# RTTOV-7 - TECHNICAL REPORT

*Roger Saunders*
*Met Office*
*&*
*Pascal Brunel*
*MétéoFrance*

This documentation was developed within the context of the EUMETSAT Satellite Application Facility on Numerical Weather Prediction (NWP SAF), under the Cooperation Agreement dated 25 November 1998, between EUMETSAT and the Met Office, UK, by one or more partners within the NWP SAF. The partners in the NWP SAF are the Met Office, ECMWF, KNMI and Météo France.



| Change record | | | |
|---|---|---|---|
| Version | Date | Author / changed by | Remarks |
| 1 | 19/3/02 | R.W. Saunders | Draft |
| 2 | 27/5/02 | R.W. Saunders | Updated to correct IFAIL documentation |
| | | | |
| | | | |
| | | | |
| | | | |

**RTTOV-7 - TECHNICAL REPORT....1**

**COPYRIGHT 2002, EUMETSAT, ALL RIGHTS RESERVED. ....1**

**1. INTRODUCTION AND SCOPE ....3**

**2. GENERAL SOFTWARE IMPLEMENTATION DETAILS....3**

**3. TECHNICAL CHANGES FROM RTTOV-6 ....3**

*3.1 COEFFICIENT FILE INGEST*....3
*3.2 SUBROUTINE INTERFACES*....4
*3.3 ADDITIONAL CLOUD ROUTINES* ....5

**4. DOCUMENTATION OF RTTOV-7 CODE....5**

**5. RTTOV-7 COEFFICIENT FILES....7**

**ANNEX-A: PLATFORMS AND COMPILERS TESTED ....9**

**ANNEX-B: LIST OF RT COEFFICIENT FILES ....10**

**ANNEX-C: EXAMPLE OF RT COEFFICIENT FILE HEADER ....12**

**ANNEX-D: RTTVI SUBROUTINE INTERFACE....14**

**ANNEX-E: RTTOV SUBROUTINE INTERFACE ....15**

**ANNEX-F: RTTOVK SUBROUTINE INTERFACE....16**

**ANNEX-G: RTTOVTL SUBROUTINE INTERFACE....17**

**ANNEX-H: RTTOVAD SUBROUTINE INTERFACE....18**

**ANNEX-I: RTTOVCLD SUBROUTINE INTERFACE ....19**

**ANNEX-J: RTTOVCLDTL SUBROUTINE INTERFACE....20**

**ANNEX-K: RTTOVCLDAD SUBROUTINE INTERFACE ....21**

**ANNEX-L: RTTOVCLDK SUBROUTINE INTERFACE ....22**

**ANNEX-M: RTTOV-7 FORTRAN 90 PARAMETERS ....23**

**ANNEX-N: RTTOV-7 FORTRAN 90 GLOBAL VARIABLES ....25**

## 1. Introduction and Scope

The purpose of this report is to document the more detailed technical aspects of the RTTOV-7 software. A users guide is available for an overview and installation of RTTOV-7 on to a users computer system. Scientific and validation documentation can be found in the RTTOV-7 scientific and validation report. This report may be updated from time to time. Updates will be published on the NWP-SAF web site on the RTTOV-7 page at:

http://www.metoffice.com/research/interproj/nwpsaf/rtm/ along with all the other documentation.

## 2. General Software Implementation Details

The model is only available in standard Fortran 90 code and unix scripts are prepared to run the test code supplied with the basic model. The FORTRAN 90 standard defined in Andrews *et. al.* (1995) *'European Standards for Writing and Documenting Exchangeable Fortran 90 Code'* has been followed wherever possible. The subroutine headers contain documentation on the input/output variables and so the user should study these headers to get details of the interface between modules. An overall routine calling tree is given in Figures 1 and 2. The code has been optimised for vector machines and also made threadsafe for MPP platforms. It has been tested on the following platforms: HP, SGI, Cray T3E, Fujitsu VPP, SUN and compiler options for these are supplied in the Makefile. Annex A gives a more complete list of platforms and compilers on which the code has been tested.

The code, scripts, test files and coefficient files are available either via ftp (as a compressed UNIX tar file) or on a CD-ROM from ECMWF data services.

## 3. Technical Changes from RTTOV-6

The scientific changes are outlined in detail in the scientific and validation report but the technical changes are given below and users who want to change from RTTOV-6 to RTTOV-7 should note these changes.

### *3.1 Coefficient file ingest*

A list of the platforms and sensors which RTTOV-7 supports is given in Table 1. The coefficient file ingest has been completely revised so that each sensor has a separate file. A list of the current coefficient files is given in Annex B. The file naming system uniquely identifies the sensor, platform and platform id (e.g. NOAA-15 AMSU-B gives a file name of *rtcoef_noaa_15_amsub.dat)*. The contents of each file are in ascii which allows it to be well commented. An example is given in Annex C. If I/O is a problem (note the AIRS file has >2378*36*43 coefficients to read in) then the RTCOEF.f90 routine could be modified to read in binary versions of the file. This is only likely to be a problem for AIRS as the other files are relatively small. The user must provide a link to the files with the correct file names in the same directory as the executable code is running.

| Platform | RTTOV id | Sat id range |
|---|---|---|
| NOAA | 1 | 1 to 16 |
| DMSP | 2 | 8 to 16 |
| Meteosat | 3 | 5 to 7 |
| GOES | 4 | 8 to 12 |
| GMS | 5 | 5 |
| FY-2 | 6 | 2 |
| TRMM | 7 | 1 |
| ERS | 8 | 1 to 2 |
| EOS | 9 | 1 to 2 |
| ENVISAT | 11 | 1 |
| MSG | 12 | 1 |
| FY-1 | 13 | 3 to 4 |

| Sensor | RTTOV id | Sensor Channel # | RTTOV Channel # |
|---|---|---|---|
| HIRS | 0 | 1 to 19 | 1 to 19 |
| MSU | 1 | 1 to 4 | 1 to 4 |
| SSU | 2 | 1 to 3 | 1 to 3 |
| AMSU-A | 3 | 1 to 15 | 1 to 15 |
| AMSU-B | 4 | 1 to 5 | 1 to 5 |
| AVHRR | 5 | 3b to 5 | 1 to 3 |
| SSMI | 6 | 1 to 7 | 1 to 7 |
| VTPR1 | 7 | 1 to 8 | 1 to 8 |
| VTPR2 | 8 | 1 to 8 | 1 to 8 |
| TMI | 9 | 1 to 9 | 1 to 9 |
| SSMIS | 10 | 1 to 24* | 1 to 24* |
| AIRS | 11 | 1 to 2378 | 1 to 2378 |
| MODIS | 13 | 1 to 17 | 1 to 17 |
| ATSR | 14 | 1 to 3 | 1 to 3 |
| MVIRI | 20 | 1 to 2 | 1 to 2 |
| SEVIRI | 21 | 4 to 11 | 1 to 8 |
| GOES-Imager | 22 | 1 to 4 | 1 to 4 |
| GOES-Sounder | 23 | 1 to 18 | 1 to 18 |
| GMS imager | 24 | 1 to 2 | 1 to 2 |
| *FY-2VISSR* | 25 | 1 to 2 | 1 to 2 |
| *FY1-MVISR* | 26 | 1 to 3 | 1 to 3 |

**channels 19-21 are not simulated accurately*

*Table 1. Platforms and sensors supported by RTTOV-7 as at 1 Jan 2002. Sensors in italics are only supported by RTTOV-7.*

### *3.2 Subroutine interfaces*

The top level subroutines which is the interface to the users program have changed their interfaces. Details of these are given in Annexes D-H (also in the users guide). All arrays must be initialised on input to the RTTOV routines and the annexes give typical values for the input/output variables. Note in particular the *radov* array has increased its dimensions from that in RTTOV-6.

### *3.3 Additional cloud routines*

A set of routines RTTOVCLD are supplied with RTTOV-7 which allow the user to compute more realistic cloudy radiances. The calling sequence for these routines are given in Figure 3. They are similar to the RTTOV model routines but are a layer above RTTOV. Their interfaces are given in Annexes I - L.

## 4. Documentation of RTTOV-7 code

RTTOV-7 consists of a setup routine and 4 associated models:

- ***RTTVI*** routine to set up arrays for RTTOV
- ***RTTOV*** the radiative transfer model itself
- ***RTTOVTL*** its tangent linear model,
- ***RTTOVAD*** its adjoint model and
- ***RTTOVK*** its gradient matrix model.

If you are only interested in the forward model and not the gradient routines then the TL/K/AD routines are not required which significantly reduces the number of routines you need.

Firstly ***RTTVI*** must be called to set up the necessary arrays for the platform required (e.g. NOAA, METEOSAT, DMSP, GOES, …) and sensor (e.g. HIRS, AMSU-A, MVIRI, SSM/I…) and the satellite ids for each series (as defined in Table 1). The module ***MOD_CPARAM.F90*** file defines the array sizes for running RTTOV-7. It is recommended the user modifies this file to set the array size for his particular application. The variables in the file are defined in Table 2. If the rt_coefficient file supplied with the code is used then only those variables in the top part of the table should be modified. ***JPNSAT*** refers to the maximum number of sensors to be used at any one time, ***JPPF*** to the maximum number of profiles to be processed in any one call to RTTOV, ***JPCH*** to the total number of channels of the sensor with the greatest number of channels required to be simulated (e.g. for ATOVS it would be 20 from HIRS). ***JPCHUS*** is the maximum number of channels required to be simulated by RTTOV which can be the same or less than ***JPCH***.

For sensors with large number of channels (e.g. AIRS) the ***IVCH*** array and ***NUMCHANS*** array can be set to non-zero values on input to ***RTTVI*** to allow coefficients for *only* those channels required to be read in to memory. In this case the number of channels required is in NUMCHANS and their number are given in the IVCH array. If IVCH is zero on input, the default, coefficients for all valid channels are read into memory.

***NIU1*** is an optional parameter input to ***RTTVI*** and if set to a non-zero value defines the fortran unit number through which the coefficient files are read in. This file should have already been opened in this case in the routine calling ***RTTVI***.

| Parameter | Description | Default value |
| --- | --- | --- |
| **Users should edit these to optimise array sizes** | | |
| JPPF | Max number of profiles per call | 3‡ |
| JPCH | Max number of channels of any sensor per call | 2378 |
| JPCHUS | Max number of channels to simulate in any one call | 200 |
| JPNSAT | Max number of sensors to simulate | 3 |
| **Global parameters only changed by code developers** | | |
| JPLEV | Number of pressure levels | 43 |
| JPNAV | Number of profile variables | 4 |
| JPNSAV | Number of surface air variables | 5 |
| JPNSSV | Number of skin variables | 6 |
| JPNCV | Number of cloud variables | 2 |
| JPCHPF | JPPF*JPCHUS | 600 |
| JPCOFM | Max number of mixed gas coeffs | 15 |
| JPCOFW | Max number of water vapour coeffs | 15 |
| JPCOFO | Max number of ozone coeffs | 15 |
| JPST | Max number of surface types | 10 |
| IU1 | Unit number for reading coeff file | 10 |
| NULOUT | Unit number for error messages | 6 |
| JMWCLDTOP | Upper level for LWP calcs for MW chans | 25 |
| JPPLAT | Max number of platforms | 15 |
| JPINST | Max number of sensors (0 rel) | 30 |
| JPGAS | Number of gases | 3 |

‡ (set to 1 for scalar machine, and to ~50 for a vector machine for optimal performance)

*Table 2. RTTOV-7 module MOD_CPARAM.F90*

***RTTVI*** sets up the arrays and loads in all the constants from the rt_coefficient file(s) and is only called once for all sensors. ***RTTOV*** actually performs the RT calculation for the specified satellite ids and channel numbers given valid profile arrays. The subroutine calling structure for ***RTTVI*** and ***RTTOV*** is shown in Figures 1 and 2. For users who require the tangent-linear, adjoint or K routines of RTTOV-7 the calls are ***RTTOVTL***, ***RTTOVAD*** and ***RTTOVK*** respectively with the same subroutines called inside with the endings TL, AD, K. The details of the calling interfaces are given in Annexes D-H. For cross reference a table of the RTTOV-7 parameters and which routines use them is given in Annex M and global variables used in modules is given in Annex N.

## 5. RTTOV-7 COEFFICIENT FILES

The RT coefficient files contain all the coefficients required by RTTOV-7 specific to a particular instrument and platform. They also define some of the fundamental constant values from which the coefficients are computed to ensure consistency throughout. There is a different coefficient file for each sensor as defined in Annex B. Annex C gives a typical example of the contents of 1 file. The coefficient files required (or a symbolic link to these) need to reside in the same directory as the executable file of RTTOV-7 (see installation guide). The cofficient files may be updated from time to time and can be downloaded from the web site. This will be announced on the RTTOV-7 web page and also on the rttov email group.

Code to write RTTOV-7 coefficient files is available on request from the NWP-SAF on an 'as is basis' but is not supported or part of the RTTOV-7 export package. Note that RTTOV-7 coefficient files cannot be used with the RTTOV-6 or RTTOV-5 codes.

**Figure 1 Subroutine tree for RTTVI**

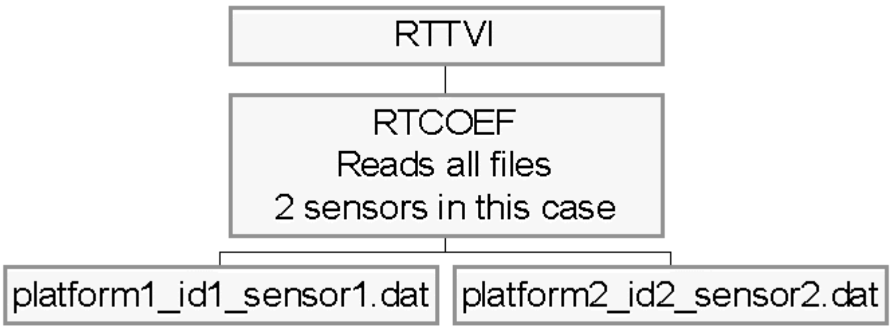
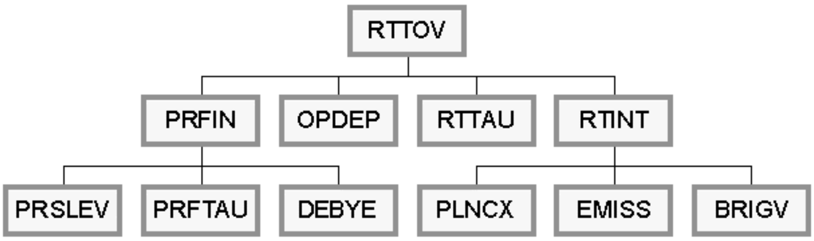
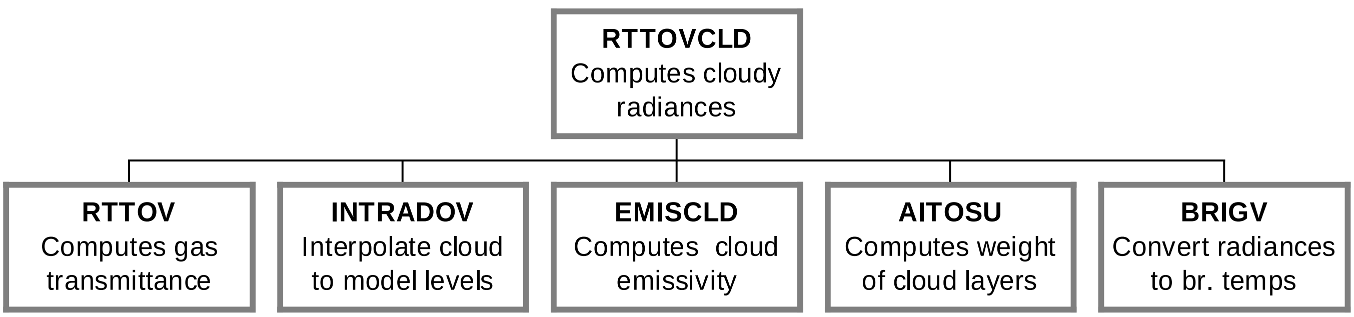


**Figure 2. Subroutine tree for RTTOV-7 main call**




## Figure 3. Calling tree for RTTOVCLD



## ANNEX-A: Platforms and compilers tested

**SUN**
machine: SunOS 5.7 Ultra-250
compiler: f90: Sun WorkShop 6 update 1 Fortran 95 6.1 2000/09/11
options: FFLAGS= -O3 -M. -xtypemap=real:64,double:64,integer:32

**SUN**
machine: SunOS 5.7 Ultra-250
compiler: frt: Fujitsu Fortran Compiler Driver Version 4.0.2.1 (Nov 26 1999 22:23:15)
options: FFLAGS=-Ad -Am -O2 -M .

**Fujitsu VPP5000**
machine: UNIX_System_V cbar1 4.1 ES 3 5000 UXP/V
compiler: frt : Fujitsu UXP/V Fortran V20L20 Driver L99121
UL61560 (Jan 25 2000 21:26:19)
options: FFLAGS=-Ad -Am -O3 -M .

**HP**
machine: HP-UX andante B.10.20 A 9000/800
compiler: HP F90 v2.4
FFLAGS =+autodbl -O2 +check=all -I .
FFLAGS =+autodbl -O2 -I .====> much faster in excution time

**HP**
Machine: HP-UX B.11.00 A 9000/785
Compiler: NAGWare Fortran 95 compiler Release 4.1(329)
FFLAGS=-gline -r8 -C=all -nan

**SGI**
machine: SGI IRIX64 6.5
Compiler: MIPSpro version 7.3.1.1m
FFLAGS=-r8 -trapuv -g

**CRAY T3E**
machine: CRAY T3E sn6702 2.0.5.52 unicosmk
compiler: Cray CF90 Version 3.3.0.2
FFLAGS=-R8 +check=all

## ANNEX-B: List of RT coefficient files

| RTTOV-7 coefficient file organisation | | | | | |
|---|---|---|---|---|---|
| **Platform** | **Sat id** | **Instrument** | **RTTOV ids** | **Channels** | **File name** |
| **1** | | | | | |
| NOAA | 2 | VTPR1 | 1,1,7 | 1 to 8 | rtcoef_noaa_1_vtpr1.dat |
| NOAA | 2 | VTPR2 | 1,1,8 | 1 to 8 | rtcoef_noaa_1_vtpr2.dat |
| NOAA | 3 | VTPR1 | 1,2,7 | 1 to 8 | rtcoef_noaa_2_vtpr1.dat |
| NOAA | 3 | VTPR2 | 1,2,8 | 1 to 8 | rtcoef_noaa_2_vtpr2.dat |
| NOAA | 4 | VTPR1 | 1,3,7 | 1 to 8 | rtcoef_noaa_3_vtpr1.dat |
| NOAA | 4 | VTPR2 | 1,3,8 | 1 to 8 | rtcoef_noaa_3_vtpr2.dat |
| NOAA | 5 | VTPR1 | 1,4,7 | 1 to 8 | rtcoef_noaa_4_vtpr1.dat |
| NOAA | 5 | VTPR2 | 1,4,8 | 1 to 8 | rtcoef_noaa_4_vtpr2.dat |
| TIROS-N | | HIRS | 1,5,0 | 1 to 20 | rtcoef_noaa_5_hirs.dat |
| TIROS-N | | MSU | 1,5,1 | 1 to 4 | rtcoef_noaa_5_msu.dat |
| TIROS-N | | SSU | 1,5,2 | 1 to 3 | rtcoef_noaa_5_ssu.dat |
| TIROS-N | | AVHRR | 1,5,5 | 1 to 2 | rtcoef_noaa_5_avhrr.dat |
| NOAA | 6 | HIRS | 1,6,0 | 1 to 20 | rtcoef_noaa_6_hirs.dat |
| NOAA | 6 | MSU | 1,6,1 | 1 to 4 | rtcoef_noaa_6_msu.dat |
| NOAA | 6 | SSU | 1,6,2 | 1 to 3 | rtcoef_noaa_6_ssu.dat |
| NOAA | 6 | AVHRR | 1,6,5 | 1 to 2 | rtcoef_noaa_6_avhrr.dat |
| etc | | | | | |
| NOAA | 15 | HIRS | 1,15,0 | 1 to 20 | rtcoef_noaa_15_hirs.dat |
| NOAA | 15 | AMSU-A | 1,15,3 | 1 to 20 | rtcoef_noaa_15_amsua.dat |
| NOAA | 15 | AMSU-B | 1,15,4 | 1 to 5 | rtcoef_noaa_15_amsub.dat |
| NOAA | 15 | AVHRR | 1,15,5 | 1 to 3 | rtcoef_noaa_15_avhrr.dat |
| NOAA | 16 | HIRS | 1,16,0 | 1 to 20 | rtcoef_noaa_16_hirs.dat |
| NOAA | 16 | AMSU-A | 1,16,3 | 1 to 20 | rtcoef_noaa_16_amsua.dat |
| NOAA | 16 | AMSU-B | 1,16,4 | 1 to 5 | rtcoef_noaa_16_amsub.dat |
| NOAA | 16 | AVHRR | 1,16,5 | 1 to 3 | rtcoef_noaa_16_avhrr.dat |
| | | | | | |
| **2** | | | | | |
| DMSP | 8 | SSM/I | 2,8,6 | 1 to 7 | rtcoef_dmsp_8_ssmi.dat |
| etc | | | | | |
| DMSP | 13 | SSM/I | 2,13,6 | 1 to 7 | rtcoef_dmsp_13_ssmi.dat |
| DMSP | 14 | SSM/I | 2,14,6 | 1 to 7 | rtcoef_dmsp_14_ssmi.dat |
| DMSP | 15 | SSM/I | 2,15,6 | 1 to 7 | rtcoef_dmsp_15_ssmi.dat |
| DMSP | 16 | SSM/I(S) | 2,16,10 | 1 to 24 | rtcoef_dmsp_16_ssmis.dat |
| | | | | | |
| **3** | | | | | |
| METEOSAT | 5 | MVIRI | 3,5,20 | 1 to 2 | rtcoef_meteosat_5_mviri.dat |
| METEOSAT | 6 | MVIRI | 3,6,20 | 1 to 2 | rtcoef_meteosat_6_mviri.dat |
| METEOSAT | 7 | MVIRI | 3,7,20 | 1 to 2 | rtcoef_meteosat_7_mviri.dat |
| | | | | | |
| **4** | | | | | |
| GOES | 8 | Imager | 4,8,22 | 1 to 4 | rtcoef_goes_8_imager.dat |

| | | | | | |
|---|---|---|---|---|---|
| GOES | 8 | Sounder | 4,8,23 | 1 to 18 | rtcoef_goes_8_sounder.dat |
| GOES | 9 | Imager | 4,9,22 | 1 to 4 | rtcoef_goes_9_imager.dat |
| GOES | 10 | Imager | 4,10,22 | 1 to 4 | rtcoef_goes_10_imager.dat |
| GOES | 10 | Sounder | 4,10,23 | 1 to 18 | rtcoef_goes_10_sounder.dat |
| GOES | 11 | Imager | 4,11,22 | 1 to 4 | rtcoef_goes_11_imager.dat |
| GOES | 11 | Sounder | 4,11,23 | 1 to 18 | rtcoef_goes_11_sounder.dat |
| GOES | 12 | Imager | 4,12,22 | 1 to 4 | rtcoef_goes_12_imager.dat |
| | | | | | |
| **5** | | | | | |
| GMS | 5 | Imager | 5,5,24 | 1 to 2 | rtcoef_gms_5_imager.dat |
| | | | | | |
| **6** | | | | | |
| FY-2 | 2 | VISSR | 6,2,25 | 1 to 2 | rtcoef_fy2_2_vissr.dat |
| | | | | | |
| **7** | | | | | |
| TRMM | 1 | TMI | 7,1,9 | 1 to 9 | rtcoef_trmm_1_tmi.dat |
| | | | | | |
| **8** | | | | | |
| ERS | 1 | ATSR | 8,1,14 | 1 to 3 | rtcoef_ers_1_atsr.dat |
| ERS | 2 | ATSR | 8,2,14 | 1 to 3 | rtcoef_ers_2_atsr.dat |
| | | | | | |
| **9** | | | | | |
| TERRA | 1 | MODIS | 9,1,13 | 1 to 17 | rtcoef_eos_1_modis.dat |
| AQUA | 2 | MODIS | 9,2,13 | 1 to 17 | rtcoef_eos_2_modis.dat |
| AQUA | 2 | AIRS | 9,2,11 | 1 to 2378 | rtcoef_eos_2_airs.dat |
| AQUA | 2 | AMSU-A | 9,2,3 | 1 to 20 | rtcoef_eos_2_amsua.dat |
| AQUA | 2 | HSB | 9,2,12 | 1 to 4 | rtcoef_eos_2_hsb.dat |
| AURA | 3 | HIRDLS | 9,3,? | 1 to ?? | rtcoef_eos_3_hirdls.dat |
| | | | | | |
| **10** | | | | | |
| METOP | 1 | HIRS | 10,1,0 | 1 to 20 | rtcoef_metop_1_hirs.dat |
| METOP | 1 | AMSU-A | 10,1,3 | 1 to 20 | rtcoef_metop_1_amsua.dat |
| METOP | 1 | MHS | 10,1,15 | 1 to 5 | rtcoef_metop_1_mhs.dat |
| METOP | 1 | AVHRR | 10,1,5 | 1 to 3 | rtcoef_metop_1_avhrr.dat |
| METOP | 1 | IASI | 10,1,16 | 1 to 8461 | rtcoef_metop_1_iasi.dat |
| | | | | | |
| **11** | | | | | |
| ENVISAT | 1 | AATSR | 11,1,14 | 1 to 3 | rtcoef_envisat_1_aatsr.dat |
| | | | | | |
| **12** | | | | | |
| MSG | 1 | SEVIRI | 12,1,21 | 1 to 8 | rtcoef_msg_1_seviri.dat |
| | | | | | |
| **13** | | | | | |
| FY1 | 3 | MVISR | 13,3,26 | 1 to 3 | rt_coef_fy1_3_mvisr.dat |
| FY1 | 4 | MVISR | 13,4,26 | 1 to 3 | rt_coef_fy1_4_mvisr.dat |

## ANNEX-C: Example of RT coefficient file header

```
! RTTOV coefficient file noaa-15   hirs
! ------------------------------------------------------
IDENTIFICATION
!
 1 15  0          ! platform sat_id instrument
 noaa-15   hirs
 ir               ! sensor type [ir,mw,hi]
  7               ! RTTOV compatibility version
 copy from original M. Matricardi coefficient file
 2001 08 28       ! creation date
! ------------------------------------------------------
LINE-BY-LINE
!
!Aug00 R.Saunders: Better ToA GENLN2 convolution; P.Rayer: Liebe trans averaging
 GENLN2           ! line-by-line
 HITRAN96         ! spectroscopic database
 CKD2.1           ! Water Vapour continuum
 2                ! Profile datasets
 TIGR-43          ! dataset name
 43 2 43 6        ! profiles gases levels secants
 NESDIS-34        ! dataset name
 34 2 43 6        ! profiles gases levels secants
! ------------------------------------------------------
FAST_MODEL_VARIABLES
!
!Predictors MG EYRE; WV RAYER+SAUNDERS+DEBLONDE; Ozone RAYER
 RTTOV6           ! fast model name
   19             ! Number of channels described in the coef file
    3             ! Number of gases described in the coef file
 Mixed_gases      ! gas identification
   10  43         ! variables/predictors  levels (pressure/absorber)
 Water_vapour     ! gas identification
   15  43         ! variables/predictors  levels (pressure/absorber)
 Ozone            ! gas identification
   11  43         ! variables/predictors  levels (pressure/absorber)
! ------------------------------------------------------
FILTER_FUNCTIONS
!
! Channel Number (from instrument original description)
! Channel status
! Central Wavenumber
! Band Correction coefficients(Offset,Slope)
! Gamma correction factor
    1    1  0.6691342163E+03  0.1917844405E-02  0.9999905229E+00  0.1000000000E+01
    2    1  0.6787655640E+03  0.9768215939E-02  0.9999527335E+00  0.1000000000E+01
    3    1  0.6904246216E+03  0.1596293226E-01  0.9999244809E+00  0.1000000000E+01
    4    1  0.7031503906E+03  0.1537587214E-01  0.9999291301E+00  0.1000000000E+01
    5    1  0.7159310303E+03  0.2067222074E-01  0.9999068379E+00  0.1000000000E+01
    6    1  0.7317467041E+03  0.1777464710E-01  0.9999219775E+00  0.1000000000E+01
    7    1  0.7476658936E+03  0.2366036363E-01  0.9998987317E+00  0.1000000000E+01
    8    1  0.8973583984E+03  0.8957295120E-01  0.9996817708E+00  0.1000000000E+01
    9    1  0.1032136475E+04  0.4540707543E-01  0.9998575449E+00  0.1000000000E+01
   10    1  0.8011007080E+03  0.1491035428E-01  0.9999408126E+00  0.1000000000E+01
   11    1  0.1362443115E+04  0.7500830293E-01  0.9998134971E+00  0.1000000000E+01
   12    1  0.1529819458E+04  0.1042257622E+00  0.9997656941E+00  0.1000000000E+01
   13    1  0.2188191406E+04  0.2469877526E-01  0.9999593496E+00  0.1000000000E+01
   14    1  0.2209933105E+04  0.1916682161E-01  0.9999685884E+00  0.1000000000E+01
   15    1  0.2235279541E+04  0.1926082000E-01  0.9999687076E+00  0.1000000000E+01
   16    1  0.2241990234E+04  0.2268156968E-01  0.9999632239E+00  0.1000000000E+01
   17    1  0.2418979248E+04  0.3350611404E-01  0.9999493957E+00  0.1000000000E+01
   18    1  0.2518802002E+04  0.4703077674E-01  0.9999316931E+00  0.1000000000E+01
   19    1  0.2657263428E+04  0.2995664477E+00  0.9995844364E+00  0.1000000000E+01
! ------------------------------------------------------
FUNDAMENTAL_CONSTANTS
!
! units of constants for spectral radiance
! first radiation constant(mW/(m2.sr.cm-4))
! second radiation constant (cm.K)
  29979245800.0            ! speed of light (cm/s)
  1.19106590E-05  1.438833  ! Planck constants
  833.0                    ! nominal satellite height (km)
```

```
 ! -------------------------------------------------------
SSIREM
 !
 ! Channel Number (from instrument original description)
 ! 5 coefficients for emissivity model ssirem
  1   ! version number
   1  0.9569390  0.0106855  0.0553136 3.0 6.0
   2  0.9584720  0.0103879  0.0554133 3.0 6.0
   3  0.9606930  0.0099114  0.0555347 3.0 6.0
   4  0.9629580  0.0093700  0.0556006 3.0 6.0
   5  0.9653470  0.0087125  0.0555577 3.0 6.0
   6  0.9686160  0.0076444  0.0552058 3.0 6.0
   7  0.9720030  0.0063550  0.0544589 3.0 6.0
   8  0.9923630  0.0075718  0.0189907 4.0 8.0
   9  0.9878380  0.0110179  0.0206534 4.0 8.0
  10  0.9831310  0.0201702  0.0293839 4.0 8.0
  11  0.9814260  0.0154611  0.0229799 4.0 8.0
  12  0.9782320  0.0173583  0.0237055 4.0 8.0
  13  0.9791930  0.0167557  0.0234646 4.0 8.0
  14  0.9791010  0.0168076  0.0234835 4.0 8.0
  15  0.9789980  0.0168656  0.0235033 4.0 8.0
  16  0.9789720  0.0168803  0.0235078 4.0 8.0
  17  0.9777260  0.0175715  0.0237401 4.0 8.0
  18  0.9768450  0.0180460  0.0238885 4.0 8.0
  19  0.9752210  0.0188794  0.0241300 4.0 8.0
 ! -------------------------------------------------------
REFERENCE_PROFILE
 !
 ! Ref.pressure (hPa)
 ! Ref.Temp (K) Ref.Mixing Ratio [Kg/Kg] for each gas
 ! Note for MxG that mixing ratio is "missing"
 !     Mixed_gases
   0.100  232.736 -0.999900E+04
   0.290  247.984 -0.999900E+04
   0.690  256.373 -0.999900E+04
   1.420  254.918 -0.999900E+04
   2.610  250.632 -0.999900E+04
   4.410  242.916 -0.999900E+04
```

…………

*etc. reference profiles and profile limits followed by RT coefficients themselves*

## ANNEX-D: RTTVI subroutine interface

**CALL RTTVI** ( KERR, KPPF, KPNSAT, KPLEV, KPCH, KPCHUS, KPNAV, KPNSAV, KPNSSV, KPNCV, NRTTOVID, PLATFORM, SATELLITE, INSTRUMENT, NUMCHANS, PRESLEV, OTMIN, OTMAX, OQMIN, OQMAX, OOZMIN, OOZMAX, IVCH, NIU1)

RTTVI is called only once for all platforms, satellites and instruments. The table below lists the variables and gives an example of what the arrays should contain to set up RTTOV for simulating NOAA-16 AMSU-A, AMSU-B and METEOSAT-7 MVIRI radiances for up to 6 profiles in each call of RTTOV. The jpxxx array sizes are set up by the module *MOD_CPARAM.f90*. For this example setting jpnsat=3, jpch=15, jppf=6 and nlev=43 is optimum to allow all the calling options given in Annexes E-H to work. UM Tables refer to tables in the User Manual.

| Parameter and size if > 1 | Type | IN/OUT | Description | Example of contents |
|---|---|---|---|---|
| KERR | INTEGER | OUT | Error if not 0 | 0 |
| KPPF | INTEGER | OUT | No. of profiles | 6 |
| KPNSAT | INTEGER | OUT | No. of sensors | 3 |
| KPLEV | INTEGER | OUT | No. of levels | 43 |
| KPCH | INTEGER | OUT | No. of channels | 15 |
| KPCHUS | INTEGER | OUT | No. of channels | 15 |
| KPNAV | INTEGER | OUT | No. of profile vars | 4 |
| KPNSAV | INTEGER | OUT | No. of 2m surface vars | 5 |
| KPNSSV | INTEGER | OUT | No. of surface skin vars | 6 |
| KPNCV | INTEGER | OUT | No. of cloud variables | 2 |
| NRTTOVID | INTEGER | IN | No. of sensors | 3 |
| PLATFORM(jpnsat) | INTEGER | IN | Platform ids (UM Table 3) | 1,1,3 |
| SATELLITE(jpnsat) | INTEGER | IN | Satellite ids (UM Table 3) | 16,16,7 |
| INSTRUMENTjpnsat) | INTEGER | IN | Instrument ids (UM Table 3) | 3,4,20 |
| NUMCHANS(jpnsat) | INTEGER | IN/OUT | Number of channels | 15,5,2 |
| PRESLEV(nlev) | REAL | OUT | Pressure levels for coeffs | UM Table 2 |
| OTMIN(nlev) | REAL | OUT | Min valid Temp | UM Table 2 |
| OTMAX(nlev) | REAL | OUT | Max valid Temp | UM Table 2 |
| OQMIN(nlev) | REAL | OUT | Min valid specific hum | UM Table 2 |
| OQMAX(nlev) | REAL | OUT | Max valid specific hum | UM Table 2 |
| OOZMIN(nlev) | REAL | OUT | Min valid ozone | UM Table 2 |
| OOZMAX(nlev) | REAL | OUT | Max valid ozone | UM Table 2 |
| IVCH(jpch,jpnsat) | INTEGER | IN/OUT | See note 1 below | 1-15,1-5,1-2 |
| NIU1(jpnsat) | INTEGER | IN/OUT | See note 2 below | 10 |

Note 1. Normally IVCH on input should be initialised to zero and the output will contain all valid channel numbers for each sensor. This can be used to check RTTOV is not called with an invalid channel number. For sensors with large number of channels (e.g. AIRS) the IVCH array and NUMCHANS array can be set to non-zero values on input to allow coefficients for *only* those channels required to be read in to memory. In this case the number of channels required is in NUMCHANS and their numbers are given in the IVCH array. If IVCH is zero on input coefficients for all valid channels are read into memory.

Note 2. NIU1 is an optional parameter and if set to a non-zero value defines the fortran unit numbers through which the coefficient files are read in. This file should have already been opened.

## ANNEX-E: RTTOV subroutine interface

**CALL RTTOV** (KNPF, KLENPF, PPRES, PANGL, PANGS, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCV, PEMIS, IFAIL, PRAD, PTB, RADOV, RADO, TAU, TAUSFC, LCLOUD)

RTTOV is called for every sensor required for KNPF profiles at a time. The table below lists the variables and gives an example of what the arrays should contain for RTTOV to simulate NOAA-16 AMSU-B for 3 profiles and 4 out of the 5 channels (omitting channel 2). This assumes the calling sequence in Annex D is followed. UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for AMSU-B |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 3 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| PPRES(jplev) | REAL | IN | Pressure levels (hPa) | UM Table 2 |
| PANGL(knpf) | REAL | IN | Sat zenith angle (deg) | 30.,32.,34. |
| PANGS(knpf) | REAL | Not used | Solar zenith angle (deg) | 0.,0.,0. |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1,1,0 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI (see annex D) | 2 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 4*3=12 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers[+] | 1,3,4,5,1,3,. |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,1,2,2,2.. |
| PAV(jplev,4,knpf) | REAL | IN | Profile array (UM Table 1) | 3 profiles |
| PSAV(5,knpf) | REAL | IN | Surface 2m array (UM Table 1) | 3 profiles |
| PSSV(6,knpf) | REAL | IN | Surface skin array (UM Table 1) | 3 profiles |
| PCV(2,knpf) | REAL | IN | Cloud array (UM Table 1) | 3 profiles |
| PEMIS(knchpf) | REAL | IN/OUT | Surface emissivity (UM Table 4) | 3 profiles |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 3*3*0 |
| PRAD(knchpf) | REAL | OUT | Radiances in $mW/m^2/sr/cm^{-1}$ | 12 radiances |
| PTB(knchpf) | REAL | OUT | Brightness temps in degK | 12 Br. Temps |
| RADOV(knchpf,2*jplev+2) | REAL | OUT | Overcast cloudy radiances[¶] | 1056 rads |
| RADO(knchpf) | REAL | OUT | O'cast radiance from cld top | 12 radiances |
| TAU(knchpf,jplev) | REAL | OUT | Layer to space transmittances | 43*12 trans |
| TAUSFC(knchpf) | REAL | OUT | Surface to space transmittances | 12 trans |
| LCLOUD | LOGICAL | IN | Switch for IR cloud calcs | .false. |

[+] If the array IVCH is non-zero on input to RTTVI then this channel index refers to the subset of channels requested in IVCH (normally only used for AIRS).

[¶] The RADOV array contains the following radiances for possible cloud computations outside RTTOV (e.g. used by RTTOVCLD):

RADOV (knchpf,1:njplev) : overcast radiances at given cloud top
RADOV (knchpf,njplev+1,2*njplev) : contribution to radiance of downward cloud emission at given cloud top
RADOV (knchpf,2*njplev+1) : clear-sky radiance without reflection term
RADOV (knchpf,2*njplev+2) : reflected clear-sky downwelling radiance

## ANNEX-F: RTTOVK subroutine interface

**CALL RTTOVK**(KNPF, KLENPF, PPRES, PANGL, PANGS, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCV, PEMIS, PAV_D, PSAV_D, PSSV_D, PCV_D, PEMIS_D, PRAD_D, PTB_D, KINRAD, LCLOUD,IFAIL,RADOV)

RTTOVK is called once for each sensor for KNPF profiles at a time. The table below lists the variables and gives an example of what the arrays should contain for RTTOVK to simulate METEOSAT MVIRI for 1 profile and both channels for a zenith angle of 30 deg. This assumes the calling sequence to RTTVI in Annex D is followed. The variables ending in _D denote direct value (same as RTTOV input/output). UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for MVIRI |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 1 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| PPRES(jplev) | REAL | IN | Pressure levels (hPa) | UM Table 2 |
| PANGL(knpf) | REAL | IN | Sat zenith angle (deg) | 30. |
| PANGS(knpf) | REAL | Not used | Solar zenith angle (deg) | 0. |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI (see annex D) | 3 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 2*1=2 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers [+] | 1,2 |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1 |
| PAV(jplev,4,knpf) | REAL | OUT | K of profile array | 1 profile |
| PSAV(5,knpf) | REAL | OUT | K of surface 2m array | 1 profile |
| PSSV(6,knpf) | REAL | OUT | K of Surface skin array | 1 profile |
| PCV(2,knpf) | REAL | OUT | K of cloud array | 1 profile |
| PEMIS(knchpf) | REAL | IN/OUT | K of surface emiss (UM table 4) | 1 profile |
| PAV_D(jplev,4,knpf) | REAL | IN | Input profile array | 1 profile |
| PSAV_D(5,knpf) | REAL | IN | Input surface 2m array | 1 profile |
| PSSV_D(6,knpf) | REAL | IN | Input surface skin array | 1 profile |
| PCV_D(2,knpf) | REAL | IN | In put cloud array | 1 profile |
| PEMIS_D(knchpf) | REAL | IN/OUT | Input surface emiss (UM Table 4) | 1 profile |
| PRAD_D(knchpf) | REAL | OUT | Radiances in $mW/m^2/sr/cm^{-1}$ | 2 radiances |
| PTB_D(knchpf) | REAL | OUT | Brightness temps in degK | 2 Br. Temps |
| KINRAD | INTEGER | IN | Switch (1=radiance, 2=BT) | 2 |
| LCLOUD | LOGICAL | IN | Switch for IR cloud calcs | .false. |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 3*3*0 |
| RADOV(knchpf,2*jplev+2) | REAL | IN/OUT | K of overcast cloudy radiances[¶] | radiances |

[+] If the array IVCH is non-zero on input to RTTVI then this channel index refers to the subset of channels requested in IVCH (normally only used for AIRS).

[¶] The RADOV array contains the following radiances for possible cloud computations outside RTTOV (e.g. used by RTTOVCLD):

- RADOV (knchpf,1:njplev) : overcast radiances at given cloud top
- RADOV (knchpf,njplev+1,2*njplev) : contribution to radiance of downward cloud emission at given cloud top
- RADOV (knchpf,2*njplev+1) : clear-sky radiance without reflection term
- RADOV (knchpf,2*njplev+2) : reflected clear-sky downwelling radiance

## ANNEX-G: RTTOVTL subroutine interface

**CALL RTTOVTL** (KNPF, KLENPF, PPRES, PANGL, PANGS, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCV, PEMIS, PAV_D, PSAV_D, PSSV_D, PCV_D, PEMIS_D, RADOV_D, PRAD, PTB, RADOV, LCLOUD, IFAIL)

RTTOVTL is called once for each sensor for KNPF profiles at a time. The table below lists the variables and gives an example of what the arrays should contain for RTTOV to simulate NOAA-16 AMSU-A for 1 profile and all 15 channels for a zenith angle of 15 deg. This assumes the calling sequence to RTTVI in Annex D is followed. The variables ending in _D denote direct value (same as RTTOV input/output). UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for AMSU-A |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 1 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| PPRES(jplev) | REAL | IN | Pressure levels (hPa) | UM Table 2 |
| PANGL(knpf) | REAL | IN | Sat zenith angle (deg) | 15. |
| PANGS(knpf) | REAL | Not used | Solar zenith angle (deg) | 0. |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI (see annex D) | 1 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 15*1=15 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers + | 1,2,3,..15 |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,…1,1 |
| PAV(jplev,4,knpf) | REAL | IN | TL of profile array | 1 profile |
| PSAV(5,knpf) | REAL | IN | TL of surface 2m array | 1 profile |
| PSSV(6,knpf) | REAL | IN | TL of surface skin array | 1 profile |
| PCV(2,knpf) | REAL | IN | TL of cloud array | 1 profile |
| PEMIS(knchpf) | REAL | IN/OUT | TL of surface emiss (UM table 4) | 1 profile |
| PAV_D(jplev,4,knpf) | REAL | IN | Input profile array | 1 profile |
| PSAV_D(5,knpf) | REAL | IN | Input surface 2m array | 1 profile |
| PSSV_D(6,knpf) | REAL | IN | Input surface skin array | 1 profile |
| PCV_D(2,knpf) | REAL | IN | In put cloud array | 1 profile |
| PEMIS_D(knchpf) | REAL | IN/OUT | Input surface emiss (UM table 4) | 1 profile |
| RADOV_D(knchpf,2*jplev+2) | REAL | OUT | Overcast cloudy radiances¶ | Radiances |
| PRAD(knchpf) | REAL | OUT | TL of radiances in $mW/m^2/sr/cm^{-1}$ | 15 radiances |
| PTB(knchpf) | REAL | OUT | TL of Brightness temps in degK | 15 B. Temps |
| RADOV(knchpf,2*jplev+2) | REAL | OUT | TL of overcast cloudy radiances¶ | Radiances |
| LCLOUD | LOGICAL | IN | Switch for IR cloud calcs | .false. |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 3*3*0 |

+ If the array IVCH is non-zero on input to RTTVI then this channel index refers to the subset of channels requested in IVCH (normally only used for AIRS).

¶ The RADOV array contains the following TL radiances for possible cloud computations outside RTTOV (e.g. used by RTTOVCLD):

| | |
|---|---|
| RADOV (knchpf,1:njplev) | : TL overcast radiances at given cloud top |
| RADOV (knchpf,njplev+1,2*njplev) | : TL contribution to radiance of downward cloud emission at given cloud top |
| RADOV (knchpf,2*njplev+1) | : TL clear-sky radiance without reflection term |
| RADOV (knchpf,2*njplev+2) | : TL reflected clear-sky downwelling radiance |

## ANNEX-H: RTTOVAD subroutine interface

**CALL RTTOVAD** (KNPF, KLENPF, PPRES, PANGL, PANGS, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCV, PEMIS, PAV_D, PSAV_D, PSSV_D, PCV_D, PEMIS_D, PRAD, PTB, RADOV, KINRAD, LCLOUD, IFAIL)

RTTOVAD is called once for each sensor for KNPF profiles at a time. The table below lists the variables and gives an example of what the arrays should contain for RTTOV to simulate NOAA-16 AMSU-A for 6 profiles and 2 channels (chans 3, 8) for a zenith angle of 25 deg. This assumes the calling sequence to RTTVI in Annex D is followed. The variables ending in _D denote direct value (same as RTTOV input/output). UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for AMSU-A |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 6 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| PPRES(jplev) | REAL | IN | Pressure levels (hPa) | UM Table 2 |
| PANGL(knpf) | REAL | IN | Sat zenith angle (deg) | 6*25. |
| PANGS(knpf) | REAL | Not used | Solar zenith angle (deg) | 6*0. |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 6*1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI (see annex D) | 1 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 2*6=12 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers + | 3,8,3,8,.. |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,2,2,3,3.. |
| PAV(jplev,4,knpf) | REAL | OUT | AD of profile array | 6 profiles |
| PSAV(5,knpf) | REAL | OUT | AD of surface 2m array | 6 profiles |
| PSSV(6,knpf) | REAL | OUT | AD of surface skin array | 6 profiles |
| PCV(2,knpf) | REAL | OUT | AD of cloud array | 6 profiles |
| PEMIS(knchpf) | REAL | IN/OUT | AD of surface emiss (UM table 4) | 6 profiles |
| PAV_D(jplev,4,knpf) | REAL | IN | Input profile array | 6 profiles |
| PSAV_D(5,knpf) | REAL | IN | Input surface 2m array | 6 profiles |
| PSSV_D(6,knpf) | REAL | IN | Input surface skin array | 6 profiles |
| PCV_D(2,knpf) | REAL | IN | In put cloud array | 6 profiles |
| PEMIS_D(knchpf) | REAL | IN/OUT | Input surface emiss (UM table 4) | 6 profiles |
| PRAD(knchpf) | REAL | IN | AD of radiances in $mW/m^2/sr/cm^{-1}$ | Ignored for kinrad=2 |
| PTB(knchpf) | REAL | IN | AD of Brightness temps in degK | 12 B. Temps |
| RADOV(knchpf,2*jplev+2) | REAL | IN/OUT | AD of overcast cloudy radiances¶ | Radiances |
| KINRAD | INTEGER | IN | Switch (1=radiance, 2=BT) | 2 |
| LCLOUD | LOGICAL | IN | Switch for IR cloud calcs | .false. |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See Table 6 | 3*3*0 |

+ If the array IVCH is non-zero on input to RTTVI then this channel index refers to the subset of channels requested in IVCH (normally only used for AIRS).

¶ The RADOV array contains the following AD radiances for possible cloud computations outside RTTOV (e.g. used by RTTOVCLD):

RADOV (knchpf,1:njplev) : AD overcast radiances at given cloud top
RADOV (knchpf,njplev+1,2*njplev) : AD contribution to radiance of downward cloud emission at given cloud top
RADOV (knchpf,2*njplev+1) : AD clear-sky radiance without reflection term
RADOV (knchpf,2*njplev+2) : AD reflected clear-sky down-welling radiance

## ANNEX-I: RTTOVCLD subroutine interface

**CALL RTTOVCLD (**KNPF, KLENPF, KLEVM, PPRES, PANGL, PANGS, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCVM, PAP, PAPH, PEMIS, IFAIL, PRAD, PTB, PRADCLD, PTBCLD, TAU, TAUSFC, PRADOVM, PCLDEMIS, PAIT, PAIS**)**

RTTOVCLD is called for every sensor required for KNPF profiles at a time. The Table below lists the variables and gives an example of what the arrays should contain for RTTOVCLD to simulate NOAA-15 HIRS for 1 profile and 3 out of the 20 channels (chans 6,7,8). This assumes NOAA-15 HIRS is loaded by RTTVI. UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for HIRS |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 1 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| KLEVM | INTEGER | IN | No. of native model levels | 60 |
| PPRES(jplev) | REAL | IN | RTTOV pressure level (hPa) | UM table 2 |
| PANGL(knpf) | REAL | IN | Sat zenith angle (deg) | 30. |
| PANGS(knpf) | REAL | Not used | Solar zenith angle (deg) | 0.,0.,0. |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI | 1 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 3*1=3 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers | 6,7,8 |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,1 |
| PAV(jplev,4,knpf) | REAL | IN | Profile array (UM Table 1) | 1 profile |
| PSAV(5,knpf) | REAL | IN | Surface 2m array (UM Table 1) | 1 profile |
| PSSV(6,knpf) | REAL | IN | Surface skin array (UM Table 1) | 1 profile |
| PCVM(knpf,klevm,4) | REAL | IN | Cloud array # | 1 profile |
| PAP(knpf,klevm) | REAL | IN | Full user model levels | 1 profile |
| PAPH(knpf,klevm+1) | REAL | IN | Half user model levels | 1 profile |
| PEMIS(knchpf) | REAL | IN/OUT | Surface emissivity (UM Table 4) | 1 profile |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 0 |
| PRAD(knchpf) | REAL | OUT | Radiances in $mW/m^2/sr/cm^{-1}$ | 3 radiances |
| PTB(knchpf) | REAL | OUT | Brightness temps in degK | 3 Br. Temps |
| PRADCLD(knchpf) | REAL | OUT | Cloud affected radiance | 3 radiances |
| PTBCLD(knchpf) | REAL | OUT | Cloud affected br. Temp | 3 Br. Temps |
| TAU(knchpf,jplev) | REAL | OUT | Layer to space transmittances | 43*3 trans |
| TAUSFC(knchpf) | REAL | OUT | Surface to space transmittances | 3 trans |
| PRADOVM(knchpf,2*klevm+2) | REAL | OUT | Overcast cloudy radiances¶ | 366 rads |
| PCLDEMIS(knchpf,klevm) | REAL | OUT | Cloud emissivity | 180 emis |
| PAIT(knchpf,klevm+1) | REAL | OUT | TOA weights to cloud layers | 183 wts |
| PAIS(knchpf,klevm+1) | REAL | OUT | Surface weights of the cloud layers | 183 wts |

¶ The PRADOVM array contains the following radiances on user levels :

PRADOVM (knchpf,1:klevm) : overcast radiances at given cloud top
PRADOVM (knchpf,klevm+1,2*klevm) : contribution to radiance of downward cloud emission at given cloud top
PRADOVM (knchpf,2*klevm+1) : clear-sky radiance without reflection term
PRADOVM (knchpf,2*klevm+2) : reflected clear-sky downwelling radiance

# PCVM contains temperature and cloud variables on klevm layers: 1=temperature (K), 2=cloud cover, 3=cloud liquid water (kg/kg), 4=cloud ice water (kg/kg)

## ANNEX-J: RTTOVCLDTL subroutine interface

**CALL RTTOVCLDTL (**KNPF, KLENPF, KLEVM, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCVM, PAP, PAPH, PEMIS, PRAD, PTB, PRADCLD, PTBCLD, PPRES_D, PANGL_D, PANGS_D, PAV_D, PSAV_D, PSSV_D, PCVM_D, PAP_D, PAPH_D, PEMIS_D, IFAIL**)**

RTTOVCLDTL is called for every sensor required for KNPF profiles at a time. The Table below lists the variables and gives an example of what the arrays should contain for RTTOVCLDTL to simulate NOAA-15 HIRS for 1 profile and 3 out of the 20 channels (chans 6,7,8). This assumes NOAA-15 HIRS is loaded by RTTVI. UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for HIRS |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 1 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| KLEVM | INTEGER | IN | No. of native model levels | 60 |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI | 1 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 3*1=3 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers | 6,7,8 |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,1 |
| PAV(jplev,4,knpf) | REAL | IN | TL of Profile array | 1 profile |
| PSAV(5,knpf) | REAL | IN | TL of surface 2m array | 1 profile |
| PSSV(6,knpf) | REAL | IN | TL of surface skin array | 1 profile |
| PCVM(knpf,klevm,4) | REAL | IN | TL of cloud array # | 1 profile |
| PAP(knpf,klevm) | REAL | IN | TL of full user model levels | 1 profile |
| PAPH(knpf,klevm+1) | REAL | IN | TL of half user model levels | 1 profile |
| PEMIS(knchpf) | REAL | IN/OUT | TL of surface emissivity | 1 profile |
| PRAD(knchpf) | REAL | OUT | TL of Radiances | 3 radiances |
| PTB(knchpf) | REAL | OUT | TL of Br. temps in degK | 3 Br. Temps |
| PRADCLD(knchpf) | REAL | OUT | TL of cloud affected radiance | 3 radiances |
| PTBCLD(knchpf) | REAL | OUT | TL of cloud affected TBs | 3 Br. Temps |
| PPRES_D(jplev) | REAL | IN | RTTOV pressure level (hPa) | UM table 2 |
| PANGL_D(knpf) | REAL | IN | Sat zenith angle (deg) | 30. |
| PANGS_D(knpf) | REAL | Not used | Solar zenith angle (deg) | 0.,0.,0. |
| PAV_D(jplev,4,knpf) | REAL | IN | Profile array (UM Table 1) | 1 profile |
| PSAV_D(5,knpf) | REAL | IN | Surface 2m array (UM Table 1) | 1 profile |
| PSSV_D(6,knpf) | REAL | IN | Surface skin array (UM Table 1) | 1 profile |
| PCVM_D(knpf,klevm,4) | REAL | IN | Cloud array# | 1 profile |
| PAP_D(knpf,klevm) | REAL | IN | Full user model levels | 1 profile |
| PAPH_D(knpf,klevm+1) | REAL | IN | Half user model levels | 1 profile |
| PEMIS_D(knchpf) | REAL | IN/OUT | Surface emissivity (UM Table 4) | 1 profile |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 0 |

# PCVM contains temperature and cloud variables on klevm layers: 1=temperature (K), 2=cloud cover, 3=cloud liquid water (kg/kg), 4=cloud ice water (kg/kg)

## ANNEX-K: RTTOVCLDAD subroutine interface

**CALL RTTOVCLDAD (**KNPF, KLENPF, KLEVM, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCVM, PAP, PAPH, PEMIS, PRAD, PTB, PRADCLD, PTBCLD, PPRES_D, PANGL_D, PANGS_D, PAV_D, PSAV_D, PSSV_D, PCVM_D, PAP_D, PAPH_D, PEMIS_D, KINRAD, IFAIL**)**

RTTOVCLDAD is called for every sensor required for KNPF profiles at a time. The Table below lists the variables and gives an example of what the arrays should contain for RTTOVCLDAD to simulate NOAA-15 HIRS for 1 profile and 3 out of the 20 channels (chans 6,7,8). This assumes NOAA-15 HIRS is loaded by RTTVI. UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for HIRS |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 1 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| KLEVM | INTEGER | IN | No. of native model levels | 60 |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI | 1 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 3*1=3 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers | 6,7,8 |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,1 |
| PAV(jplev,4,knpf) | REAL | OUT | AD of Profile array | 1 profile |
| PSAV(5,knpf) | REAL | OUT | AD of surface 2m array | 1 profile |
| PSSV(6,knpf) | REAL | OUT | AD of surface skin array | 1 profile |
| PCVM(knpf,klevm,4) | REAL | IN/OUT | AD of cloud array # | 1 profile |
| PAP(knpf,klevm) | REAL | IN/OUT | AD of full user model levels | 1 profile |
| PAPH(knpf,klevm+1) | REAL | IN/OUT | AD of half user model levels | 1 profile |
| PEMIS(knchpf) | REAL | IN/OUT | AD of surface emissivity | 1 profile |
| PRAD(knchpf) | REAL | IN/OUT | AD of Radiances | 3 radiances |
| PTB(knchpf) | REAL | IN/OUT | AD of Br. temps in degK | 3 Br. Temps |
| PRADCLD(knchpf) | REAL | IN/OUT | AD of cloud affected radiance | 3 radiances |
| PTBCLD(knchpf) | REAL | IN/OUT | AD of cloud affected TBs | 3 Br. Temps |
| PPRES_D(jplev) | REAL | IN | RTTOV pressure level (hPa) | UM table 2 |
| PANGL_D(knpf) | REAL | IN | Sat zenith angle (deg) | 30. |
| PANGS_D(knpf) | REAL | Not used | Solar zenith angle (deg) | 0.,0.,0. |
| PAV_D(jplev,4,knpf) | REAL | IN | Profile array (UM Table 1) | 1 profile |
| PSAV_D(5,knpf) | REAL | IN | Surface 2m array (UM Table 1) | 1 profile |
| PSSV_D(6,knpf) | REAL | IN | Surface skin array (UM Table 1) | 1 profile |
| PCVM_D(knpf,klevm,4) | REAL | IN | Cloud array# | 1 profile |
| PAP_D(knpf,klevm) | REAL | IN | Full user model levels | 1 profile |
| PAPH_D(knpf,klevm+1) | REAL | IN | Half user model levels | 1 profile |
| PEMIS_D(knchpf) | REAL | IN/OUT | Surface emissivity (UM Table 4) | 1 profile |
| KINRAD | INTEGER | IN | Switch (1=radiance, 2=BT) | 2 |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 0 |

# PCVM contains temperature and cloud variables on klevm layers: 1=temperature (K), 2=cloud cover, 3=cloud liquid water (kg/kg), 4=cloud ice water (kg/kg)

## ANNEX-L: RTTOVCLDK subroutine interface

**CALL RTTOVCLDK (**KNPF, KLENPF, KLEVM, KSURF, KSAT, KNCHPF, KCHAN, KPROF, PAV, PSAV, PSSV, PCVM, PAP, PAPH, PEMIS, PRAD, PTB, PRADCLD, PTBCLD, PPRES_D, PANGL_D, PANGS_D, PAV_D, PSAV_D, PSSV_D, PCVM_D, PAP_D, PAPH_D, PEMIS_D, KINRAD, IFAIL**)**

RTTOVCLDK is called for every sensor required for KNPF profiles at a time. The Table below lists the variables and gives an example of what the arrays should contain for RTTOVCLDK to simulate NOAA-15 HIRS for 1 profile and 3 out of the 20 channels (chans 6,7,8). This assumes NOAA-15 HIRS is loaded by RTTVI. UM Tables refer to tables in the User Manual.

| Parameter and size if >1 | Type | IN/OUT | Description | Example of contents for HIRS |
|---|---|---|---|---|
| KNPF | INTEGER | IN | No. of profiles | 1 |
| KLENPF | INTEGER | IN | No. of levels in profiles | 43 |
| KLEVM | INTEGER | IN | No. of native model levels | 60 |
| KSURF(knpf) | INTEGER | IN | 0=land, 1=sea, 2=sea-ice | 1 |
| KSAT | INTEGER | IN | Sequence number as loaded by RTTVI | 1 |
| KNCHPF | INTEGER | IN | No chans*No. profiles | 3*1=3 |
| KCHAN(knchpf) | INTEGER | IN | Channel numbers | 6,7,8 |
| KPROF(knchpf) | INTEGER | IN | Profile numbers | 1,1,1 |
| PAV(jplev,4,knchpf) | REAL | OUT | K of Profile array | 1 profile |
| PSAV(5,knchpf) | REAL | OUT | K of surface 2m array | 1 profile |
| PSSV(6,knchpf) | REAL | OUT | K of surface skin array | 1 profile |
| PCVM(knpf,klevm,4) | REAL | IN/OUT | K of cloud array# | 1 profile |
| PAP(knpf,klevm) | REAL | IN/OUT | K of full user model levels | 1 profile |
| PAPH(knpf,klevm+1) | REAL | IN/OUT | K of half user model levels | 1 profile |
| PEMIS(knchpf) | REAL | IN/OUT | K of surface emissivity | 1 profile |
| PRAD(knchpf) | REAL | IN/OUT | K of Radiances | 3 radiances |
| PTB(knchpf) | REAL | IN/OUT | K of Br. temps in degK | 3 Br. Temps |
| PRADCLD(knchpf) | REAL | IN/OUT | K of cloud affected radiance | 3 radiances |
| PTBCLD(knchpf) | REAL | IN/OUT | K of cloud affected TBs | 3 Br. Temps |
| PPRES_D(jplev) | REAL | IN | RTTOV pressure level (hPa) | UM table 2 |
| PANGL_D(knpf) | REAL | IN | Sat zenith angle (deg) | 30. |
| PANGS_D(knpf) | REAL | Not used | Solar zenith angle (deg) | 0.,0.,0. |
| PAV_D(jplev,4,knchpf) | REAL | IN | Profile array (UM Table 1) | 1 profile |
| PSAV_D(5,knchpf) | REAL | IN | Surface 2m array (UM Table 1) | 1 profile |
| PSSV_D(6,knchpf) | REAL | IN | Surface skin array (UM Table 1) | 1 profile |
| PCVM_D(knpf,klevm,4) | REAL | IN | Cloud array# | 1 profile |
| PAP_D(knpf,klevm) | REAL | IN | Full user model levels | 1 profile |
| PAPH_D(knpf,klevm+1) | REAL | IN | Half user model levels | 1 profile |
| PEMIS_D(knchpf) | REAL | IN/OUT | Surface emissivity (UM Table 4) | 1 profile |
| KINRAD | INTEGER | IN | Switch (1=radiance, 2=BT) | 2 |
| IFAIL(knpf,knpsat) | INTEGER | OUT | See UM Table 6 | 0 |

[#] PCVM contains temperature and cloud variables on klevm layers: 1=temperature (K), 2=cloud cover, 3=cloud liquid water (kg/kg), 4=cloud ice water (kg/kg)

## ANNEX-M: RTTOV-7 Fortran 90 parameters

| Module | Variable | Profile | Type | Direct | Tangent Linear | Adjoint | K |
|---|---|---|---|---|---|---|---|
| MOD_CPARAM.f90 | jppf | xxx | I | DEBYE.f90 EMISS.f90 PRFIN.f90 PRFTAU.f90 RTINT.f90 RTTAU.f90 RTTOV.f90 RTTOVK_TEST.f90 RTTVI.f90 SUCOMTOVS.f90 tstrad.f90 | DEBYETL.f90 EMISSTL.f90 MOD_GRODYTL.f90 MOD_IRCLDTL.f90 MOD_PRFVARTL.f90 MOD_SURFTL.f90 PRFINTL.f90 PRFTAUTL.f90 RTINTTL.f90 RTTAUTL.f90 RTTOVTL.f90 TSTRADTL.f90 | DEBYEAD.f90 EMISSAD.f90 PRFINAD.f90 PRFTAUAD.f90 RTINTAD.f90 RTTAUAD.f90 RTTOVAD.f90 TSTRADAD.f90 | DEBYEK.f90 EMISSK.f90 MOD_PRFVARK.f90 PRFINK.f90 PRFTAUK.f90 RTINTK.f90 RTTAUK.f90 RTTOVK.f90 TSTRADK.f90 |
| MOD_CPARAM.f90 | jpch | xxx | I | EMISS.f90 PRFIN.f90 RTINT.f90 RTTOV.f90 RTTOVCF.f90 RTTOVK_TEST.f90 RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | EMISSTL.f90 MOD_PRFVARTL.f90 PRFINTL.f90 RTINTTL.f90 RTTOVTL.f90 TSTRADTL.f90 | EMISSAD.f90 PRFINAD.f90 RTINTAD.f90 RTTOVAD.f90 TSTRADAD.f90 | DEBYEK.f90 EMISSK.f90 MOD_GRODYK.f90 MOD_IRCLDK.f90 MOD_PRFVARK.f90 MOD_SURFK.f90 PRFINK.f90 PRFTAUK.f90 RTINTK.f90 RTTAUK.f90 RTTOVK.f90 TSTRADK.f90 |
| MOD_CPARAM.f90 | jpchus | xxx | I | RTTVI.f90 SUCOMTOVS.f90 tstrad.f90 | | | |
| MOD_CPARAM.f90 | jpnsat | xxx | I | CLEANUP.f90 RTTOVCF.f90 RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | TSTRADTL.f90 | TSTRADAD.f90 | TSTRADK.f90 |
| MOD_CPARAM.f90 | jplev | xxx | I | DEBYE.f90 OPDEP.f90 PRFIN.f90 PRFTAU.f90 PRSLEV.f90 RTINT.f90 RTTAU.f90 RTTOV.f90 RTTOVCF.f90 RTTOVK_TEST.f90 RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | DEBYETL.f90 MOD_PRFVARTL.f90 OPDEPTL.f90 PRFINTL.f90 PRFTAUTL.f90 RTINTTL.f90 RTTOVTL.f90 TSTRADTL.f90 | DEBYEAD.f90 OPDEPAD.f90 PRFINAD.f90 PRFTAUAD.f90 RTINTAD.f90 RTTOVAD.f90 TSTRADAD.f90 | DEBYEK.f90 MOD_PRFVARK.f90 PRFINK.f90 PRFTAUK.f90 RTINTK.f90 RTTOVK.f90 TSTRADK.f90 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MOD_CPARAM.f90 | jpnav | xxx | I | RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | TSTRADTL.f90 | TSTRADAD.f90 | TSTRADK.f90 |
| MOD_CPARAM.f90 | jpnsav | xxx | I | RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | TSTRADTL.f90 | TSTRADAD.f90 | TSTRADK.f90 |
| MOD_CPARAM.f90 | jpnssv | xxx | I | RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | TSTRADTL.f90 | TSTRADAD.f90 | TSTRADK.f90 |
| MOD_CPARAM.f90 | jpncv | xxx | I | RTTVI.f90 SUCOMTOVS.f90 main_testad.f90 main_testk.f90 tstrad.f90 | TSTRADTL.f90 | TSTRADAD.f90 | TSTRADK.f90 |
| MOD_CPARAM.f90 | jpchpf | xxx | I | EMISS.f90 PRFIN.f90 RTINT.f90 RTTOV.f90 RTTOVK_TEST.f90 SUCOMTOVS.f90 tstrad.f90 | EMISSTL.f90 MOD_PRFVARTL.f90 PRFINTL.f90 RTINTTL.f90 RTTOVTL.f90 TSTRADTL.f90 | EMISSAD.f90 PRFINAD.f90 RTINTAD.f90 RTTOVAD.f90 TSTRADAD.f90 | DEBYEK.f90 EMISSK.f90 MOD_GRODYK.f90 MOD_IRCLDK.f90 MOD_PRFVARK.f90 MOD_SURFK.f90 PRFINK.f90 PRFTAUK.f90 RTINTK.f90 RTTAUK.f90 RTTOVK.f90 TSTRADK.f90 |
| MOD_CPARAM.f90 | jpcofm | xxx | I | RTTOVCF.f90 SUCOMTOVS.f90 | | | |
| MOD_CPARAM.f90 | jpcofw | xxx | I | SUCOMTOVS.f90 | | | |
| MOD_CPARAM.f90 | jpcofo | xxx | I | SUCOMTOVS.f90 | | | |
| MOD_CPARAM.f90 | jpst | xxx | I | SUCOMTOVS.f90 | | | |
| MOD_CPARAM.f90 | iu1 | xxx | I | RTTOVCF.f90 | | | |
| MOD_CPARAM.f90 | nulout | xxx | I | ERRORREPORT.f90 RTTOVCF.f90 RTTVI.f90 | | | |
| MOD_CPARAM.f90 | jmwcldtop | xxx | I | SUCOMTOVS.f90 | | | |
| MOD_CPARAM.f90 | pi | xxx | R | EMISS.f90 PRFIN.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 |
| MOD_CPARAM.f90 | deg2rad | xxx | R | EMISS.f90 PRFIN.f90 | | | |
| MOD_CPARAM.f90 | rcnv | xxx | R | PRFIN.f90 RTTOVCF.f90 RTTVI.f90 main_testad.f90 main_testk.f90 tstrad.f90 | PRFINTL.f90 | PRFINAD.f90 | PRFINK.f90 |
| MOD_CPARAM.f90 | rcnw | xxx | R | PRFIN.f90 RTTOVCF.f90 RTTVI.f90 | PRFINTL.f90 | PRFINAD.f90 | PRFINK.f90 |
| MOD_CPARAM.f90 | gravity | xxx | R | EMISCLD.f90 PRFTAU.f90 | EMISCLDTL.f90 PRFTAUTL.f90 | EMISCLDAD.f90 PRFTAUAD.f90 | PRFTAUK.f90 |
| MOD_CPARAM.f90 | speedl | xxx | R | RTTOVCF.f90 | | | |
| MOD_CPARAM.f90 | plcon1 | xxx | R | RTTOVCF.f90 | | | |
| MOD_CPARAM.f90 | plcon2 | xxx | R | RTTOVCF.f90 | | | |

## ANNEX-N: RTTOV-7 Fortran 90 global variables

| Module | Variable | Profile | Type | Direct | Tangent Linear | Adjoint | K | Initialised |
|---|---|---|---|---|---|---|---|---|
| MOD_CPARAM.f90 | njpnsat | | I | PRFIN.f90<br>RTTOV.f90<br>RTTOVCLD.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | RTTOVCLDTL.f90<br>RTTOVTL.f90 | RTTOVAD.f90<br>RTTOVCLDAD.f90 | RTTOVCLDK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njplev | | I | DEBYE.f90<br>OPDEP.f90<br>PRFIN.f90<br>PRFTAU.f90<br>RTINT.f90<br>RTTAU.f90<br>RTTOV.f90<br>RTTOVCLD.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | DEBYETL.f90<br>OPDEPTL.f90<br>PRFINTL.f90<br>PRFTAUTL.f90<br>RTINTTL.f90<br>RTTAUTL.f90<br>RTTOVCLDTL.f90<br>RTTOVTL.f90 | DEBYEAD.f90<br>OPDEPAD.f90<br>PRFINAD.f90<br>PRFTAUAD.f90<br>RTINTAD.f90<br>RTTAUAD.f90<br>RTTOVAD.f90<br>RTTOVCLDAD.f90 | DEBYEK.f90<br>OPDEPK.f90<br>PRFINK.f90<br>PRFTAUK.f90<br>RTINTK.f90<br>RTTAUK.f90<br>RTTOVCLDK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpnav | | I | PRFIN.f90<br>RTTOV.f90<br>RTTOVCLD.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | PRFINTL.f90<br>RTTOVCLDTL.f90<br>RTTOVTL.f90 | PRFINAD.f90<br>RTTOVAD.f90<br>RTTOVCLDAD.f90 | PRFINK.f90<br>RTTOVCLDK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpnsav | | I | PRFIN.f90<br>RTTOV.f90<br>RTTOVCLD.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | PRFINTL.f90<br>RTTOVCLDTL.f90<br>RTTOVTL.f90 | PRFINAD.f90<br>RTTOVAD.f90<br>RTTOVCLDAD.f90 | PRFINK.f90<br>RTTOVCLDK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpnssv | | I | PRFIN.f90<br>RTTOV.f90<br>RTTOVCLD.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | PRFINTL.f90<br>RTTOVCLDTL.f90<br>RTTOVTL.f90 | PRFINAD.f90<br>RTTOVAD.f90<br>RTTOVCLDAD.f90 | PRFINK.f90<br>RTTOVCLDK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpncv | | I | PRFIN.f90<br>RTTOV.f90<br>RTTOVCLD.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | PRFINTL.f90<br>RTTOVCLDTL.f90<br>RTTOVTL.f90 | PRFINAD.f90<br>RTTOVAD.f90<br>RTTOVCLDAD.f90 | PRFINK.f90<br>RTTOVCLDK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njppf | | I | SUCOMTOVS.f90 | | | | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpch | | I | SUCOMTOVS.f90 | | | | SUCOMTOVS.f90 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| MOD_CPARAM.f90 | njpchus | | I | SUCOMTOVS.f90 | | | | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpchpf | | I | SUCOMTOVS.f90 | | | | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpcofm | | I | OPDEP.f90<br>PRFIN.f90<br>PRFTAU.f90<br>RTTOV.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | OPDEPTL.f90<br>PRFINTL.f90<br>PRFTAUTL.f90<br>RTTOVTL.f90 | OPDEPAD.f90<br>PRFINAD.f90<br>PRFTAUAD.f90<br>RTTOVAD.f90 | OPDEPK.f90<br>PRFINK.f90<br>PRFTAUK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpcofw | | I | OPDEP.f90<br>PRFIN.f90<br>PRFTAU.f90<br>RTTOV.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | OPDEPTL.f90<br>PRFINTL.f90<br>PRFTAUTL.f90<br>RTTOVTL.f90 | OPDEPAD.f90<br>PRFINAD.f90<br>PRFTAUAD.f90<br>RTTOVAD.f90 | OPDEPK.f90<br>PRFINK.f90<br>PRFTAUK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpcofo | | I | OPDEP.f90<br>PRFIN.f90<br>PRFTAU.f90<br>RTTOV.f90<br>RTTOVK_TEST.f90<br>SUCOMTOVS.f90 | OPDEPTL.f90<br>PRFINTL.f90<br>PRFTAUTL.f90<br>RTTOVTL.f90 | OPDEPAD.f90<br>PRFINAD.f90<br>PRFTAUAD.f90<br>RTTOVAD.f90 | OPDEPK.f90<br>PRFINK.f90<br>PRFTAUK.f90<br>RTTOVK.f90 | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | njpst | | I | SUCOMTOVS.f90 | | | | SUCOMTOVS.f90 |
| MOD_CPARAM.f90 | nmwcldtop | | I | DEBYE.f90<br>OPDEP.f90<br>SUCOMTOVS.f90 | DEBYETL.f90<br>OPDEPTL.f90 | DEBYEAD.f90<br>OPDEPAD.f90 | DEBYEK.f90<br>OPDEPK.f90 | SUCOMTOVS.f90 |
| MOD_EMISIR.f90 | EmissIR | jpnsat | EmissIRType | CLEANUP.f90<br>EMISS.f90<br>RTTOVCF.f90 | | | | RTTOVCF.f90 |
| MOD_EMISMW.f90 | freqghz | | R | EMISS.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 | EMISS.f90<br>EMISSTL.f90<br>EMISSAD.f90<br>EMISSK.f90 |
| MOD_EMISMW.f90 | pcc | | R | EMISS.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 | EMISS.f90<br>EMISSTL.f90<br>EMISSAD.f90<br>EMISSK.f90 |
| MOD_EMISMW.f90 | pc2 | | R | EMISS.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 | EMISS.f90<br>EMISSTL.f90<br>EMISSAD.f90<br>EMISSK.f90 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| MOD_EMISMW.f90 | ps2 | | R | EMISS.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 | EMISS.f90<br>EMISSTL.f90<br>EMISSAD.f90<br>EMISSK.f90 |
| MOD_EMISMW.f90 | emc | 140 | R | EMISS.f90<br>RTTOVCF.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 | RTTOVCF.f90 |
| MOD_EMISMW.f90 | EmissMW | jpnsat | EmissMWTyp<br>e | CLEANUP.f90<br>EMISS.f90<br>RTTOVCF.f90 | EMISSTL.f90 | EMISSAD.f90 | EMISSK.f90 | RTTOVCF.f90 |
| MOD_FMV.f90 | sensor | jpnsat | R | OPDEP.f90<br>RTINT.f90<br>RTTOVCF.f90 | OPDEPTL.f90<br>RTINTTL.f90 | OPDEPAD.f90<br>RTINTAD.f90 | OPDEPK.f90<br>RTINTK.f90 | RTTOVCF.f90 |
| MOD_GEOCON.f90 | htsat | jpnsat | R | RTTOVCF.f90 | | | | RTTOVCF.f90 |
| MOD_GEOCON.f90 | ratoe | jpnsat | R | PRFIN.f90<br>RTTOVCF.f90 | | | | RTTOVCF.f90 |
| MOD_GEOPTH.f90 | xpath | jppf | R | | | | | |
| MOD_GEOPTH.f90 | xpath1 | jppf | R | | | | | |
| MOD_GEOPTH.f90 | sqtpth | jppf | R | | | | | |
| MOD_GEOPTH.f90 | xpaths | jppf | R | | | | | |
| MOD_GEOPTH.f90 | snad2 | jppf | R | | | | | |
| MOD_GEOPTH.f90 | cnad2 | jppf | R | | | | | |
| MOD_GEOPTH.f90 | czen | jppf | R | | | | | |
| MOD_GEOPTH.f90 | czen2 | jppf | R | | | | | |
| MOD_GEOPTH.f90 | szen2 | jppf | R | | | | | |
| MOD_GRODY.f90 | plandfastem | 6,jppf | R | EMISS.f90<br>PRFIN.f90<br>RTTOVK_TEST.f<br>90 | EMISSTL.f90<br>PRFINTL.f90 | EMISSAD.f90<br>PRFINAD.f90<br>RTTOVAD.f90 | EMISSK.f90<br>PRFINK.f90<br>RTTOVK.f90 | PRFIN.f90<br>RTTOVK_TEST.f90<br>EMISSTL.f90<br>PRFINTL.f90<br>EMISSAD.f90<br>RTTOVAD.f90<br>EMISSK.f90<br>RTTOVK.f90 |
| MOD_GRODYK.f90 | plandfastem | 6,jpchpf | R | RTTOVK_TEST.f<br>90 | | | EMISSK.f90<br>PRFINK.f90<br>RTTOVK.f90 | RTTOVK_TEST.f90<br>EMISSK.f90<br>RTTOVK.f90 |
| MOD_GRODYTL.f90 | plandfastem | 6,jppf | R | | EMISSTL.f90<br>PRFINTL.f90 | EMISSAD.f90<br>PRFINAD.f90<br>RTTOVAD.f90 | | EMISSTL.f90<br>PRFINTL.f90<br>EMISSAD.f90<br>RTTOVAD.f90 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| MOD_IRCLD.f90 | nlevcd | jppf | I | | | | | |
| MOD_IRCLD.f90 | fracpc | jppf | R | | | | | |
| MOD_IRCLDK.f90 | nlevcd | jpchpf | I | | | | | |
| MOD_IRCLDK.f90 | fracpc | jpchpf | R | | | | | |
| MOD_IRCLDTL.f90 | nlevcd | jppf | I | | | | | |
| MOD_IRCLDTL.f90 | fracpc | jppf | R | | | | | |
| MOD_PRFCON.f90 | nrttov | | I | RTTOVCF.f90 RTTVI.f90 | | | | RTTOVCF.f90 |
| MOD_PRFCON.f90 | nlevw1 | | I | | | | | |
| MOD_PRFCON.f90 | nlevw2 | | I | | | | | |
| MOD_PRFCON.f90 | nlevw | | I | RTTVI.f90 | | | | RTTVI.f90 |
| MOD_PRFCON.f90 | wmin | | R | RTTVI.f90 | | | | RTTVI.f90 |
| MOD_PRFCON.f90 | ozmin1 | | R | RTTVI.f90 | | | | RTTVI.f90 |
| MOD_PRFCON.f90 | xpres | jplev | R | PRFIN.f90<br>PRFTAU.f90<br>PRSLEV.f90<br>RTTVI.f90 | PRFTAUTL.f90 | PRFTAUAD.f90 | PRFTAUK.f90 | RTTVI.f90 |
| MOD_PRFCON.f90 | xpres2 | jplev | R | RTTVI.f90 | | | | RTTVI.f90 |
| MOD_PRFCON.f90 | dpres | jplev | R | PRFTAU.f90<br>PRSLEV.f90<br>RTTVI.f90 | PRFTAUTL.f90<br>PRSLEVTL.f90 | PRFTAUAD.f90<br>PRSLEVAD.f90 | PRFTAUK.f90<br>PRSLEVK.f90 | RTTVI.f90 |
| MOD_PRFCON.f90 | wpres3 | jplev | R | | | | | |
| MOD_PRFLIM.f90 | tmax | jplev,jpnsat | R | PRFIN.f90<br>RTTOVCF.f90<br>RTTVI.f90 | | | | RTTOVCF.f90 |
| MOD_PRFLIM.f90 | tmin | jplev,jpnsat | R | PRFIN.f90<br>RTTOVCF.f90<br>RTTVI.f90 | | | | RTTOVCF.f90 |
| MOD_PRFLIM.f90 | qmax | jplev,jpnsat | R | PRFIN.f90<br>RTTOVCF.f90<br>RTTVI.f90 | | | | RTTOVCF.f90 |
| MOD_PRFLIM.f90 | qmin | jplev,jpnsat | R | PRFIN.f90<br>RTTOVCF.f90<br>RTTVI.f90 | | | | RTTOVCF.f90 |
| MOD_PRFLIM.f90 | ozmax | jplev,jpnsat | R | PRFIN.f90<br>RTTOVCF.f90<br>RTTVI.f90 | | | | RTTOVCF.f90 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| MOD_PRFLIM.f90 | ozmin | jplev,jpnsat | R | PRFIN.f90<br>RTTOVCF.f90<br>RTTVI.f90 | | | | RTTOVCF.f90<br>RTTVI.f90 |
| MOD_PRFREF.f90 | pref | jplev,jpnsat | R | RTTOVCF.f90 RTTVI.f90 | | | | RTTOVCF.f90 |
| MOD_PRFREF.f90 | tref | jplev,jpnsat | R | PRFTAU.f90<br>RTTOVCF.f90 | PRFTAUTL.f90 | PRFTAUAD.f90 | PRFTAUK.f90 | RTTOVCF.f90 |
| MOD_PRFREF.f90 | wref | jplev,jpnsat | R | PRFTAU.f90<br>RTTOVCF.f90 | PRFTAUTL.f90 | PRFTAUAD.f90 | PRFTAUK.f90 | RTTOVCF.f90 |
| MOD_PRFREF.f90 | tozref | jplev,jpnsat | R | PRFTAU.f90<br>RTTOVCF.f90 | PRFTAUTL.f90 | PRFTAUAD.f90 | PRFTAUK.f90 | RTTOVCF.f90 |
| MOD_PRFREF.f90 | ozref | jplev,jpnsat | R | PRFTAU.f90<br>RTTOVCF.f90 | PRFTAUTL.f90 | PRFTAUAD.f90 | PRFTAUK.f90 | RTTOVCF.f90 |
| MOD_PRFVAR.f90 | temp | jplev,jppf | R | | | | | |
| MOD_PRFVAR.f90 | wmix | jplev,jppf | R | | | | | |
| MOD_PRFVAR.f90 | ozon | jplev,jppf | R | | | | | |
| MOD_PRFVAR.f90 | cldw | jplev,jppf | R | | | | | |
| MOD_PRFVAR.f90 | ta | jppf | R | | | | | |
| MOD_PRFVAR.f90 | wmixs | jppf | R | | | | | |
| MOD_PRFVAR.f90 | ts | jppf | R | | | | | |
| MOD_PRFVAR.f90 | surfp | jppf | R | | | | | |
| MOD_PRFVAR.f90 | cldp | jppf | R | | | | | |
| MOD_PRFVAR.f90 | cldf | jppf | R | | | | | |
| MOD_PRFVAR.f90 | emis | jpchpf | R | | | | | |
| MOD_PRFVAR.f90 | emis1 | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | temp | jplev,jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | wmix | jplev,jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | ozon | jplev,jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | cldw | jplev,jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | ta | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | wmixs | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | ts | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | surfp | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | cldp | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | cldf | jpchpf | R | | | | | |
| MOD_PRFVARK.f90 | emis | jpchpf | R | | | | | |
| MOD_PRFVARTL.f90 | temp | jplev,jppf | R | | | | | |
| MOD_PRFVARTL.f90 | wmix | jplev,jppf | R | | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| MOD_PRFVARTL.f90 | ozon | jplev,jppf | R | | | | | |
| MOD_PRFVARTL.f90 | cldw | jplev,jppf | R | | | | | |
| MOD_PRFVARTL.f90 | ta | jppf | R | | | | | |
| MOD_PRFVARTL.f90 | wmixs | jppf | R | | | | | |
| MOD_PRFVARTL.f90 | ts | jppf | R | | | | | |
| MOD_PRFVARTL.f90 | surfp | jppf | R | | | | | |
| MOD_PRFVARTL.f90 | cldp | jppf | R | | | | | |
| MOD_PRFVARTL.f90 | cldf | jppf | R | | | | | |
| MOD_PRFVARTL.f90 | emis | jpchpf | R | | | | | |
| MOD_SURF.f90 | nlevsf | jppf | I | | | | | |
| MOD_SURF.f90 | nstype | jppf | I | | | | | |
| MOD_SURF.f90 | fracps | jppf | R | | | | | |
| MOD_SURF.f90 | surfw | jppf | R | | | | | |
| MOD_SURF.f90 | surfd | jppf | R | | | | | |
| MOD_SURFK.f90 | nlevsf | jpchpf | I | | | | | |
| MOD_SURFK.f90 | nstype | jpchpf | I | | | | | |
| MOD_SURFK.f90 | fracps | jpchpf | R | | | | | |
| MOD_SURFK.f90 | surfw | jpchpf | R | | | | | |
| MOD_SURFK.f90 | surfd | jpchpf | R | | | | | |
| MOD_SURFTL.f90 | nlevsf | jppf | I | | | | | |
| MOD_SURFTL.f90 | nstype | jppf | I | | | | | |
| MOD_SURFTL.f90 | fracps | jppf | R | | | | | |
| MOD_SURFTL.f90 | surfw | jppf | R | | | | | |
| MOD_SURFTL.f90 | surfd | jppf | R | | | | | |
| MOD_TAUCFN.f90 | TransCoeffs | jpnsat | TransCoeffsType | CLEANUP.f90<br>OPDEP.f90<br>RTTOVCF.f90 | OPDEPTL.f90 | OPDEPAD.f90 | OPDEPK.f90 | RTTOVCF.f90 |
| MOD_TOVCHN.f90 | ChanConstants | jpnsat | ChanConstantsType | BRIGV.f90<br>CLEANUP.f90<br>EMISCLD.f90<br>EMISS.f90<br>OPDEP.f90<br>PLNCX.f90<br>RTTOVCF.f90 | BRIGVTL.f90<br>EMISCLDTL.f90<br>EMISSTL.f90<br>OPDEPTL.f90<br>PLNCXTL.f90 | BRIGVAD.f90<br>EMISCLDAD.f90<br>EMISSAD.f90<br>OPDEPAD.f90<br>PLNCXAD.f90 | BRIGVK.f90<br>EMISSK.f90<br>OPDEPK.f90<br>PLNCXK.f90 | RTTOVCF.f90 |