Use variable frost albedo. Uses AF1 & AF2 (real # 29,30)
9   LVFT    Use variable frost temperatures
10  LKOFT   Use temperature-dependent conductivity and specific heat
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
11  LPORB   Call PORB1 just after full input set
12  LKEY    Read change item from terminal after main input set
13  LSC     Read change cards from input file at start of each season
14  LNOTIF  spare
15  LOCAL   Use each layer for scaling depth
16  LD16    Print hourly table to fort.76 [TLATS]
17  LPTAVE  Print <T>-<TSUR> at midnight for each layer [TDAY]
18  LD18    spare
19  LD19    Output to fort.79 [TLATS] insolation and atm.rad. arrays
20  LONE    (Computed) Set TRUE if KRC is in the "one-point" mode
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
followed in 'krccom' by:
[real*4] TITLE(20) 80-character title
[real*4] DAYTIM(5) 20-character run  date and time
================================================================

Latitude(s) (10F7.2)   N4 latitudes in degrees, no internal separations.
Latitudes to be in order; south to north. [[If last latitude is
.LE. 0, will assume symmetric results for global integrations]]

Elevation(s) (10F7.2)  N4 values in Km corresponding to latitudes

Orbital Parameters (LPORB=T) Format identical to that produced by PORB
program set ASCII file output. So these can be directly pasted with an
editor. see PORBCM.INC
 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

4 PARAMETER CHANGES

Fortran List Directed. Change the values in KRCCOM White-separated, a "/" terminates the read and leaves remaining values unchanged The 4 required items are:

```
    1: Type (integer): see table below
    2: Index in array (integer): as listed in Input File table above
    3: New value, numeric: will read as real and convert. 0.=false.
    4: Reason, text string within single quotes:
   [ after a / (forward slash) nothing is read, so you can use for comments]
```

The print file will list each change as read, followed by the title of the changed item. It is a good idea to look at this print to be sure you changed what you intended.

```
 Type     Meaning                                           Valid Index

   0   End of Current Changes                                any
   1   Real Parameter                                      1:NFDR
   2   Integer Parameter                                   1:NIDR
   3   Logical Parameter                                   1:NLDR
   4   New Latitude Card(s) Follow                           any
   5   New Elevation Card(s) Follow                          any
   6   New Orbital Parm Cards Follow (LPORB Must be True)  any
   7   Text becomes new Title                                any
   8   Text becomes new disk or season-variation file name
         if index=22, call SEASALB to read variable ALBEDO
         if index=23, call SEASTAU to read variable TAUD
 if index=24, call CLIMTAU to read Mars climate
   9   Complete new set of input follows                     any
  10   Text becomes new One-Point input file name
  11   This is a set of parameters for "one-point" model
          For this type, 9 values must appear in a rigid format
  12   Set of 2*4 coefficents for T-dep. conductivity.  List-directed IO
  13   Set of 2*4 coefficents for T-dep. specific heat. List-directed IO
```

For Type 12 and 13, 8 white-space-separated coefficients must follow after the type on the same line, with no intervening index or text

For Type 8, SEASALB and SEASTAU read 2-column, white-separated text files.
To start variable albedo, use input card:
8 22 0 'AlbedoFileName' / Variable albedo text file name
Can revert to constant albedo by hokey technique of using a bad name. E.g.,
8 22 0 'badName' / turn variable albedo off
Text table files of value versus season will be read at the start of a run. These will apply to ALL latitudes. See example valb1.tab
Variable Tau done the same way, with 22 being replaced with 23

CLIMTAU files have dust and ice opacity over season and latitude. Uses BINF5 to read a binary array (72 seasons, 36 latitudes, 2=dust/ice) of opacities. The sample file *THEMIS1yearDustIce.bin5* is described in section [159] of Ref 1.

5 Contents of COMMOMS

COMMON /KRCCOM/ Input and transfer variables. See krccom.inc
COMMON /DAYCOM/ Layer and time-of-day items. See daycom.inc
COMMON /FILCOM/ File names. See filcom.inc
COMMON /HATCOM/ Store post-2003 items. See hatcom.inc
COMMON /LATCOM/ Latitude-dependent items. See latcom.inc
COMMON /PORBCM/ PORB system geometry matrix. See porbcm.inc
COMMON /UNITS/ Logical units for I/O and errors. See units.inc

Because the binding routines to IDL are intolerant of any errors, changes to KRCCOM, DAYCOM and LATCOM are avoided if possible. Rather, in 2004July HATCOM was added as a "catch-all" for any new items.

A listing of all Fortran commons can be generated by these Linux commands:
cd /home/hkieffer/krc/src [replace top part of path with local installation]
rm allinc.txt
cat krccom.inc latcom.inc daycom.inc hatcom.inc filcom.inc units.inc porbcm.inc > allinc.txt

6 Error Returns

```
 "Parameter error in TDAY(1)" : Convergence factor < .8 classic.
        Instability anticipated.
 "UNSTABLE; Layer..... TDAY(1):

DRSET: 0=>     Reset by delta_average_T for each layer:
                 else: reset by {linear + DRSET*quadratic}*{<surf>-<botm>}
TDAY: LRESET   Reset midnight T's for all but top layer.
      LDAY     Last day computations
```

7 DISK BINARY FILES

The routine TDISK is used to read or write direct-access binary files or bin5 files. The first season to write is specified by JDISK, all following seasons will go to the same file. For direct-access files, each file record consists of KRCCOM plus LATCOM or KRCCOM plus DAYCOM.

Disk output is largely controlled by the KRC and TSEAS routines.

7.1 Items which control file I/O

```
KOLD & KEEP on first input line
  KOLD: 0= input card set follows;  else=disk record number to start from,
         then will read any change cards.
        If LPORB in old file was True, then there must be a PORB card set
          as the set of lines following the KEEP,KOLD line
  KEEP: 0= close disk file after reading seasonal record KOLD;
       >0= value of JJJJJ at which to start saving seasons in same disk
        file [overrides JDISK].
  To start from a prior seasonal run, need to determine the record
  corresponding to the desired season;
         KOLD=J5_target - JDISK(old) ; >0
         set KEEP=1, change card J5=number of new seasons, set K4OUT.

JDISK sets the first season to save results

N5    sets the last season to run

K4OUT sets the record content:
 -      Will output first record of KRCCOM,ALAT,ELEV, then records of TSF & TPF
 0      Will output records of KRCCOM+LATCOM. Usual for large data-base.
 +n<=50 Will output records of KRCCOM+DAYCOM for the last computed latitude.
```

>50 Will write custom bin5 file at the end of a run, with dimensionality from 3 to 5 (more possible). All 5x outputs allow multiple cases, each with a "prefix" for each case consisting of 4 size integers (converted to Float) followed by KRCCOM; after this may come vectors of parameters versus season. The next-to-last dimension is increased to allow room for the prefix to be embedded in the bin5 array. KRC input items that would change any of the bin5 dimensions are not allowed to change between cases. Each dimension is adjusted to the necessary size. Each case has the same structure; this simplifies coding although some items are then present redundantly. The number of cases allowed is set by the size of case one, and printed as MASE at the end of the first case in the print output. Cases beyond the maximum that can be stored will be executed, but not saved.

The first 4 words of the prefix, and of thus of the bin5 array, are:

```
(1)=FLOAT(NWKRC)   ! Number of words in KRCCOM
(2)=FLOAT(IDX)     ! 1-based index of dimension with extra values
(3)=FLOAT(NDX)     ! Number of those extra
```

```
(4)=FLOAT(NSOUT)   ! [Available of other use]

    51=(N24 hours, 2: TSF TPF, N4 lats, NDX+ seasons, cases)
The prefix section contains: sub_array(seasons,5)(0-based index)
  0)=DJU5  1)=SUBS  2)=PZREF  3)=TAUD  4)=SUMF

    52=(N24 hours, 7 items, N4 lats, NDX+ seasons, cases)
The 7 items are:  1)=TSF  2)=TPF  3)=TAF  4)=DOWNVIS  5)=DOWNIR
6) packed with [NDJ4,DTM4,TTA4,      followed by TIN(2+
7) packed with [FROST4,AFRO4,HEATMM, followed by TAX(2+
  The number of layers for TIN and TAX is the smaller of: the number computed
and that fit here.
The prefix is identical to Type 51

    54= (seasons, 5 items, NDX +nlat, cases)
        Items are (0-based index):
        0= TSF=surface temperature at 1 am, 1= TSF at 13 hours,
        2= HEATMM=heat flow, 3= FROST4=frost amount,
        4= TTB4 = predicted mean bottom temperature
        The prefix contains DJU5

    55= (seasons,NDX+ items,cases).  For seasonal studies at one latitude
        ITEMS intended to be recoded as needed. Initial version is 9 items:
        [Tsur@ 1am,3am,1pm, spare, Tplan @1am,1pm, Surface heat flow,
        frost budget, T_bottom]
        The prefix contains DJU5
         Can hold very large number of seasons and cases.
        THIS MODE DOES NOT SUPPORT CONTINUATION RUNS

    56= [vectors&items, latitudes, NDX+ seasons, cases]
The first dimension is: TSF for all hours, TPF at all hours,
 T4 for all layers at midnight, then FROST4,HEATMM,TTA4
The prefix is identical to Type 51
```

Once a disk file is opened, any records written will go into that file until a new filename is specified (Type 8 Change line), which closes the current file. It is best to ensure that output file does not already exist. If the file already exists, new output may be written in same area, even if larger than needed.

8 Handy things

The first "hour" in printout and output arrays is 1/24 (strictly, 1/N24) of a sol after midnight. E.g., the last time is midnight, not the first.

Atmospheric scale height, SCALEH, depends upon physical constants GRAV [input] and TATMAVE which (2007nov) is TATM [input] for the first season and thereafter the diurnal average of the prior season.

To run and save various cases for a single season, set N5 and JDISK to 1.

To extract a detailed day by saving DAYCOM to disk, set JDISK=N5, set a new file name, and set K4OUT to desired latitude index (normally 1):

To run continuously with output every K ((1-3) days, set DELJUL=K*PERIOD This will force prediction terms to near 0.
  setting N3=1 will turn off all prediction.
  set GGT large (to avoid iteration for convergence)
  set NRSET=999 (to avoid reset of layers)

To continue run with new parameters (e.g., DELJUL)

3 21 1 'flag set to continue'
Note: changing DELJUL will cause reset of DJUL
Must increase the value of N5: e.g., 2 5 [bigger] 'Increase stopping season'
Reset will not occur because J5 continues incrementing

8.1 ASCII Output Files

```
krc.prt  General results. Stuff output is controlled by LP1:6 & LPGLOB

fort.76
tlats.f: mimic Mike Mellon ASCII files
        if (ld16) then
          write(76,761)subs,dlat,alb,skrc,taud,pres
 761      format(/,'      Ls      Lt       A       I    TauD       P'
 762      format(f7.2,f9.3,f8.3,f9.3)
            write(76,762)qh,tsfh(i),adgr(j),qs

          do i=1,n24
            j=(i*n2)/n24
            qh=i*qhs
            qs=(1.-alb)*asol(j) ! absorbed insolation
            write(76,762)qh,tsfh(i),adgr(j),qs
          enddo

fort.78
tlats.f:  for average and maximum:
        if (ld18) write(78,*)cosi_(i), t_(i),ADGs(i),ADGP(i)
        if (ld18) write(78,*)j5,j4,sol,ave_a,adgir,c52,beta

fort.79
tlats.f:   for each time-step
       if (ld19) write(79,*)adgr(jj),qa,direct,diffuse
   col 1 = downgoing thermal radiation
   col 2 = total insolation reaching surface
   col 3 = direct  fraction of insolation
   col 4 = diffuse fraction of insolation
```

8.2 To run two material types

Set IC to the first layer to have the lower material properties (¿= 3)
Set COND2 to the lower material conductivity
Set DENS2 to the lower material density
Set SPHT2 to the lower material specific heat
If LOCAL is False, then initial setting of all layer thicknesses is based upon the scale of the upper material; if it is set True, the thickness of the lower layers is set by their scale.
TDAY no longer allows unstable (thin) layers, and will increase the thickness of the layer IC to satisfy the convergence safety factor FCONV if needed. However, the code to check on convergence was retained.

8.3 Setting temperature-dependant properties

Basic Flag is L10=LKOFT . If this is true, then the 8 input parmeters ConUp0 to ConLo3 must be set to yield thermal conductivity as a function of temperature for the upper and lower materials. $k = c0 + c_1 x + c_2 x^2 + c_3 x^3$ where $x = (T - 200.) * 0.01$

Correspondingly, the 8 input parmeters SphUp0 to SphLo3 must be set for specific heat

One way to generate the coefficients is to run for each of the upper and lower materials the IDL procedure KOFTOP, which can call all of the temperature-dependant routines. KOFTOP allows change of its parameters, including grain radius and pressure, and will print the required parameters ready for input to KRC.

Below are sample coefficients for thermal conductivity based on Sylvain Piqueux's numerical model for un-cemented soils; the fit error is $< 0.1\%$ over 120-320K. Left column is grain radius in micrometers, then the four normalized coefficients ready for inclusion in a KRC input file, followed by the thermal inertia at 220K for nominal density and specific heat.

R(mu)	c0	c1	c2	c3	Iner
10.	0.008274	0.000735	-0.000376	0.000148	89.8
20.	0.012379	0.001280	-0.000629	0.000250	109.9
50.	0.021485	0.002647	-0.001201	0.000483	144.7
100.	0.032051	0.004528	-0.001874	0.000761	176.8
200.	0.046023	0.007569	-0.002743	0.001129	211.8
500.	0.068387	0.014075	-0.003874	0.001687	258.2
1000.	0.086303	0.021288	-0.004146	0.002099	290.1
2000.	0.103743	0.030909	-0.003141	0.002535	318.0
5000.	0.127172	0.049907	0.002019	0.003469	352.1
10000.	0.149810	0.074734	0.011546	0.004939	382.2
20000.	0.185706	0.119913	0.030938	0.007877	425.5
50000.	0.283361	0.250283	0.089327	0.016714	525.6

---

9 RUNNING THE "ONE-POINT" MODE

A parameter initialization file *Mone.inp* is provided. It sets the KRC system into a reasonable mode for one-point calculations. Do not change that file unless you have read this entire file.

A line near the end of that file points to a file 'oneA.one' which can contain any number of one-point conditions. You can replace that name with your own; the named file is intended to be edited to contain the cases you want; however, it must maintain the input format of the sample file.

First Line is any title you wish. It must be present.
The second line is an alignment guide for the location lines. It must be there.

Each following line must start with an '11 '; this is a code that tells the full-up KRC that is a one-point line. The next 9 fields are read with a fixed format, and each item should be aligned with the last character of the Column title. All items must be present, each line must extend at least to the m in Azim; comments may extend beyond that, but they will not appear in the output file. Be sure to have a <CR> at the end of the last input line; i.e., no blank lines!

```
The fields (after the 11) in the one-point input are:
Ls  L_sub_S season, in degrees
Lat Aerographic latitude in degrees
Hour Local time, in 1/24'ths of a Martian Day
Elev Surface elevation (relative to a mean surface Geoid), in Km
Alb Bolometric Albedo, dimensionless
Inerti Thermal Inertia, in SI units
Opac Atmospheric dust opacity in the Solar wavelength region
Slop_ Regional slope, in degrees from horizontal
Azim Azimuth of the down-slope direction, Degrees East of North.

The two additional columns in the output file are:
   TkSur Surface kinetic temperature
TbPla Planetary bolometric brightness temperature
```

Try running the binary file first. If that fails, a Makefile is provided to compile and link the program; simply enter "make krc" and pray. If this fails, have your local guru look over the Makefile for local dependancies. Suggestions of making the Makefile more universal are welcome.

To run the program, change to the directory where the program was built, and enter "krc". You should get a prompt:
?* Input file name or / for default = Mone.inp
If the initialization file still has this name and is in the same directory, enter a single "/" and <CR>. Otherwise, enter the full pathname to the initialization file, with no quotes and no blanks.

A second prompt is for the name of the output file:
?* Print file name or / for default = krc.prt
Again, if this is satisfactory, simply enter / <CR> , else enter the desired file path-name.

9.0.1 Comments on the One-point model

The initialization file *Mone.inp* is set to compute the temperatures at the season requested without seasonal memory. It uses layers that extend to 5 diurnal skin depths. It does not treat the seasonal frost properly, so don't believe the results near the edge of the polar cap. Execution time on a circa 2001 PC may be the order of 0.01 seconds per case.

The underlying model is the full version of KRC. By modifying the initialization file, you can compute almost anything you might want. If you choose to try this, best to read all of this document.

10 DEBUG OPTIONS

If the first input line has a no-zero third number, then the second line is 6 white-separated debug-control integers: IDB1 to IDB6

```
tcard.f:75:     IF (IDB2.GE.5) WRITE(IOSP,*) 'TCARD-A',IQ
tcard.f:123:    IF (IDB1.GE.1) PRINT *,'Before PORBO'
tcard.f:125:    IF (IDB1.GE.1) PRINT *,'AFTER PORBO'
tcard.f:349:    IF (IDB1.NE.0) WRITE(IOSP,*)'TCARD Exit: IRET=',IRET,NFD,ID(1)
tday.f:63:      IF (IDB2.GE.5) WRITE(IOSP,*) 'TDAY IQ,J4=',IQ,J4,jjo
tday.f:544: 9   IF (IDB2.GE.6) WRITE(IOSP,*) 'TDAYx'
tdisk.f:90:     IF (IDB3.NE.0) WRITE(IOSP,*)'TDISKa ',KODE,KREC,NCASE,J5,K4OUT
tdisk.f:424:       IF (IDB3.GE.3) WRITE(IOSP,*)'TDISKc  KREC=',KREC,LOPN2,IOD2,I
tdisk.f:431:    IF (IDB3.GE.3) WRITE(IOSP,*)'TDISKx  KREC=',KREC
tlats.f:56:     LQ1=IDB2.GE.3           ! once per season or latitude
tlats.f:+       LQ2=IDB2.GE.6           ! each day
tlats.f:+       IF (IDB2.NE.0) WRITE(IOSP,*)'TLATSa',N3,N4,J5,LATM,LQ1,LQ2
tlats.f:422: 9      IF (IDB2.GE.3) WRITE(IOSP,*)'TLATSx',N1,N1PIB,N2,N24,J3
tseas.f:41:     IF (IDB1.NE.0) WRITE(IOSP,*)'MSEASa',IQ,IR,J5,LSC,N5,LONE
in tlats.f:
 98: IF (LQ1) THEN
   WRITE(75,*) 'J5+',J5,SUBS,SLOPE,SLOAZI,SKYFAC
   WRITE(75,*) 'MXX+',MXX,DIP,SAZ
   WRITE(75,*) 'PXX+',PXX
top of Lat loop
130:    IF (LQ1) PRINT *,'TLAT1 J5,TBLOW=',J5,TBLOW
170: IF (LQ1) THEN
   WRITE(75,*)'FXX+',FXX,J4,DLAT
   WRITE(75,*)'RXX=',RXX  ! R should be 90 deg from F
   WRITE(75,*)'TXX=',TXX
221:    IF (LQ1) print *,'TLATS: J4,SOLR...',J4,SOLR,ACOSLIM,COSIAM(1)
 top of time loop
299:    IF (LQ1.AND.(MOD(JJ,24).EQ.1)) THEN
            WRITE(75,*)'HXX+',HXX,JJ
              WRITE(75,*)'ANG:',ANGLE,COSI,COS2,DIRECT,QI
303:    IF (LQ2) WRITE(IOSP,*),'TLatc',JJ,COSI,COS3,DIRECT,DIFFUSE
```

```
309:    IF (LD19) WRITE(79,777) QA,QI,DIRECT,DIFFUSE,BOUNCE
 end time loop
341:    IF (LQ1) then
           PRINT *,'AVEA ...',AVEA,AVEE,AVEI,AVEH
           PRINT *,'CABR...',CABR,TAUD,TAUIR,FACTOR,TAUEFF
           PRINT *,'BETA...',BETA,QS,SIGSB
           PRINT *,'TAEQ4,TSEQ4,TEQUIL',TAEQ4,TSEQ4,TEQUIL
356:       IF (LQ1) PRINT *,'TSUR,TBOT',TEQUIL,TSUR,TBOT
           IF (LQ1) PRINT *,'XCEN',XCEN
379:    IF (LQ1) PRINT *,'TTJ',TTJ
453:    IF (LD16) THEN
   WRITE(76,761)SUBS,DLAT,ALB,SKRC,TAUD,PRES
           loop on hour: WRITE(76,762)QH,TSFH(I),ADGR(J),QS,TPFH(I)
end lat loop
```

Set LD19 to write bottom-of-atmosphere downgoing fluxes to separate file for every time-step for every latitude, every time tlats is called.

11 Reading type 5x files

IDL routines do not access files directly unless specifically listed.

DEFINEKRC Define structures in IDL that correspond for Fortran commons
Calls: None == None other than IDL library
Firm code of common definitions. Must be recoded if a Fortran *.inc changes

KRCSIZES Compute array and common sizes for KRC Fortran
Test procedure to compute array sizes or hours.
Must recode if any size in *.inc changes
Calls: None

READKRCCOM Read a KRCCOM structure from a bin5 file
Uses 3-element HOLD array. Returns a structure of krccom
Options to open or close bin5 file or read one case
Calls: DEFINEKRC
Files: bin5
HOLD is: 0]=logical unit 1]=number of words in a case 2]=# cases in the file

KRCHANGE Find changes in KRC input values in common KRCCOM
Calls: READKRCCOM MAKEKRCVAL
Reads and stores krccom for first case. For each additional case, makes a list of any changes in the flaot, integer or logical input values.

KRCCOMLAB Print KRC common input items
all items via arguments
Calls: None

MAKEKRCVAL Make string of selected KRC inputs: Key=val
Calls: DEFINEKRC

KRCLAYER Compute center depth of KRC layers
all items via arguments
Calls: None

12 Notes on how some aspects of the code work

12.1 New file name

TCARD reads a card of Type 8, (and index is not 22 or 23)
it calls TDISK(4,0), which closes current file and sets LOPN2=.FALSE.
TCARD then moves new file name into common
KRC checks if current (new) values of N5 and JDISK call for file output;
with LOPN2=.FALSE., KRC calls CALL TDISK (1,0) to open new file.

12.2 End of a case and end of a run

TCARD sets KOUNT=0 at entry; this is incremented for every card except those of type 0 (or less) or type 11 (one-point mode). When type 0 is encountered, if KOUNT is positive, does normal check of changes before return with IR=1 to indicate start of a new case; if KOUNT is zero, returns with IR=5 and prints 'END OF DATA ON INPUT UNIT'

12.3 Setting one-point mode

This can be done only in the first case, and there is no way to leave the one-point mode except to end the run.

TCARD encounters: " 10 * filename" as change card in the initial case.
sets this as new input file name, then returns with IRET=4
[Thus, nothing following this change card in initial file is read]
KRC closes prior input file, opens the new one, and reads past first two lines
then calls TCARD to read first one-point line and sets LONE=true
and drops into the top of the "case" loop.
The master one-point should have a single latitude, no binary output file.
The small number of layers, days to converge, and seasons ignores the seasonal effect.

One-point request values are read by TCARD @ 310, which computes starting DJUL

TPRINT does linear interpolation of TOUT, which has N2 points be sol. To get Tp, does interpolation of Tp-Ts at the hour points, and adds to interpolated Ts.

12.3.1 How one-point converts Ls to date

Ver 212: XREAD is the 2nd column in the OnePoint file, i.e., Ls.

In TCARD 310: calls PORBIT to get the date of the desired Ls, then backs up (N5-1)*DELJUL to the starting date.

12.4 Starting conditions and date

Initial N5-JDISK sets the size of output files. There could be any number of interior seasons where parameter changes are made; based on successive values of IDOWN.

KRC initially calls TCARD(1
For each case loop, sets IQ=TCARD_return. If one-point mode, sets IQ=1

TSEAS uses IQ as key. It this is 1, then sets J5=0 and sets DJU5 to season -1., else, increments J5 and increments DJU5 with current DELDUL. This allows use of variable resolution dates. (so J5 never 0 when TCARD(2 called)
If J5 equals IDISK2, then TSEAS calls TCARD(2 to read changes, and proceeds to next season.

TLATS uses J5 as the key; if it is <= 1, then starts from equilibrium conditions, else uses predictions from prior season

The default is that change cards cause a fresh calculation of starting conditions. Exceptions are when J5=IDOWN> 0 at TCARD entry

12.5 Changing parameters within a seasonal run = Continue from memory.

When J5 reaches IDOWN, TSEAS calls TCARD, which will set IRET=3 before reading the new parameters. May change DELJUL to get finer seasonal resolution, but must NOT change N5

Use: Normal restrictions for what may not change for Type 5x files apply. E.g., type 56 must NOT change number of latitudes nor total number of seasons.

Set N5 to be the total number of seasons desired, including those after any number of parameter changes; it must NOT be changed later.

Set IDOWN to the season at the beginning of which wish to (first) change parameters. The next set of changes could include a revised (larger) IDOWN.

12.6 Use of common PORBCM

Contents are described in porbcm.inc
PORBCM is filled by TCARD calling PORB0, which reads the first 30 items in 5G15.7 from the input file and sets the value of π and radians-to-degrees. KRC references porbcm.inc but does not use it.

TSEAS call PORBIT to get Ls, the heliocentric range and the sub-solar latitude.

12.7 Lower boundary condition and resetting (jumping) layer temperatures.

At the start of a case, TLATS sets the temperature profile linear with depth in one of three ways:
IB=0: top and bottom at equilbrium temperature
IIB=1: top at equilbrium temperature, the bottom at TDEEP
IIB=2: top and bottom at TDEEP

The kind of resetting is controlled by IB. In TCARD, if IB>0, then N1PIB=N1+1, else N1PIB=N1. T(N1+1) is not reset in the time calculations. In TDAY, for each time step, the temperature of the lower boundary is set equal to T(N1PIB), which results in either zero heat flow (IB=0) or a constant temperature.

12.8 Seasonal variation of albedo or opacity or "climate"

When TCARD encounters a type of 8 and an index of 23(tau) [or 22(albedo)], it transfers the text item into FVTAU (which is in COMMON /FILCOM/) and then calls SEASTAU with an Ls of -999 . SEASTAU when called with LSUB LT -90 calls (providing IOD3) READTXT360, which reads file. Maximum number of rows is 360, more will be ignored. First and last entry read are wrapped with ± 360 to Ls to ensure no interpolation faults later. TCARD sets the variable Tau flag, KVTAU, true if table-read was successful, else it is set false. If KVTAU is set, TSEAS calls SEASTAU at start of each season, resetting TAUD.

If type 8 and index 23, the same as above except names are -ALB rather than -TAU

If type 8 and index 24, then TCARD calls CLIMTAU to open and read a .bin5 climate file, and sets KVTAU=2

CLIMTAU expects to read a .bin5 file (season,latitude,2) with dust and ice infrared opacities; this file is normally made by the IDL routine mopacity.pro . Season is assumed to the uniform in Ls from 0 to 360-delta and latitude assumes to be uniform from -90+delta/2 to +90-delta/2. CLIMTAU has firm-coded sizes, 72 seasons and 36 latitudes, and stores the file. Upon later calls from TLATS, it returns the two opacities at a requested Ls and latitude, using bi-linear interpolation.

12.9 Cap-dependent pressure

```
TSEAS:   BUF(1)=0.       ! flag for  TINT to compute areas
IF (N4.GT.8) CALL TINT (FROST4, BUF, SUMF)

Tlats
        IF (N4.GT.8) THEN       ! use global integrations
```

```
         PCAP = SUMF*GRAV      ! cap_frost equivalent surface pressure

KPREF.EQ.2
PZREF = PTOTAL - PCAP
         PCO2G = PCO2M -PCAP ! all changes are pure CO2
```