KARLSRUHER INSTITUT FÜR TECHNOLOGIE (KIT)

Extensive Air Shower Simulation with CORSIKA: A User's Guide (Version 74xxx from March 2, 2015)

D. Heck and T. Pierog

Institut für Kernphysik

KIT - Universität des Landes Baden-Württemberg und
nationales Forschungszentrum in der Helmholtz-Gemeinschaft

Copyright Notice

Abstract

Extensive Air Shower Simulation with CORSIKA:
A User's Guide

CORSIKA is a detailed simulation program for extensive air showers initiated by high energy cosmic particles. The user's guide explains the installation of the code, all the necessary input data sets, the selection of simulation parameters, and the structure of the program outputs.

Zusammenfassung

Simulation ausgedehnter Luftschauer mit CORSIKA:
Eine Benutzeranleitung

CORSIKA ist ein Programm zur detaillierten Simulation von ausgedehnten Luftschauern, die durch hochenergetische kosmische Strahlung ausgelöst werden. Die vorliegende Anleitung erläutert die Installation des Programms, alle nötigen Eingabedateien, die Wahl der Simulationsparameter und die Struktur der Ausgaben des Programms.

Contents

1 Introduction

CORSIKA (**CO**smic **R**ay **SI**mulations for **KA**scade) is a detailed Monte Carlo program to study the evolution and properties of extensive air showers in the atmosphere. It was developed to perform simulations for the KASCADE experiment [1] at Karlsruhe in Germany. This experiment measured the elemental composition of the primary cosmic radiation in the energy range 3×10^{14} to 1×10^{17} eV and after its upgrade to KASCADE-Grande [2] it reached 10^{18} eV.

The CORSIKA program [3] allows to simulate interactions and decays of nuclei, hadrons, muons, electrons, and photons in the atmosphere up to energies of some 10^{20} eV. It gives type, energy, location, direction and arrival times of all secondary particles that are created in an air shower and pass a selected observation level.

CORSIKA is a complete set of standard FORTRAN routines. It uses no additional program libraries for the simulation of air showers. Therefore, it runs on (almost) every computer where FORTRAN is available.

The CORSIKA program consists basically of 4 parts. The first part is a general program frame handling the in- and output, performing decay of unstable particles, and tracking of the particles taking into account ionization energy loss and deflection by multiple scattering and the Earth's magnetic field. The second part treats the hadronic interactions of nuclei and hadrons with the air nuclei at higher energies. The third part simulates the hadronic interactions at lower energies and the fourth part describes transport and interaction of electrons, positrons, and photons. CORSIKA contains several models for the latter three program parts that may be activated optionally with varying precision of the simulation and consumption of CPU time.

High-energy hadronic interactions may be treated by one of the models: The Dual Parton Model DPMJET [4], the simple Monte Carlo generator HDPM [5] which is inspired by the Dual Parton Model and tries to reproduce relevant kinematical distributions being measured, the quark-gluon-string model QGSJET01 [6], the mini-jet model SIBYLL [7, 8], or VENUS [9]. As a sixth alternative there is added a link to the NEXUS model [10], which combines algorithms of VENUS and QGSJET with new ideas, based on H1 and Zeus data. The last models included are EPOS LHC (v3400) [11] (based on the NEXUS framework but with important improvement concerning hard interactions and nuclear and high density effect, and taking into account LHC data) and QGSJET II-04 [12] which has treatment of Pomeron-Pomeron interaction at any order including loop diagrams (and tuned to LHC). EPOS is the only model available here reproducing heavy ion data from RHIC and LHC.

The low-energy hadronic interactions are simulated alternatively with one of the codes: FLUKA [13] which is a very refined model with many details of nuclear effects, GHEISHA [14] that is a well approved detector Monte Carlo program in the energy region up to some hundred GeV, or UrQMD [15] which describes microscopically the low energetic hadron-nucleus collisions.

For all models the hadronic interaction cross-sections at higher and lower energies are adopted according to the used model.

The interactions of electrons and photons can be treated either with the adapted EGS4 code [16] following each particle and its reactions explicitly, or using the analytic NKG formulae [17] to obtain electron densities at selected locations and the total number of electrons at up to 10 observation levels.

Further on it is optionally possible to explicitly generate Cherenkov light in the atmosphere, to handle electronic and muonic neutrinos and anti-neutrinos, and to simulate showers with flat incidence. Moreover the HERWIG [18] interaction routines have been linked [19] with CORSIKA to handle primary neutrinos.

To shorten the computing times for ultra-high energy showers above 10^{16} eV the thin sampling option exists, by which only a fraction of the secondary particles is followed in the shower development. A different approach to reduce the simulation time solves numerically the cascade equations for the different particle species. This approach is best suited to determine the longitudinal shower profile as measured by the fluorescence telescopes of the P. Auger observatory [20] and is realized in the program CONEX [21, 22] now linked with CORSIKA. To enable a full Monte Carlo simulation without thinning the bulk of subshowers may be run in parallel on a cluster of many CPU-slaves as recently realized in the PARALLEL option [23].

There exists as well a program version that is not suited for air shower simulation but for testing the hadronic interaction models.

A detailed description of the CORSIKA program frame, the used cross-sections, the hadronic interaction model HDPM, the electromagnetic interaction models, and the particle decays has been published in Ref. [3]. For details of the DPMJET, EPOS, NEXUS, QGSJET, SIBYLL, VENUS, FLUKA, GHEISHA, UrQMD, and EGS4 programs see Refs. [4, 24, 11, 10, 6, 12, 7, 9, 13, 14, 15, 16]. However, minor modifications were made to these codes to adapt them for simulation of extensive air showers. A comparison of the various hadronic interaction models is given in Refs. [25, 26, 27].

Besides the explanation [3] of the physics implemented in CORSIKA, this CORSIKA GUIDE is a supplementary description of the technical handling and running of CORSIKA74xxx. It contains information about the installation of the program, the required input data, file formats, parameter settings, outputs, and other technical details. This CORSIKA GUIDE is an updated version of Ref. [28].

For **citation of CORSIKA in your publications** you might use Ref. [3] which is available from the CORSIKA www-page `http://www-ik.fzk.de/corsika/` .

Please do not forget also to make a **reference to the hadronic interaction models** which you used in your simulations. The correct references you find in the bibliography (page 144) of this CORSIKA GUIDE and (in LaTeX format) in the file '*references*.tex' (see directory `doc/`).

If you have problems in installing or running the program, suggestions to improve the code concerning physics, computing, or handling, please contact:

Dr. T. Pierog
Karlsruhe Institute of Technology
Institut für Kernphysik
Hermann-von-Helmholtz-Platz 1
D-76344 Eggenstein-Leopoldshafen
e-mail: <tanguy.pierog@kit.edu>
Fax: +49-(0)721-608-24075
Tel.: +49-(0)721-608-28134

Dr. D. Heck
Karlsruhe Institute of Technology
Institut für Kernphysik
Hermann-von-Helmholtz-Platz 1
D-76344 Eggenstein-Leopoldshafen
e-mail: <dieter.heck@kit.edu>
Fax: +49-(0)721-608-24075
Tel.: +49-(0)721-608-23777

All users of CORSIKA are kindly asked **not to hand over the program to interested new users**, but rather to send their name and address to the above addresses. By these means new users can be provided with news about the latest program version, error corrections, and updates. Thus problems with outdated versions should be avoided.

2 Installation

2.1 CORSIKA Files

The CORSIKA 7.4005 set is distributed as a gzipped .tar file consisting of several files and directories:

- a file *README* giving these short instructions how to proceed,
- an executable shell script file *./coconut* to be used to install CORSIKA,
- a file *AUTHORS* giving the names of authors,
- a file *COPYING* giving copyright instructions,
- a file *ChangeLog* giving the CORSIKA history,
- a file *INSTALL* giving detailed instructions how to use *./coconut* to install CORSIKA,
- a file *NEWS* giving latest news about CORSIKA,
- a file *cDO_NOT_RUN_CONFIGURE* to remind users not to use *./configure* but *./coconut* instead,
- files $Makefile$.am, $Makefile$.in, $acinclude.m4$, $aclocal$.m4, $configure$.in, $configure$. These files and the subdirectory `config/` are needed for the installation and should not be changed unless you know exactly what you are doing. All subdirectories have a $Makefile$.am and a $Makefile$.in which are needed by $configure$ to create the proper $Makefile$s.
- a subdirectory `bernlohr/` containing 'bernlohr' package[2]. It is a set of **C**-routines for Imaging Atmospheric Cherenkov Telescopes (IACT) and for use of external atmospheric profiles in the ATMEXT option; several atmospheric profiles *atmprof*`i.dat` are included together with various auxiliary files. This package is provided by K. Bernlöhr.
- a subdirectory `coast/` in which the COAST interface package[3] is included. COAST is the basis for the COASTUSERLIB, COREAS, INCLINED, and ROOTOUT options.
- a subdirectory `conex/` containing the routines and data sets for the CONEX program to treat the essential part of the shower development by solving numerically the cascade equations to describe the development of the various particle species.
- a subdirectory `doc/` containing:

[2] *bernlohr-1.46*.tar.gz is automatically extracted if needed.
[3] see `http://www-ik.fzk.de/~rulrich/coast`

  - a postscript file *CORSIKA_PHYSICS*.ps containing the physics description of CORSIKA (Report FZKA 6019) (The file *CORSIKA_PHYSICS*.pdf contains it in Acrobat format.),
  - a postscript file *CORSIKA_GUIDE74xxx*.ps giving this User's Guide (The file *CORSIKA_GUIDE74xxx*.pdf contains it in Acrobat format.),
  - a file $references$.tex giving the actual references to CORSIKA and the hadronic interaction models in LaTeX format (to be used in your publications),
  - a file *MPI-Runner_GUIDE*.pdf giving a description how to run CORSIKA on multi-core computers in parallel using Message Passing Interface (MPI),
  - a file *MPI-Runner_PHYSICS*.pdf explains how algorithmically and scientificaly the parallel CORSIKA runs using Message Pasing Interface (MPI) are organized,

- a subdirectory `dpmjet/` containing the source files of the DPMJET-II.55 model ($dpmjet253bc$.f, $dpmjet253c$.f,, $dpmjet254$.f, $dpmjet255c$.f, $dpmjet256c$.f),

- a subdirectory `epos/` containing the source package for the EPOS model[4].

- a subdirectory `herwig/` for installation of HERWIG[5] and containing a *README* file inside,

- a subdirectory `include/` containing the $config.h$.in file which is needed by *configure* to do $config$.h. The file $config$.h contains all the preprocessor commands needed by CORSIKA.

- a subdirectory `lib/` together with a script file *libtool* are created when compiling CORSIKA. The subdirectory contains all object files and libraries.

- a subdirectory `nexus/` containing the source package for the NEXUS model[6].

- a subdirectory `pythia/` containing the modified and extended PYTHIA 6.411 package[7] and a *README.corsika* file inside,

- a subdirectory `run/` to be used to run CORSIKA containing:

  - 14 data sets containing the energy dependent cross-sections for nucleon-nucleus processes (*NUCNUCCS*), for electromagnetic interactions (6 sets *EGSDAT6*_x.x), for QGSJET01 (*QGSDAT01* and *SECTNU*), for QGSJET-II (*sectnu-II-04*) and *qgsdat-II-04*), numerical data to be used by the VENUS routines (*VENUSDAT*), cross-

[4] $epos-lhc-v3400\_cors$.tar.gz is automatically extracted if needed.
[5] Version $herwig6510$ should be used.
[6] $nexus3.97\_cors$.tar.gz is automatically extracted if needed.
[7] $pythia$.tar.gz is automatically extracted if needed.

section table for UrQMD (*UrQMD-⟨VER⟩-xs*.dat) and for the DPMJET routines the Glauber tables (*GLAUBTAR*.DAT) and nuclear data (*NUCLEAR*.BIN[8]),

  - 8 input examples *all-inputs**, *conex-3D-inputs*, and *parallel-inputs* to steer the simulation with any model (with/out THIN, with STACKIN, with EPOS, with NEXUS, with CONEX, or with PARALLEL),
  - 3 data sets *atmabs*.dat, *mirreff*.dat, and *quanteff*.dat to take into account the atmospheric absorption, mirror reflectivity, and quantum efficiency of Cherenkov radiation,
  - the executable binary files of CORSIKA and of utility programs after compilation.

- a subdirectory `src/` containing:

  - the major part of the source code ($corsika$.F and $corsika$.h) which is a FORTRAN code file with some **C**-preprocessor commands, that contains the code of CORSIKA including the EGS4 and HDPM model routines. It contains also the interfaces to FLUKA, GHEISHA, UrQMD, DPMJET, EPOS, NEXUS, QGSJET, SIBYLL, and VENUS, but without those hadronic interaction codes. Also the interface for linking with the CONEX routines are contained within ($corsika$.F and $corsika$.h),
  - 5 separate files with pure FORTRAN code of the GHEISHA, QGSJET01d, QGSJET-II-04, SIBYLL2.1, and VENUS routines ($gheisha\_2002d$.f, $qgsjet01d$.f, $qgsjet-II-04$.f, $sibyll2.1$.f, and $venus$.f),
  - 1 separate files with the pure **C**-code of the EFIELD option ($elfield$.c),
  - 3 special **C**-routines needed for compilation ($timerc$.c, $tobuf$.c, and $trapfpe$.c),
  - a subsubdirectory `parallel/` containing the shell scripts and **C**-programs to run CORSIKA distributed on many CPUs in parallel. This directory also houses the `userguide-pll-*`.txt files which describe the parallel handling of CORSIKA on various CPU-clusters using these shell scripts and some FORTRAN programs to handle the results of a parallel simulation.
  - a subsubdirectory `preshower2.0-rev/` containing the **C**-programs for production of γ-induced preshowers before reaching the top of atmosphere ($preshw$.c, $utils$.c, $utils$.h, $veto$.c, and $veto$.h),
  - a subsubdirectory `utils/` containing utility programs:
    - a **C**$^{++}$-routine to read the binary particle output file ($corsikaread$.cpp),
    - a FORTRAN routine $cors2input$.f to reestablish the used input from the binary particle output file,
    - 3 FORTRAN routines to read the binary particle output files ($corsikaread$.f, $corsikaread\_history$.f, and $corsikaread\_thin$.f),

[8]The *NUCLEAR*.BIN file may be used only with those computers (DEC-UNIX, LINUX) applying the IEEE standard for direct access read and write in **32-bit mode**. In the 64-bit mode the compiler option `-frecord-marker=4` must be used. For other computers (DEC-VAX, HP) a different *NUCLEAR*.BIN file is necessary which is available from CERN or from the author of DPMJET <johannes.ranft@cern.ch>.

- a FORTRAN routine $corsplitevts$.f to split a large particle output file containing several CORSIKA events into files containing single events each,
- a **C**-routine $map2png$.c to visualize the shower plots established with the option PLOTSH2,
- a FORTRAN routine $modelprint$.f which displays the used high-energy and low-energy models from the binary particle output file,
- a FORTRAN routine $plottracks3c$.f and its include file $work$.inc to visualize shower plots established with the option PLOTSH,
- a FORTRAN routine $readcsk2asci$.f converting a binary particle output file to a (readable) ASCII file,
- a FORTRAN routine $readtimes$.f to sum up the CPU times of several .lst files (useful at parallel runs),
- two FORTRAN routines $showsimprods$.f and $showsimulist$.f to establish a tabular of several simulations giving characteristic values of each simulation (primary-id, energy, angles of incidence, ...) from the binary particle output files,
- a FORTRAN routine $sumlistnkginfo$.f to sum up the NKG-output of .lst files to get XMAX averaged over the used showers,
- a FORTRAN routine $sumlongifiles$.f to sum up the content of several .longi files (also in case of parallel simulations).

- a subdirectory `urqmd/` containing the source package of the UrQMD model[9].

The CORSIKA code and the files belonging to it can be obtained from Karlsruhe Institute of Technology by anonymous ftp. Before access the host name of your computer is checked for authorization. If you want to transfer CORSIKA files to your computer system you might proceed as follows (commands that you have to type are underlined):

```
ftp ftp-ik.fzk.de
Trying 141.52.67.78...
Connected to ikserv.fzk.de.
220 ftp-ik FTP server (Version [number & date]) ready.
Name (ftp-ik.fzk.de:username):  anonymous [or ftp]
331 Guest login ok, send your complete e-mail address as password.
Password:  (your_mail_address)
230-   PROBLEMS
230-   If your FTP client crashes or hangs shortly after login please try
230-   using a dash (-) as the first character of your password.  This will
230-   turn off the informational messages that may be confusing your FTP
230-   client. Administrative contact: bekk@ik.fzk.de
```

[9] $urqmd1.3\_cors$.tar.gz is automatically extracted if needed.

```
230-
230-
230-                        Welcome to ftp-ik.fzk.de, the
230-                ******  I N F O and S O F T  Server  ******
230-           Forschungszentrum Karlsruhe, Institut fuer Kernphysik
230-                   KASCADE Collaboration Karlsruhe
230-
230-
230- current directory: /     local time:  [date & time]
230- You are user number # out of a possible total of 12 in your domain class
230- All transfers to and from ftp-ik.fzk.de are logged. If you don't like this
230- then disconnect now!
230-
230- Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp > cd pub/corsika/v740
250 CWD command successful.
ftp > mget *
mget corsika-74xxx.tar.gz?  y
.
.
.
ftp > quit
221-You have transferred # bytes in # files.
221-Total traffic for this session was # bytes in # transfers.
221-Thank you for using the FTP service on ftp-ik.fzk.de.
221 Goodbye.
```

For faster transmission we have compressed the *corsika-74xxx.tar* file. Before using this file first you have to decompress it by applying the 'gunzip' procedure.

If you are not successful to fetch files from the subdirectory `pub/corsika/v740` (because of 'permission denied' or 'No such file or directory'), then please try to copy the *README* file located in the subdirectory `pub/` . This file has no limited access and should be readable by you. If you have copied this *README* file successfully please send an e-mail to <tanguy.pierog@kit.edu> or <dieter.heck@kit.edu>. The automatic registration of each access to this file from outside gives a hint what to change to enable your access to the CORSIKA files.

The files belonging to FLUKA are collected in an object file library. This library and the necessary data files are distributed by the FLUKA organization for different computers and compiler versions. You find further details on the web page: `http://www.fluka.org/` .

The files belonging to the HERWIG code[10] may be downloaded from the web page:
`http://www.hep.phy.cam.ac.uk/theory/webber/herwig/ .`

2.2 Program File Preparation

By unpacking the *corsika-74xxx.tar* file with the command
`tar xvf` *corsika-74xxx.tar*
the file structure of CORSIKA will be established. To install CORSIKA you type

```
./coconut
```

(no argument) and answer the questions ... it's done !

The main source file is $corsika$.F with all common blocks in $corsika$.h. It's a FORTRAN source file with some **C**-preprocessor commands which allows optional compilation of some part of the code. It uses standard $Makefile$ and compiler options, and therefore doesn't require any special software to be installed on your machine[11]. Using the shell script *./coconut*, the user can interactively select the specific CORSIKA version for his application and compile it to get directly ready-to-run executable binary files in the `run/` subdirectory. Computer dependent options and $Makefile$s are prepared automatically by the shell script $configure$ called by *./coconut*. The $configure$ is a standard portable shell script used together with $make$ by GNU-packages to be installed, but $configure$ should **NOT** be used directly to get a proper installation of CORSIKA. The *./coconut* shell script has been designed to get a "user friendly" machine dependent installation, so that binary files can be compiled in parallel from the same source directory but on different systems[12] (in a large computer farm for instance). All object files and libraries are "hidden" in a `lib/` subdirectory. If you already used a former version of CORSIKA, and you want to use a $corsika\_compilefile$.f as before, you can optionally save this file during the installation process to compile it yourself.
If you extract executable binary files with different compilers which produce object files and libraries which are not compatible with each other (e.g. 'g77' and 'gfortran') then it is recommended to call

```
./coconut -d
```

between the extractions to erase all established libraries (see Sect. 11 page 124).
The *./coconut* script checks for all options if they can be used on your computer, so comments appearing during the installation should be read carefully.

2.2.1 Options

At present CORSIKA versions may be generated with the following hadronic interaction models with their cross-section (for determining the mean free path between the interactions) using

[10] Version $herwig6510$ should be used.
[11] A shell terminal, a **C**-compiler and a FORTRAN compiler are included in any UNIX based system.
[12] In principle, any system could be used, but only LINUX, Mac OSX, Dec OSF, SunOS, AIX have been tested.

the options:

DPMJET selects DPMJET 2.55 routines for the simulation of high energy hadron-nucleus and nucleus-nucleus collisions. Also the DPMJET cross-sections are selected.

EPOS selects the EPOS LHC (v3400) routines for the simulation of high energy hadron-nucleus and nucleus-nucleus collisions. Also the EPOS cross-sections are selected.

NEXUS selects the NEXUS 3.97 routines for the simulation of high energy hadron-nucleus and nucleus-nucleus collisions. Also the NEXUS 3.97 cross-sections are selected.

QGSJET selects QGSJET01d routines for the simulation of high energy hadron-nucleus and nucleus-nucleus collisions. Also the QGSJET01d cross-sections are selected.

QGSII selects QGSJET-II-04 routines for the simulation of high-energy hadron-nucleus and nucleus-nucleus collisions. Also the QGSJET-II-04 cross-sections are selected.

SIBYLL selects SIBYLL 2.1 routines for the simulation of high-energy hadron-nucleus and nucleus-nucleus collisions. Also the SIBYLL cross-sections are selected.

VENUS selects VENUS routines for the simulation of high-energy hadron-nucleus and nucleus-nucleus collisions. Also the VENUS cross-sections are selected.

FLUKA selects the FLUKA 2011 model[13] for the simulation of low-energy hadron-nucleus collisions. Always the appropriate FLUKA cross-sections are used. This option may be combined with all high energy interaction models excluding DPMJET 2.55.

GHEISHA selects the GHEISHA 2002d routines for the simulation of low-energy hadron-nucleus collisions. Always the appropriate GHEISHA cross-sections are used. This option may be combined with all high energy interaction models.

URQMD selects the UrQMD 1.3_cors routines for the simulation of low-energy hadron-nucleus collisions. Always the appropriate UrQMD cross-sections are used. This option may be combined with all high energy interaction models.

The **default** setting obtained without specifying any option is QGSJET. For low-energy hadronic interactions the GHEISHA model is taken.
In addition to these options you may select the following preprocessing options:

[13] As there exist several FLUKA libraries suited for different CPU/compiler versions (Linux 32-bit, Linux 64-bit for gfortran $<$ 4.5.1, Linux 64-bit for more recent gfortran versions), you should select the FLUKA library fitting with your installation.

ANAHIST selects code to generate a histogram file $datnnnnnn$.lhbook for a short analysis of essential properties of the particles arriving at ground (as it is usually performed for the showers simulated at Lyon for the Auger experiment). This option needs the THIN option and linking with the CERN library[14] to get the routines of the HBOOK and ZEBRA packages [29].

ATMEXT selects code for treatment of the atmosphere according to MODTRAN model for various atmospheres by tabulated values. ATMEXT is recommended with the CERENKOV option for careful treatment of refractive index. This option links with (compiled) *atmo*.c, *fileopen*.c, and *straux*.c routines of the 'bernlohr' package.

AUGCERLONG selects code to fill the Cherenkov column in the table giving the longitudinal particle distribution.

AUGERHIST selects code to generate a histogram file $datnnnnnn$.lhbook, containing various histograms of different particle types at up to 20 vertical atmospheric depths to follow the development of shower properties, which are of interest for the Auger experiment (Cherenkov and fluorescence photon production). This option needs the THIN + AUGERINFO options and linking with the CERN library[14] to get the routines of the HBOOK and ZEBRA packages [29].

AUGERINFO selects code which writes a file named $DATnnnnnn$.info instead of DAT-$nnnnnn$.dbase file.

CEFFIC selects code to respect the atmospheric absorption, mirror reflectivity, and photomultiplier quantum efficiency of Cherenkov light. This option is only available in connection with the CERENKOV option.

CERENKOV selects code for additional generation of Cherenkov light. It needs the simulation with EGS4.

CERWLEN selects Cherenkov code including the wavelength dependent generation of Cherenkov photons and respects their wavelength dependent refraction of the atmosphere. This option is only available in connection with the CERENKOV option.

CHARM selects code for the explicite treatment of charmed particles (only in connection with interaction codes which produce them) and the τ-lepton rsp. the τ-neutrinos (in connection with the NEUTRINO or NUPRIM option). CHARM excludes the TAULEP option.

COASTUSERLIB activates the use of a COAST user library (libCOAST.so) which has to be compiled separately and the parent directory of which must be specified in the environment variable `$COAST_USER_LIB`. For this option *root* needs to be installed on your system.

[14] The CERN library is only available for 32bit computers.

COMPACT selects a compacted output format for the particle file (unit MPATAP) which is different from the standard output, but better suited for simulations of very large numbers of low energy showers, which produce mostly no or only very few particles.

CONEX selects code to solve numerically the cascade equations which describe the shower development after the very first interactions. This gives an enormeous speed-up of the simulation especially at primary energies above 10^{17} eV. This option works only in connection with the high-energy hadronic interaction models EPOS, QGSJET, QGSII, or SIBYLL in combination with all three low-energy interaction models. The CONEX option is only available in connection with the CURVED + SLANT + THIN + UPWARD options.

COREAS activates the direct calculation of radio emission from extensive air showers on the basis of the "endpoint formalism". The enclosed COAST package is used to provide the interfacing with the radio emission code. Please refer to the separate CoREAS-manual (available after compiling with the COREAS option switched on) for further details.

CURVED selects special code to treat showers with large zenith angles $70^\circ < \theta < 90^\circ$.

EFIELD selects special code to take into account the atmospheric electrical fields on the transport of electrons/positrons.

EHISTORY selects additional code to bring (in combination with the MUADDI, NUADDI, or EMADDI keyword) additional information on the muon, neutrino, and/or electron and photon histories to the particle output file.

IACT selects code for simulation of **I**maging **A**tmospheric **C**herenkov **T**elescope arrays. This option is only available in connection with the CERENKOV option and links with (compiled) *iact*.c routines of the 'bernlohr' package.

IACTEXT selects code for extended interfacing with the *iact*.c routines of the 'bernlohr' package. This option is only available in connection with the CERENKOV + IACT option.

INCLINED enables the output of ground particles on an inclined plane, which can be user-defined by the INCLIN keywork. The `InclinedPlane` routines used by this option are provided by the enclosed COAST package.

INTCLONG selects the longitudinal distribution of Cherenkov photons in longitudinally integrated form. (The default gives the generation of photons/step, so-called differential longitudinal distribution.) INTCLONG excludes the NOCLONG option. This option is only available in connection with the CERENKOV or AUGCERLONG option.

INTTEST selects special features for the test of the interaction models (needs routines of the HBOOK and ZEBRA packages [29]). This option is not suited for extensive air shower simu-

lation.

LPM selects those EGS4 routines which include the LPM-effect, but without using the THIN option.

MUONHIST selects code to generate a histogram file $datnnnnnn$.lhbook for the analysis of muon properties of showers. This option needs linking with the CERN library[15] to get the routines of the HBOOK and ZEBRA packages [29].

MUPROD selects additional code to bring (in combination with the MUADDI keyword) information to the particle output file on those muons which decay or interact before reaching the observation level. A combination with the EHISTORY option is recommended.

NEUTRINO selects code by which neutrinos emerging from pion, kaon, and muon decays are tracked explicitly but without interaction.

NOCLONG deselects the longitudinal Cherenkov photon distribution. NOCLONG excludes the INTCLONG option. This option is only available in connection with the CERENKOV or AUGCERLONG option.

NUPRIM selects the HERWIG code to treat the first interaction of a primary neutrino.

PARALLEL selects code for calculation of a shower distributed on many cores in parallel using a set of shell scripts.

PARALLELIB selects code for calculation of a shower distributed on many cores in parallel with MPI (Message Passing Interface). This option is only available in connection with the PARALLEL option.

PLOTSH selects code which enables plotting the tracks of the electromagnetic, muonic, and hadronic particles to demonstrate the development of a shower optically.

PLOTSH2 selects code which enables plotting the tracks of the electromagnetic, muonic, and hadronic particles to demonstrate the development of a shower optically, avoiding the large output files of PLOTSH.

PRESHOWER selects routines for coupling with a **C**-program package to describe the interaction of primary gammas with the Earth's magnetic field at EeV energies and to treat the resulting swarm of em-particles falling onto the top of atmosphere as one shower.

ROOTOUT selects routines for writing the particle output in a *root* file e.g. for off-line analysis

[15]The CERN library is only available for 32bit computers.

of the particle output by *root* routines (on the basis of the COAST package).

SLANT selects a slant depth scale for the longitudinal distributions instead of the vertical depth scale used otherwise.

STACKIN selects code to read in the parameters (type, energy, momenta) of secondary particles resulting from the interaction of an exotic primary (WIMP) which has been treated off-line by a separate program.

TAULEP selects code for the explicite treatment of the τ-lepton rsp. the τ-neutrinos (in connection with the NEUTRINO or NUPRIM option). TAULEP excludes the CHARM option.

THIN selects the thinning mechanism to reduce the computing time for full simulations, especially for EGS4, and activates the LPM-effect.

TRAJECT selects the zenith and azimuth angles in a manner that a (gamma or neutrino) source is followed along its trajectory in the sky.

UPWARD selects code to treat upward going particles. This option treats particles in the upward direction (zenith angle $\geq 90^{\circ}$).

VIEWCONE selects the primary direction to come from a cone around a fixed zenith and azimuth angle.

VOLUMECORR selects the angular dependence of the zenith angle distribution as it is needed for a vertical long string detector e.g. AMANDA or ICECUBE (see page 58). (The default takes the zenith angle distribution as observed by a horizontal flat detector.)

VOLUMEDET selects the angular dependence of the zenith angle distribution as it will be observed by a volume detector e.g. Cherenkov telescope (see page 58). (The default takes the zenith angle distribution as observed by a horizontal flat detector.)

The major part of the program is machine independent due to the restriction to FORTRAN standards. Nevertheless, there are a few points where computer specific adaptations were necessary. There are prepared options for PCs and work stations running under UNIX (also LINUX) and for Apple Macintosh computers. The versions are automatically[16] activated for the following options:

UNIX selects code for calculation on UNIX systems including derivatives like LINUX (e.g. DEC-station under ULTRIX, ALPHA-station under DEC-UNIX [Tru64])[17].

[16] Thanks to $configure$, *./coconut* checks your machine configuration and sets the proper options.

[17] Some UNIX machines (HP, IBM RS6000, and installations with GNU g77 compiler [e.g. LINUX]) need the record length parameter RECL (used in the OPEN statement for the external stack MEXST) in bytes instead of

MAC selects code for calculation on a former 68k-APPLE Macintosh.

Within the $corsika$.F file the optional code for the various versions (to be unpacked by the preprocessor) is marked by

```
#if __flag__
     ...
#else
     ...
#endif
```

blocks. If you are using a sequential computer other than the ones listed here you should try to adapt the UNIX (or perhaps MAC) version to your machine as this is most straight forward to understand.
In the `src/` subdirectory, the $corsika$.F file contains some explanations, the main program with more than 121 subroutines and functions, the EGS4 routines, the NKG routines, the HDPM model routines, the interface routines to FLUKA, GHEISHA, UrQMD, DPMJET, EPOS NEXUS, QGSJET, SIBYLL, and VENUS, the interface routines to CONEX, the special routines for Cherenkov light generation, the routines for generation of Auger-oriented histograms, and the interaction test routines. It is about 77500 lines long. The $corsika$.h file (length about 4900 lines) contains the general common blocks with their explanations. These common blocks are included into the source file during preprocessing.
The **C**-file $timerc$.c contains a 'date and time' routine *timerc* for those UNIX or LINUX systems where the more modern *date_and_time* system routine is not available. The subsubdirectory `src/preshower2.0-rev/` contains the **C**-routines $preshw$.c, $utils$.c, $utils$.h, $veto$.c, and $veto$.h.
The $gheisha\_2002d$.f, $qgsjet$-II-04.f, $qgsjet01d$.f, $sibyll2.1$.f, and $venus$.f routines are about 18200, 17700, 7500, 8100, and 18000 lines, respectively.
Because of its size the DPMJET code has been divided into 5 portions $dpmjet253$bc.f, $dpmjet$-253c.f, $dpmjet254$.f, $dpmjet255$c.f, and $dpmjet256$c.f with 5280, 42400, 2300, 42100, and 30000 lines each in the `dpmjet/` subdirectory.

4-byte words. This holds also for DEC-UNIX running the f77 compiler with the option '-assume bytrecl'. For easy adaptation the additional preprocessor-option **BYTERECL** is available to enlarge the RECL-parameter by a factor 4.

Most UNIX systems offer a system routine *date_and_time* which overcomes the millenary border. In systems without this internal routine the preprocessor-option **OLDDATE** selects an older routine for date and time. The similar option **OLDDATE2** selects an alternative needed for PGF77 environment on LINUX platforms (IN2P3 Computing Center at Lyon) or HP-UX machines (with the +E1 option of the fort77 compiler). This selection is attainable only by giving '*./coconut -e*' (expert mode of coconut).

The preprocessor-option **IBMRISC** selects the routines $date$ and $clock$ available on IBM RS6000 machines.

If necessary you might select the option **TIMERC** or you should adapt subroutine $prtime$ to call the routines of your system for date and time.

The options **BYTERECL**, **OLDDATE**, **OLDDATE2**, **IBMRISC**, and **TIMERC** are only available in connection with the UNIX option and can not be selected if it's not adapted to your system. They are attainable by calling '*./coconut -e*' (expert mode of coconut').

The 30 UrQMD 1.3_cors program files with the 12 include files are collected within the urqmd-1.3_cors.tar.gz[18] file in the `urqmd/` subdirectory.
The 24 EPOS program files with the 6 include files and the 7 data files are collected within the epos-lhc-v3400_cors.tar.gz[18] file in the `epos/` subdirectory.
The 21 NEXUS 3.97 program files with the 5 include files and the 4 data files are collected within the nexus3.97_cors.tar.gz[18] file in the `nexus/` subdirectory.
The CONEX program and data files are collected within the conex-4.3700.tar.gz[18] file in the `conex/` subdirectory.
The load modules of the FLUKA 2011 library routines and the needed FLUKA commons and data files may be downloaded (after being authorized) from the official FLUKA web page `http://www.fluka.org/` and properly installed on your machine by setting the environment variable FLUPRO[19]. For further information on the FLUKA package you may consult the FLUKA web page or contact A. Ferrari[20].
The HERWIG routines[21] needed for the NUPRIM version may be downloaded from `http://www.hep.phy.cam.ac.uk/theory/webber/herwig/` and adapted according to the README file in the `herwig/` subdirectory (see Sect. A.1 page 131).

2.2.2 Example of Program Installation

As an example, if you have the file *corsika-74xxx*.tar.gz in the current directory, the installation on a LINUX system will look like (commands you are giving are underlined):

```
> tar -zxf corsika-74xxx.tar.gz
> cd corsika-74xxx/
> ./coconut

|=============================================================================
|                          Welcome to COCONUT (v3.1)
|                 -- the CORSIKA CONfiguration UTility --
|=============================================================================
|
|           create an executable of a specific CORSIKA version
|
|
|       Please read the documentation for a detailed description
|                    of the options and how to use it.
|
|             Try './coconut -h' to get some help about COCONUT
|         Use './coconut --expert' to enable additional configuration steps.
```

[18] Automatically unpacked if selected.
[19] Assuming *csh* shell one uses: `setenv FLUPRO` *flukadirectory*. For bash/sh shells you give: `export FLUPRO=`*flukadirectory*.
[20] <alfredo.ferrari@cern.ch> or <alfredo.ferrari@mi.infn.it>.
[21] Version *herwig*6510 should be used.

```
|
|   (press 'Enter' to select an option followed by "[DEFAULT]" or "[CACHED]")
|
|=============================================================================

*****************************************************************************
*    WARNING : File c̈ompilefile.fF̈ound in src/ !
*
*            This temporary file is renamed "compilefile.f.bak"
*            to prevenrt conflict with the following insatallation...
*****************************************************************************

*****************************************************************************
*    INFO:
*         You are using the cached configuration from "include/config.h".
*         To turn off this you may use the --no-cache option.
*****************************************************************************

-----------------------------------------------------------------------
Compile in 32 or 64bit mode ?
1 - Force 32bit mode
2 - Use compiler default ('-m64' on a 64bit machine) [CACHED]

r - restart (reset all options)
x - exit make

(only one choice possible):
SELECTED : NOM32

-----------------------------------------------------------------------
Which high energy hadronic interaction model do you want to use ?
1 - DPMJET 2.55
2 - EPOS LHC
3 - NEXUS 3.97
4 - QGSJET 01D (enlarged commons) [CACHED]
5 - QGSJETII-04
6 - SIBYLL 2.1
7 - VENUS 4.12

r - restart make
x - exit make

(only one choice possible)
```

```
4
SELECTED: QGSJET01
```

Answer the questions to select the options you want to use in CORSIKA (see Sect. 2.2.1 page 9 and Sect. 3 page 28 for details). Press "Enter" to select the options marked by "[CACHED]". After each question, you can choose to restart the installation at the first question ("r") or to stop now ("x").
If an option needs some other files, the installation program will test if they exist[22]*. If a problem occurs* [23]*, the installation program will suggest the solution. Just follow what is written.*
Using the default options on a LINUX system, you will get:

```
...
------------------------------------------------------------------------
      Your final selection to build CORSIKA is:

  options:  HORIZONTAL QGSJET01 GHEISHA TIMENEW

------------------------------------------------------------------------
 Configuration is finished.  How do you want to proceed ?
   f - Compiling and remove temporary files [DEFAULT]
   k - Compile and keep extracted CORSIKA source code
   n - Just extract source code.  Do not compile!

   r - restart
   x - exit make

   (only one choice possible):
```

At this point, you should press "Enter" (rsp. "f"). If you want to see the code used to compile your CORSIKA program, you can answer "k", but this is not needed by the automatic compilation. Finally the installation program creates proper Makefiles and then runs `make install` *to compile your CORSIKA program. Select "n" only if you want to modify the source code before compiling or if you want to use your former installation tools, you prepared for an older version of CORSIKA using a "compilefile.f". This option will preprocess the source code, save it in subdirectory* `src/` *as "corsika" followed by the version number "74xxx", an underscore with the selected high energy hadronic interaction model*[24] *and an underscore with the selected low energy hadronic interaction model*[25] *(and an optional "_int" if you select the "INTTEST" option), and create the* $Makefiles$*. But it will not run* `make`*. See Sect. 11 (page 124) for more details.*

[22] The bernlohr, epos, nexus, and urqmd packages will be automatically unpacked in their subdirectories.
[23] Trying to use the CERN library when not installed for instance.
[24] DPMJET, EPOS, NEXUS, QGSJET, QGSII, SIBYLL, or VENUS.
[25] fluka, gheisha, or urqmd.

```
  SELECTED        :  COMPILE
checking whether to enable maintainer-specific portions of Makefiles...  no
checking build system type...  x86_64-suse-linux
checking host system type...  x86_64-suse-linux
checking for a BSD-compatible install...  /usr/bin/install -c
checking whether build environment is sane...  yes
checking for gawk...  gawk
checking whether make sets $(MAKE)...  yes
checking for pgf77...  no
checking for ifc...  no
checking for ifort...  no
checking for g77...  g77
checking whether the Fortran 77 compiler works...  yes
checking for Fortran 77 compiler default output file name...  a.out
checking for suffix of executables...
checking whether we are cross compiling...  no
checking for suffix of object files...  o
checking whether we are using the GNU Fortran 77 compiler...  yes
checking whether g77 accepts -g...  yes
checking for cc...  cc
...
```

It may take a while ...

```
...
checking do not compile binaries, just extract CORSIKA compilefile...  (cached) no
checking to keep the CORSIKA compilefile...  (cached) no
configure:  creating ./config.status
config.status:  creating Makefile
config.status:  creating bernlohr/Makefile
config.status:  creating conex/Makefile
config.status:  creating dpmjet/Makefile
config.status:  creating epos/Makefile
config.status:  creating pythia/Makefile
config.status:  creating herwig/Makefile
config.status:  creating nexus/Makefile
config.status:  creating urqmd/Makefile
config.status:  creating src/Makefile
config.status:  creating run/Makefile
config.status:  creating doc/Makefile
config.status:  creating lib/Makefile
config.status:  creating include/config.h
config.status:  executing depfiles commands
config.status:  executing libtool commands
```

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
configure:  Optional packages not found or deactivated on this system:  FLUKA COAST ROOT PYTHIA

Compile CORSIKA in "../../../lib/Linux" and copy executable in "../../../run"

Making install in src
make[1]:  Entering directory ‘../../../src’
cc -E corsika.F -DHAVE_CONFIG_H -I../include -traditional-cpp -o corsikacompilefile.f
g77 -O0 -g -fbounds-check -c -o corsika-corsikacompilefile.o ‘test -f
          ’corsikacompilefile.f’ || echo ’./’‘corsikacompilefile.f
if cc -DHAVE_CONFIG_H -I. -I. -I../include -g -MT trapfpe.o -MD -MP -MF ".deps/trapfpe.Tpo"
          -c -o trapfpe.o trapfpe.c;
then mv -f ".deps/trapfpe.Tpo" ".deps/trapfpe.Po"; else rm -f ".deps/trapfpe.Tpo"; exit 1; fi
g77 -O0 -g -fbounds-check -c -o corsika-qgsjet01d.o ‘test -f ’qgsjet01d.f’ || echo
          ’./’‘qgsjet01d.f
g77 -O0 -g -fbounds-check -c -o corsika-gheisha_2002d.o ‘test -f
          ’gheisha_2002d.f’ || echo ’./’‘gheisha_2002d.f
/bin/sh ../libtool --mode=link g77 -g -fbounds-check -o corsika corsika-corsikacompilefile.o
          trapfpe.o corsika-qgsjet01d.o corsika-gheisha_2002d.o -L/../../lib/Linux
mkdir .libs
g77 -g -fbounds-check -o corsika corsika-corsikacompilefile.o trapfpe.o corsika-qgsjet01d.o
          corsika-gheisha_2002d.o -L/../../lib/Linux
g77 -g -fbounds-check -c -o plottracks3c.o plottracks3c.f
/bin/sh ../libtool --mode=link g77 -g -fbounds-check -o plottracks plottracks3c.o
g77 -g -fbounds-check -o plottracks plottracks3c.o
make[2]:  Entering directory ‘../../../src’
test -z "/../../run" || mkdir -p -- .  "../../../run"
 /bin/sh ../libtool --mode=install /usr/bin/install -c ‘corsika’ ‘../../../run/corsika’
/usr/bin/install -c corsika ../../../run/corsika
 /bin/sh ../libtool --mode=install /usr/bin/install -c ‘plottracks’ ‘../../../run/plottracks’
/usr/bin/install -c plottracks ../../../run/plottracks
make[2]:  Nothing to be done for ‘install-data-am’.
make[2]:  Leaving directory ‘../../../src’
make[1]:  Leaving directory ‘../../../src’
Making install in .
make[1]:  Entering directory ‘../../..’
make[2]:  Entering directory ‘../../..’
make install-exec-hook
make[3]:  Entering directory ‘../../..’

--> "corsika74xxxLinux_QGSJET_gheisha" successfully installed in :
   run/
```

```
--> You can run CORSIKA in run/ using for instance :
   ./corsika74xxxLinux_QGSJET_gheisha < all-inputs > output.lst

make[3]:  Leaving directory '../../..'
make[2]:  Nothing to be done for 'install-data-am'.
make[2]:  Leaving directory '../../..'
make[1]:  Leaving directory '../../..'
```

Now the installation is finished. As written, an executable binary file is copied into the `run/` subdirectory where all data files are placed. The name of this file is composed by "corsika" followed by the version number "74xxx", the system name as given by the `uname` UNIX command, an underscore with the selected high energy hadronic interaction model[26] and an underscore with the selected low energy hadronic interaction model[27] (and an optional "_int" if you select the "INTTEST" option or an optional "_conex" if you selected the "CONEX" option). As a consequence, you can select different model combinations on different systems without any conflict[28].

[26] HDPM, DPMJET, EPOS, NEXUS, QGSJET, QGSII, SIBYLL, or VENUS.

[27] fluka, gheisha, or urqmd.

[28] In case of conflict, using different options but with the same models for instance, the program will ask you if you want to rename the binary file.

2.3 Data Files

2.3.1 Input Files

To run a simulation one needs to read several input files. These are:

- The Glauber tables to derive nucleon-nucleus and nucleus-nucleus cross-sections from hadron-nucleon cross-sections are listed in file *NUCNUCCS* which is 2873 lines long. They are read via logical unit NUCNUC (NUCleus-NUCleus interactions, by default 11).
- The cross-sections and branching ratios for the EGS4 routines are contained in the data files *EGSDAT6_*x.x with a length of 3021 lines each. These files differ in the lowest kinetic energy to be followed within a range between 3 MeV (x.x = 3.) and 50 keV (x.x = .05). Only one of the sets is selected by the program and connected to the logical unit KMPI (by default 12) in a manner appropriate to the lowest energy of em-particles specified by the user. As the muon nuclear interactions use the routines to treat photonuclear interactions, always an *EGSDAT6_*x.x file must be read in.
- The DPMJET routines need the Glauber tables named *GLAUBTAR*.DAT. Additionally some parameterized data contained within the data file $pomtab$.dat are read in. The latter will be generated at the first call within that directory from where you are calling CORSIKA. This calculation needs c. 20 min on DEC 3000/600 AXP with 175 MHz. Later calls will read in this data file $pomtab$.dat. This file is written and read via logical unit 37, and *GLAUBTAR*.DAT is read via unit 47.

  The binary file *NUCLEAR*.BIN with a length of 198472 bytes is read from unit 14 and fits only for computers which read data in the direct access mode according with the IEEE conventions (DEC-UNIX, LINUX) in **32-bit mode**. With `gfortran` compilers with version $\geq$ 4.2 in 64-bit mode the option `-frecord-marker=4` must be used[29]. For other machines (DEC-VAX, HP) a different binary file is needed, which you may get from CERN or from the author of DPMJET <johannes.ranft@cern.ch>.
- The EPOS routines get user-specified parameters from a scratch file via logical unit EPOPRM (by default 97) and need some parameterized data contained within the data files $epos$.inics.lhc, $epos$.iniev, $epos$.ini1b, $epos$.inirj.lhc and $epos$.initl. If the latter files are not existent or do not fit with the user-specified parameters, they are established in a time consuming procedure (some 100 h on a DEC 3000/600 AXP with 175 MHz).
- The NEXUS routines get user-specified parameters from a scratch file via logical unit NEXPRM (by default 97) and need some parameterized data contained within the data files $nexus$.inics, $nexus$.iniev, $nexus$.inirj, and $nexus$.initl. If the latter files are not existent or do not fit with the user-specified parameters, they are established in a time consuming procedure (some 100 h on a DEC 3000/600 AXP with 175 MHz).
- The more recent QGSJET-II-04 routines need some parameterized data contained within the data files *qgsdat-II-04* and *sectnu-II-04*. (The QGSJET01d routines need some parameterized data contained within the data files *QGSDAT01* and *SECTNU*). These data

[29] For the compiler option `-frecord-marker=8` a *NUCLEAR.BIN* file with a length of 198768 bytes is needed.

files are written and read via logical units 1 and 2. **Attention: The** *qgsdat-II-04* **file has a size of $\approx$ 250 MB.**

- The STACKIN option needs the parameters (type, energy, momenta) of the secondary particles coming from the interaction of an exotic primary in a separate file. Its file name is specified by the keyword INFILE (page 65). The first line of the file contains (format free after a leading blank) the number of secondaries and the primary energy (GeV). The following lines contain the current particle number, the particle type, the energy (GeV), the longitudinal momentum, and the two transverse momenta (GeV/c) in the format (2I5,4(1X,E15.7)). The momenta are relative to the direction of the (exotic) primary defined by THETAP (page 62) and PHIP (page 63).
- The VENUS option reads the file *VENUSDAT* which contains parameterized structure function integrals and is 2051 lines long. The file is read via logical unit 14.
- The FLUKA option needs various data files from the FLUKA library, so you should set an environment variable pointing to the FLUKA library[30].
- The URQMD option looks for the existence of the *tables*.dat decay width file. If this file exists, it is read in, otherwise the decay width tables are calculated and this file is created.
- The ATMEXT option needs tabulated atmosphere input data of the MODTRAN model contained in the *atmprof*`i.dat` files which are read in by the *atmo*.c routines of the 'bernlohr' package. Details are given in the comments at the beginning of *atmo*.c and in the documentation supplied with the 'bernlohr' package.
- The CEFFIC option reads the atmospheric absorption table *atmabs*.dat, photomultiplier quantum efficiency table *quanteff*.dat and/or mirror reflectivity table *mirreff*.dat via logical units MCERABS (by default 20), MCERQEF (by default 21), and MCERMIR (by default 22) respectively.
- The CONEX routine *ConexInit* reads the steering commands for the CONEX calculations from the *conex_HIGHMODEL_lowmodel.param* file via the logical unit INLUN (by default 38). This file is copied from the appropriate *conex_HIGHMODEL_lowmodel.paramin* file contained within the `conex/` subdirectory. These steering commands give the files with the parameters for the calculations of the cascade equations.
- Besides these data files CORSIKA needs the input of steering keywords to select the subject and the parameters of the simulation. They have to be supplied by the user. They are read via logical unit MONIIN (MONItor INput, by default 5). The format of the steering keywords and their effect is described in detail in Sect. 4 (page 60 ff.). Examples are given on page 101 and in the file 'inputs'.

All these files are placed in the `run/` subdirectory, where the program can be executed. Using the DATDIR keyword (see page 88) the data input files may be placed in an arbitrary directory (with exception of FLUKA data files and the steering input file with the keywords).

[30] Assuming *csh* shell one uses: `setenv FLUPRO` *flukadirectory*. For bash/sh shells you give: `export FLUPRO=`*flukadirectory*.

Logical units for in- and output			
Unit name	default	I/O	File name and file
MONIIN	5	I	*input*, steering keywords
MONIOU	6	O	simulation control output on line printer
MDEBUG	6	O	debug output if DEBUG it selected
NUCNUC	11	I	$NUCNUCCS$, nucleus-nucleus cross-sections
MPATAP	90	O	$DATnnnnnn$, particle output and simulation results
MEXST	96	I/O	external particle stack, scratch file
	1	I	$sigmapi$.bin, table for FLUKA
	1	I/O	$epos$.inics.lhc, various tables for EPOS
	1	I/O	$epos$.iniev, various tables for EPOS
	1	I/O	$epos$.ini1b, various tables for EPOS
	1	I/O	$epos$.inirj.lhc, various tables for EPOS
	1	I/O	$epos$.initl, various tables for EPOS
	1	I/O	$nexus$.inics, various tables for NEXUS
	1	I/O	$nexus$.iniev, various tables for NEXUS
	1	I/O	$nexus$.inirj, various tables for NEXUS
	1	I/O	$nexus$.initl, various tables for NEXUS
	1	I/O	*qgsdat-II-04* (rsp. *QGSDAT01*), various tables for qgsjet-II-04 (rsp. QGSJET01d)
	2	I/O	*sectnu-II-04* (rsp. *SECTNU*), nucleus-nucleus cross-sections for qgsjet-II-04 (rsp. QGSJET01d)
LUNOUT	11	O	control output of FLUKA
KMPI	12	I	*EGSDAT6*_x.x, EGS4 cross-sections
	14	I	*VENUSDAT*, structure function integrals for VENUS
NBERTP	14	I	*NUCLEAR*.BIN, nuclear data for DPMJET
	14	I	*nuclear*.bin, nuclear data for FLUKA
LUNERR	15	O	error output of FLUKA
MCERABS	20	I	*atmabs*.dat for atmospheric absorption of Cherenkov light (CEFFIC option)
MCERQEF	21	I	*quanteff*.dat for photomultiplier quantum efficiency of Cherenkov light (CEFFIC option)
MCERMIR	22	I	*mirreff*.dat for mirror reflectivity of Cherenkov light (CEFFIC option)
LSTCK	23	I	STACKIN input data file
LSTCK2	24	O	output data file of particles from first interaction
ifcx & ifch	31	O	EPOS or NEXUS check file (not opened)
ifhi	35	O	EPOS or NEXUS histo file (not opened)
IUNIT	37	I/O	$pomtab$.dat, various data for DPMJET
INLUN	38	I	$conex$_HMODEL_lmodel.param steering parameters for CONEX
MDBASE	45	O	$DATnnnnnn$.dbase (rsp. $DATnnnnnn$.info), run summary file for use in an air shower library
MTABOUT	46	O	$DATnnnnnn$.tab, table output of γ, $e^{\pm}$, and $\mu^{\pm}$
	47	I	*GLAUBTAR*.DAT, Glauber tables for DPMJET
MLONGOUT	48	O	$DATnnnnnn$.long, output of longitudinal particle numbers and energy deposit

Table 1: Logical units for in- and output with default values and file names (to be continued).

Logical units for in- and output (continued)			
Unit name	default	I/O	File name and file
ifdt	51	O	EPOS or NEXUS data file (not opened)
ifcp	52	O	EPOS or NEXUS copy file (not opened)
LUNPLT	54	O	histogram output file for INTTEST version
	55	O	$DATnnnnnn$.<spec>_<proj>.map, output of PLOTSH2
	55	O	$DATnnnnnn$.track_em, output of PLOTSH (em comp.)
	56	O	$DATnnnnnn$.track_mu, output of PLOTSH (muon comp.)
	57	O	$DATnnnnnn$.track_hd, output of PLOTSH (hadron comp.)
	75	I/O	(*tables*.dat) decay widths tables for UrQMD
	76	I/O	(*UrQMD-*$\langle$*VER*$\rangle$*-xs*.dat) total cross-section table for UrQMD
	88	O	histogram output file for ANAHIST/AUGERHIST/MUONHIST vers.
MCETAP	91	O	$CERnnnnnn$, Cherenkov photon output
MPACUT	92	I/O	cut parameters in/out for PARALLEL version
MPAINP	93	I	$steering$ file for PARALLEL version
MPAJOB	94	O	parameter output for PARALLEL version
MEXSTJ	95	I/O	stack in/out for PARALLEL version
NEXPRM	97	I/O	EPOS or NEXUS parameters, scratch file

Table 1: (continued) Logical units for in- and output with their default values and file names.

2.3.2 Output Files

There are several streams of CORSIKA output:

- One is control information about the simulation run itself. This (standard) output[31] comes via the logical unit MONIOU (MONItor OUtput, by default 6). In case of a debugging run very much information is written to the logical unit MDEBUG (Monitor for DEBUGging, by default 6). Further details on this file are given in Sect. 10.1 (page 108).

- The second output stream contains the information about all the particles that reach the observation level. It gets the file name '$DATnnnnnn$' and is written onto the output directory DSN (keyword DIRECT page 87) via output unit MPATAP (PArticle TAPe, by default 90) as a 'sequential' 'unformatted' FORTRAN file[32]. $nnnnnn$ is the run number specified in the keyword RUNNR (page 60). This output may be suppressed (see keyword DIRECT and keyword PAROUT, page 87). Further details on this file are given in Sect. 10.2 (page 110).

- For the INCLINED option with an inclined observation plane a binary data output is written to the file named '$DATnnnnnn$.inclined'. The format is identical to the normal

[31] Renaming this output to '$DATnnnnnn$.lst' and redirecting it to the directory specified by the keyword DIRECT is convenient as by the shell commands 'dir' or 'ls -l' all files belonging to one run are displayed consecutively which facilitates book-keeping.

[32] To read this data set the FORTRAN programs *corsikaread*.f, *corsikaread_history*.f rsp. *corsikaread_thin*.f available in the `src/utils/` subsubdirectory may be used.

binary output data, besides the fact that all coordinates are given within the inclined plane, with the origin at the intersection of the shower axis with the inclined observation plane. The keyword INCLIN explained in Sec. 4.62 (page 85) is used to specify the geometry of the inclined sampling plane. Further details on the output file format are given in Sect. 10.3 (page 119).

- Optionally (steered by keyword OUTFILE page 66) a file is written out via the logical unit LSTCK2 (by default 24) containing the parameters of the secondary particles emerging from the first interaction of a hadronic primary. This file may be used later in a run with the STACKIN option (see Sect. 3.5.27 page 53) to maintain the first interaction even for different random number seeds.

- Optionally a table of the number of the binned γ's, $e^{\pm}$ and $\mu^{\pm}$ particles might be written out to the file '$DATnnnnnn$.tab' onto the output directory DSN (keyword DIRECT page 87) via output unit MTABOUT (TABle OUTput, by default 46). $nnnnnn$ is the run number specified in the keyword RUNNR (page 60). To be activated by the keyword PAROUT (page 87). Further details on this file are given in Sect. 10.6 (page 120).

- The longitudinal distribution of particle numbers and energy deposits can be written out to the file '$DATnnnnnn$.long' onto the output directory DSN (keyword DIRECT page 87) via the unit MLONGOUT (LONGitudinal OUTput, by default 48). $nnnnnn$ is the run number specified in the keyword RUNNR (page 60). This output is activated by the FLONGOUT flag (see keyword LONGI page 82). Further details on this file are given in Sect. 10.5 (page 119).

- Another (optional) output file contains the compressed information of the Cherenkov photons. It gets the file name '$CERnnnnnn$' and is written onto the output directory DSN (keyword DIRECT page 87) via output unit MCETAP (CErenkov TAPe, by default 91). $nnnnnn$ is the run number specified in the keyword RUNNR (page 60).

- To establish a summary file on the contents of an air shower library an optional output file containing a run summary is written to '$DATnnnnnn$.dbase' on the directory DSN (keyword DIRECT page 87) via output unit MDBASE (Data BASE, by default 45), activated by keyword DATBAS (page 94). $nnnnnn$ is the run number specified in the keyword RUNNR (page 60). Using the AUGERINFO option the name of this file will be '$DATnnnnnn$.info' and the formats are adapted to the requirements of the Auger experiment. Further details on this file are given in Sect. 10.7 (page 121).

- The PLOTSH files contain all tracking steps for each particle with start and end point to produce demonstration plots of the development of showers. They are written onto the directory DSN via the units 55 (file '$DATnnnnnn$.track_em' for em-particles), 56 (file '$DATnnnnnn$.track_mu' for muons), and 57 (file '$DATnnnnnn$.track_hd' for hadrons) (see Sect. 3.5.22 page 50). $nnnnnn$ is the run number specified in the keyword RUNNR (page 60).

- The PLOTSH2 map files are, basically, two-dimensional histograms containing the number of tracks in each xy-/xz-/yz-bin. Details are given in Sect. 3.5.23, page 51. The map files are named '*DATnnnnnn*.<spec>_<proj>.map', where <spec> stands for 'em', 'mu', or 'hd', and <proj> stands for 'xy', 'xz', or 'yz'. They are written onto the directory DSN via the unit 55. $nnnnnn$ is the run number specified in the keyword RUNNR (page 60).

- The interaction test option INTTEST generates histograms; their data are written to the file with the name defined by keyword HISTDS (page 99) via the output unit LUNPLT (Logical UNit for PLoT, by default 52).

- The ANAHIST, AUGERHIST, and MUONHIST options generate histograms; their data are written to the file named '$datnnnnnn$.lhbook' via the output logical unit 88. $nnnnnn$ is the run number specified in the keyword RUNNR (page 60). Further details on this file are given in Sect. 10.4 (page 119).

During the calculation the program uses a temporary data set (scratch file) as an external particle stack if the internal one is over-full. This data set is connected to unit MEXST (EXternal STack, by default 96).
The names of EPOS data files $epos$.inics.lhc, $epos$.iniev, $epos$.ini1b, $epos$.inirj.lhc, and $epos$.initl may be changed using the keyword EPOPAR (page 70). The parameters given in the keyword EPOPAR are written to a scratch file connected to unit NEXPRM (by default 97). This file is read by the EPOS routines when initializing EPOS.
Similarly, the names of NEXUS data files $nexus$.inics, $nexus$.iniev, $nexus$.inirj, and $nexus$.initl may be changed using the keyword NEXPAR (page 72). The parameters given in the keyword NEXPAR are written to a scratch file connected to unit NEXPRM (**neX**US **P**a**R**a**M**eters, by default 97). This file is read by the NEXUS routines when initializing NEXUS.
Some values of the in- and output units may be redefined by changing their values in the corresponding BLOCK DATA subprograms. Table 1 lists all units together with their default values and the corresponding file names.

3 Program Options

3.1 High-Energy Hadronic Interaction Models

3.1.1 DPMJET Option

DPMJET [4] (**D**ual **P**arton **M**odel with **JET**s) is a program developed to describe high-energy hadronic interactions of hadron-nucleus and nucleus-nucleus collisions using the two-component Dual Parton Model with soft chains and multiple mini-jets at each elementary interaction. For CORSIKA the version DPMJET 2.55 is available. **Because of the** *NUCLEAR.BIN* **data file you should prefer the 32-bit mode of your CPU**.
For using DPMJET you first have to select the DPMJET option when extracting the FORTRAN code from the source file. The $Makefile$ will automatically compile[33] $dpmjet253$bc.f, $dpmjet253$c.f, $dpmjet254$.f, $dpmjet255$c.f, $dpmjet256$c.f, together in a library $libdpm$.a and then link it with the compiled $gheisha\_2002d$.f code (rsp. UrQMD library) and with your CORSIKA program. In your input file you may supply the keyword (page 69):

```
DPMJET    T   0
```

Setting DPMJET to .false. (F) the simple HDPM routines are used (see also Sect. 3.1.3 page 30).
If in your calling directory the data set '$pomtab$.dat' is not yet existent it will be calculated at the first call of subroutine $prblm2$ (of $dpmjet256$.f) (which takes c. 20 min on a DEC 3000/600 AXP with 175 MHz). The correct reading of the *NUCLEAR.BIN* file is described in page 22. The DPMJET option[34] needs about the same CPU-time as the VENUS option (NKG enabled, EGS4 disabled) at primary energies of 10^{15} eV. Technically it is possible to use DPMJET up to the highest energies.
DPMJET activates also the inelastic hadron-nucleus cross-sections at higher energies which are calculated by the subroutine $dpjsig$. Nucleus-nucleus cross-sections are derived from the DPMJET nucleon-nucleon cross-sections using the Glauber tables of CORSIKA [3]. The DPMJET cross-sections are selected automatically when the DPMJET option has been used for extracting the FORTRAN code from the source file. In your input file you may supply the keyword (page 70):

```
DPJSIG    T
```

Setting DPJSIG to .false. (F) you will use the default cross-sections of CORSIKA as described in Ref. [3].
It should be emphasized, that the DPMJET option cannot be combined with the FLUKA option because of several identical common and subroutine names used within DPMJET and FLUKA.

[33] The FORTRAN compiler options described in Sect. A.1 (page 129) are used.

[34] Experience shows that because of the complexity and the sparse internal documentation of DPMJET not all possibilities for error stops or crashes are detected and eliminated. These errors are difficult to trace back, and we are not able to support users in those cases.

3.1.2 EPOS Option

EPOS [24] (**E**nergy conserving quantum mechanical multi-scattering approach, based on **P**artons, **O**ff-shell remnants and **S**plitting parton ladders) like NEXUS combines features of the former VENUS [9] and QGSJET01 [6] with extensions enabling a safe extrapolation up to higher energies, using the universality hypothesis to treat the high energy interactions [10]. Compared to NEXUS, many technical problems have been solved and the screening effects have been simplified using a more phenomenological approach mainly based on the recent RHIC data. In addition, high density effects have been included. The most actual version is EPOS LHC (v3400) [11], in which LHC data are taken into account to constrain model parameters.
For using EPOS you first have to select the EPOS option when extracting the FORTRAN code from the source file. The $Makefile$ will compile[35] *epos*`*.f` in the $libepos$.a library and then link it with the compiled $gheisha\_2002d$.f code (rsp. FLUKA library or UrQMD library) and with your CORSIKA program. In your input file you may supply the keyword (page 70):

```
EPOS     T    0
```

Setting EPOS to .false. (`F`) the simple HDPM routines are used (see also Sect. 3.1.3 below).
The standard parameters for EPOS are set in subroutine $aaset$ of the $epos$-bas-lhc.f file. If in your calling directory the data sets $epos$.inics.lhc, $epos$.iniev, $epos$.ini1b, $epos$.inirj.lhc, and $epos$.initl are not existent or not compatible with the selected parameters, they will be calculated at the first call of subroutine $psaini$ of $epos$-sem-xxx.f (which takes some 100 h on a DEC 3000/600 AXP with 175 MHz).
Normally all parameters of EPOS are set by subroutine *aaset* (of the $epos$-bas-lhc.f file) called from subroutine *nexini*. In special cases it may be necessary to overwrite one or more of these parameters or to rename the data files to identify $epos$.inixx files established for different parameter sets. This is performed using the keyword

```
EPOPAR        aaaaaaaaaa
```

in the input file (see footnote to keyword EPOPAR, Sect. 4.28 page 70). *aaaaaaaaaa* is a command line as described in the EPOS documentation. These *aaaaaaaaaa* commands are written onto the file connected with the logical unit NEXPRM (by default 97) and read by subroutine *aread* of the *epos*-`bas-lhc.f` file.
The EPOS option needs roughly 7.5 times more CPU-time than the VENUS option ($E_0 = 10^{15}$ eV, NKG enabled, EGS4 disabled).
EPOS activates also the inelastic hadron-nucleus interaction cross-sections at higher energies. They are calculated by the subroutine *nexsig*. The EPOS cross-sections are selected automatically when the EPOS option has been used for extracting the FORTRAN code from the source file. In your input file you may supply the keyword (page 71):

```
EPOSIG     T
```

[35] The FORTRAN compiler options described in Sect. A.1 (page 129) are used.

Setting EPOSIG to .false. (F) you will use the default cross-sections of CORSIKA as described in Ref. [3].
The *all-inputs-epos* in `run/` subdirectory is an example input file to run CORSIKA with EPOS.

3.1.3 HDPM Routines

HDPM is a set of routines to simulate high-energy hadronic interactions. These routines are fast and adjusted to experimental data where available. Proton-proton interactions simulated with HDPM and other models agree fairly well with each other (see Ref. [25]). Experimental data are however rare for high energy nucleon-nucleus or nucleus-nucleus collisions and here the results start to disagree due to the simpler modeling in HDPM. If one is interested in differences of air showers induced by different nuclei one probably gets more realistic results by the detailed simulation with other models (DPMJET, EPOS, NEXUS, QGSJET, SIBYLL, VENUS) than with the HDPM routines.
As the HDPM routines are default you have nothing to specify when extracting the FORTRAN code from the source file. But the compiled $gheisha\_2002d$.f code (rsp. FLUKA library or UrQMD library) will be linked with your CORSIKA program.

3.1.4 NEXUS Option

neXus [10] (**NEX**t generation of **U**nified **S**cattering approach) combines features of the former VENUS [9] and QGSJET [6] with extensions enabling a safe extrapolation up to higher energies, using the universality hypothesis to treat the high energy interactions [10]. It handles nucleus-nucleus collisions with an up to date theoretical approach. The most actual version is NEXUS 3.97.
For using NEXUS you first have to select the NEXUS option when extracting the FORTRAN code from the source file. The $Makefile$ will compile[36] *nexus*-`xxx.f` in the $libnexus$.a library and then link it with the compiled $gheisha\_2002d$.f code (rsp. FLUKA library or UrQMD library) and with your CORSIKA program. In your input file you may supply the keyword (page 71):

```
NEXUS     T    0
```

Setting NEXUS to .false. (F) the simple HDPM routines are used (see also Sect. 3.1.3 above).
The standard parameters for NEXUS are set in subroutine $aaset$ of the $nexus$-bas.f file. If in your calling directory the data sets $nexus$.inics, $nexus$.inicsei, $nexus$.inidi, $nexus$.iniev, $nexus$.inigrv, $nexus$.inirj, $nexus$.inirjei, and $nexus$.initl are not existent or not compatible with the selected parameters, they will be calculated at the first call of subroutine $psaini$ of $nexus$-sem.f (which takes some 10 h on a DEC 3000/600 AXP with 175 MHz).
Normally all parameters of NEXUS are set by subroutine *aaset* (of the $nexus$-bas.f file) called from subroutine *nexini*. In special cases it may be necessary to overwrite one or more of these parameters or to rename the data files to identify $nexus$.inixx files established for different parameter sets. This is performed using the keyword

[36] The FORTRAN compiler options described in Sect. A.1 (page 130) are used.

```
NEXPAR          aaaaaaaaaa
```

in the input file (see footnote to keyword NEXPAR, page 72). *aaaaaaaaaa* is a command line as described in the NEXUS documentation. These *aaaaaaaaaa* commands are written onto the file connected with the logical unit NEXPRM (by default 97) and read by subroutine *aread* of the *nexus*-`bas.f` file.
The NEXUS option needs roughly 7.5 times more CPU-time than the VENUS option ($E_0 = 10^{15}$ eV, NKG enabled, EGS4 disabled).
NEXUS activates also the inelastic hadron-nucleus interaction cross-sections at higher energies. They are calculated by the subroutine *nexsig*. The NEXUS cross-sections are selected automatically when the NEXUS option has been used for extracting the FORTRAN code from the source file. In your input file you may supply the keyword (page 72):

```
NEXSIG     T
```

Setting NEXSIG to .false. (`F`) you will use the default cross-sections of CORSIKA as described in Ref. [3].
The *all-inputs-nexus* in `run/` subdirectory is an example input file to run CORSIKA with NEXUS.

3.1.5 QGSJET Option

QGSJET [6, 12] (**Q**uark **G**luon **S**tring model with **JET**s) is a program developed to describe high-energy hadronic interactions using the quasi-eikonal Pomeron parameterization for the elastic hadron-nucleon scattering amplitude. The hadronization process is treated in the quark gluon string model. The most actual version is QGSJET-II-04 [12] including Pomeron loop and the cross-section is tune to LHC data.
For using QGSJET you first have to select the QGSJET or QGSII options when extracting the FORTRAN code from the source file. Without QGSII you will extract the link routines for the older QGSJET01d program.
The $Makefile$ will link the compiled $qgsjet$-II-04.f (rsp. $qgsjet01d$.f) and $gheisha\_2002d$.f codes (rsp. FLUKA library or UrQMD library) with your CORSIKA program. The $qgsjet$-II-04.f package will behave differently compared with the older $qgsjet01d$.f.
In your input file you may supply the keyword (page 72):

```
QGSJET     T   0
```

Setting QGSJET to .false. (`F`) the simple HDPM routines are used (see also Sect. 3.1.3 page 30).
If in your calling directory the data sets *qgsdat-II-04*[37] and *sectnu-II-04* (rsp. $QGSDAT01$ and $SECTNU$) are not yet existent they will be calculated at the first call of subroutine `gqaini` (rsp. `psaini`) (which takes c. **4 days** on a 1 GHz Pentium LINUX rsp. 30 h for *QGSDAT01* on a DEC 3000/600 AXP with 175 MHz). The resulting *qgsdat-II-04* file will have a **size of** $\approx$ **131**

[37] Binary type file is available for LINUX systems on the CORSIKA ftp server for downloading.

MB. The QGSJET option needs about 3 times more (qgsjet-II-04) rsp. the same (qgsjet01d) CPU-time than/as the HDPM option (NKG enabled, EGS4 disabled).
QGSJET activates also the inelastic hadron-nucleus interaction cross-sections at higher energies which are supplied in the *qgsdat-II-04* file read in by the qgsjet-II-04 [12] (rsp. the *QGSDAT01 file* read in by the qgsjet01d [6]) program package. The nucleus-nucleus cross-sections are contained in the file *sectnu-II-04* (rsp. *SECTNU*). The qgsjet-II-04 cross-sections (rsp. QGSJET01d cross-sections[38]) are selected automatically when the QGSJET option has been used for extracting the FORTRAN code from the source file. In your input file you may supply the keyword (page 73):

```
QGSSIG    T
```

Setting QGSSIG to .false. you will use the default cross-sections of CORSIKA as described in Ref. [3].

3.1.6 SIBYLL Option

SIBYLL [7] is a program developed to simulate hadronic interactions at extreme high energies based on the QCD mini-jet model. The actual [8] version is SIBYLL 2.1.
For using SIBYLL you first have to select the SIBYLL option when extracting the FORTRAN code from the source file. The $Makefile$ will link the (compiled) $sibyll2.1$.f and $gheisha\_2002d$.f codes (rsp. FLUKA library or UrQMD library) with your CORSIKA program. In your input file you may supply the keyword (page 73):

```
SIBYLL    T   0
```

Setting SIBYLL to .false. (F) the simple HDPM routines are used (see also Sect. 3.1.3 page 30). The SIBYLL option needs about the same CPU-time as the HDPM option (NKG enabled, EGS4 disabled).
SIBYLL activates also the inelastic hadronic interaction cross-sections at higher energies which are supplied with the SIBYLL [7] program package. They are based on QCD calculations, details are given in [7]. SIBYLL also delivers nucleus-nucleus cross-sections. The SIBYLL cross-sections are selected automatically when the SIBYLL option has been used for extracting the FORTRAN code from the source file.
In your input file you may supply the keyword (page 73):

```
SIBSIG    T
```

Setting SIBSIG to .false. (F) you will use the default cross-sections of CORSIKA as described in Ref. [3].

[38] Omitting the default QGSJETOLD selection uses hadron-air cross-sections increased by 3 % to take into account the individual nuclear radii of ^{14}N and ^{16}O as stated in Ref. [31].

3.1.7 VENUS Option

VENUS [9] (**V**ery **E**nergetic **NU**clear **S**cattering) is a program developed to simulate ultra-relativistic heavy ion collisions. The actual version is VENUS 4.12.
For using VENUS you first have to select the VENUS option when extracting the FORTRAN code from the source file. The $Makefile$ will link the compiled $venus$.f and $gheisha\_2002d$.f codes (rsp. FLUKA library or UrQMD library) with your CORSIKA program. In your input file you may supply the keyword (page 73):

```
VENUS    T   0
```

Setting VENUS to .false. (F) the simple HDPM routines are used (see also Sect. 3.1.3 page 30). Normally all parameters for VENUS are supplied by the routine `venini`. In special cases it may be necessary to overwrite one or more of these parameters specified by its name PARCHA and its new value PARVAL. This is performed using the keyword (page 74)

```
VENPAR     PARCHA    PARVAL
```

in the input file (page 74).
The VENUS option needs roughly 15 times more CPU-time than the HDPM option (NKG enabled, EGS4 disabled).
VENUS activates also the inelastic hadron-nucleus interaction cross-sections at higher energies which are calculated by the subroutine `vensig`. Nucleus-nucleus cross-sections are derived from the VENUS nucleon-nucleon cross-sections using the Glauber tables of CORSIKA [3]. The VENUS cross-sections are selected automatically when the VENUS option has been used for extracting the FORTRAN code from the source file. In your input file you may supply the keyword (page 74):

```
VENSIG     T
```

Setting VENSIG to .false. you will use the default cross-sections of CORSIKA as described in Ref. [3].

3.2 Low-Energy Hadronic Interaction Models

3.2.1 FLUKA Option

FLUKA (**FLU**ctuating **KA**scade) [13] is a package of routines to follow energetic particles through matter by the Monte Carlo method. In combination with CORSIKA only that part is used which describes the low-energy hadronic interactions. A detailed description of the processes simulated by FLUKA 2011 may be found on the FLUKA web page :
`http://www.fluka.org/` .
FLUKA is used within CORSIKA to calculate the inelastic hadron cross-sections with the components of air and to perform their interaction and secondary particle production, including many details of the de-excitation of the target nucleus.

If you have selected the FLUKA option, the $Makefile$ will link the FLUKA library[39] with your CORSIKA program (pages 130 and 132). Please verify that the compiler (gortran rsp. g77 in 32 or 64 bit mode) which your system uses for the CORSIKA program fits with that employed to compile the available FLUKA package.
To run the FLUKA version, an environment variable[40] tells the system where to find the binary data files (page 23) needed by the FLUKA routines.
It should be emphasized, that the FLUKA option cannot be combined with the DPMJET option because of several identical subroutine names used within FLUKA and DPMJET.

3.2.2 GHEISHA Option

GHEISHA (**G**amma **H**adron **E**lectron **I**nteraction **SH**ower code) is an interaction package widely used in the detector Monte Carlo program GEANT [32] that has proven its qualities in describing hadronic collisions up to some 100 GeV in many experiments. A detailed description of the physics processes covered by GHEISHA may be found in Ref. [14]. The GHEISHA version is taken as distributed in October 17, 1994 with the GEANT package [32] version 3.21/03 by CERN. Recently some errors were eliminated using fixes obtained from SLAC [33] and now all variables are used in double precision. To discriminate against the uncorrected single-precision GHEISHA version it is renamed to $gheisha\_2002d$.f. GHEISHA is used in CORSIKA to calculate the elastic and inelastic cross-sections of hadrons below 80 GeV in air and their interaction and particle production.
The $Makefile$ will link the compiled $gheisha\_2002d$.f code with your CORSIKA program, if you have selected the GHEISHA option.

3.2.3 URQMD Option

UrQMD (**U**ltra-**r**elativistic **Q**uantum **M**olecular **D**ynamics) is an interaction package designed to treat low energy hadron-nucleus interactions. A detailed description of this model may be found in Ref. [15]. UrQMD 1.3_cors is used in CORSIKA to perform the elastic and inelastic interactions of hadrons below 80 GeV in air. The actual linking routines operate with the special UrQMD 1.3_cors version adapted to CORSIKA.
For using UrQMD you first have to select the URQMD option when extracting the FORTRAN code from the source file. For compilation of the *compilefile*.f the UrQMD 1.3_cors include files $boxinc$.f, $colltab$.f, $comres$.f, $coms$.f, $inputs$.f, $newpart$.f, and $options$.f must be available. So the *./coconut* script will install these files if needed before doing a $Makefile$ to make a UrQMD library $liburqmd$.a from the UrQMD 1.3_cors FORTRAN files and then link it with your CORSIKA program (pages 130 and 132). In your input file you may supply the keyword (page 96):

```
URQMD     T    0
```

Setting the first parameter FURQMD to .false. (F) the program will stop.

[39] See footnote page 10 for the different available FLUKA libraries.

[40] Assuming csh shell one uses: `setenv FLUPRO` *flukadirectory*. For bash/sh shells you give: `export FLUPRO=`*flukadirectory*.

3.3 Electromagnetic Interactions (NKG/EGS4 Option)

The NKG and EGS4 options are selected by flags of the input file keyword ELMFLG only. A detailed description of the EGS4 program can be found in Ref. [16], and the modifications applied to it are published in Ref. [3].
For using NKG and/or EGS4 you have to activate the flags of the keyword (page 76)

```
ELMFLG    T   T
```

in the input file.

It must be emphasized that at the highest electron and γ-energies above 10^{17} eV the NKG option does not contain the Landau-Pomeranchuk-Migdal effect (which is added to EGS4) which may alter the shower development by the decrease of the pair formation and bremsstrahlung cross-sections with increasing energy. Therefore the analytical NKG treatment deviates more and more from results gained by the EGS4 option. For example in a γ-induced shower (10^{19} eV energy, inclined with $\theta = 60^{\circ}$, without PRESHOWER option) the maximum of the electron longitudinal development simulated by EGS4 is reached deeper in the atmosphere by ≈ 100 g/cm^2 slant depth than predicted by NKG.

3.3.1 NKG Treatment

The first flag activates the analytic NKG treatment of the electromagnetic component. The longitudinal electron numbers and pseudo-age parameters[41] are calculated every 100 g/cm^2 above the lowest observation level and the lateral electron densities are calculated for a radial grid of 80 points at the two lowest observation levels. The inner grid radius is fixed at 100 cm, while the outer radius RADNKG (in cm) is selected by the keyword (page 76):

```
RADNKG    200.E2
```

Also lateral pseudo-age parameters[41] are calculated.
As the NKG formulas do not take into account the curvature of the Earth's surface, for the CURVED option the NKG option is suppressed. As with the COMPACT option the NKG parameters cannot be written out onto the particle file, the NKG flag must be disabled in this case.

3.3.2 EGS4 Treatment

The second flag of the keyword ELMFLG activates the full Monte Carlo treatment of the electromagnetic component by the EGS4 package. Both flags may be activated or deactivated independently. No special option for extracting the FORTRAN code from the source file is required. By selecting the CERENKOV option the EGS4 routines are activated automatically.
In most applications (especially Cherenkov radiation from showers induced by primaries with energies in the GeV range) an abbreviated treatment of the multiple scattering of electrons

[41] These pseudo-age parameters should only be used qualitatively. For scientific applications you extract better age parameters from a fit to the lateral distribution of the electrons as simulated by the EGS4 option.

within the EGS4 code is not recommended. If you can afford a lower quality of your simulations but no long computing times, you may specify an enlarged step length factor STEPFC by the keyword (page 76):

```
STEPFC    1.0
```

(See also the comments in Sect. 4.44 page 76.) A detailed discussion on the use of this step length is given in [34].
In the standard version treating pair production and bremsstrahlung, the EGS4 routines do not regard the Landau-Pomeranchuk-Migdal (LPM) effect which should be applied at energies above $E_{lab} > 10^{16}$ eV. The LPM-effect is switched on automatically using the THIN option (see Sect. 4.51 page 79) or the LPM option (see Sect. 3.5.16 page 48).
The files named *EGSDAT6*_x.x replace the files *EGSDAT5*_x.x, *EGSDAT4*_x.x, *EGSDAT3*_x.x, *EGSDAT2*_x.x, or *EGSDATA* used in older CORSIKA versions. For the extrapolation to the highest energies the photo-nuclear cross-section is extrapolated according to Cudell et al. [35] published by the Particle Data Group. The low energy threshold of these files ranges from 0.05 MeV to 3 MeV. They differ from the older data sets by the arrangements of the tables containing the $e^{\pm}$-branching ratios and γ-branching ratios, thus giving a more smooth branching ratio for the rare processes of electro-nuclear and photo-nuclear interactions rsp. of $\mu^+\mu^-$ pair formation [36]. A data set with an energy threshold far below ELCUT(4) implies the explicit, but unnecessary production of many bremsstrahlung photons above threshold but below ELCUT(4), resulting in a considerable prolongation of wasted CPU-time. Therefore CORSIKA automatically selects the *EGSDAT6*_x.x set best suited for the user's specification of the ELCUT(3) and ELCUT(4), thus saving CPU-time.

3.4 Cherenkov Options

3.4.1 Cherenkov Standard Option

The routines treating the Cherenkov radiation have been supplied by the HEGRA Collaboration [37] and considerably improved by K. Bernlöhr [38]. The Cherenkov light production by electrons, positrons, muons, and charged hadrons is considered in the subroutine `cerenk`. The Cherenkov photons are considered within a wavelength band which may be specified by the lower and upper limits WAVLGL and WAVLGU. Atmospheric absorption of the Cherenkov photons is not taken into account by default, but might be added by the CEFFIC option (see Sect. 3.4.5 page 39). Only Cherenkov photons arriving at the lowest observation level are recorded.
Charged particles create Cherenkov photons at each tracking step when the condition $\beta > 1/n$ ($\beta = v/c$ and $n =$ refractive index) is fulfilled. The step is subdivided into smaller sub-steps such that the number of Cherenkov photons per sub-step is less than the fixed number CERSIZ, predefined by an input keyword. In such a sub-step all the photons are sent in a compact bunch along a straight line, defined by the emission angle θ_C relative to the electron or hadron direction and a random value for the angle ϕ around this direction.
As the major part of the Cherenkov light is produced by electrons it makes no sense to simulate showers with Cherenkov light production unless using the EGS4 option. Therefore the

CERENKOV option automatically activates the EGS4 option, too. The CERENKOV option reduces the step length factor STEPFC to 1 by default (page 76).
For higher primary energies it is impossible to write all the photon bunches of one shower to the output file. Therefore, only those bunches are recorded which hit an array at the lowest observation level consisting of NCERX $\times$ NCERY photon detectors arranged with a grid spacing of DCERX and DCERY cm in x and y direction respectively and with ACERX $\times$ ACERY cm^2 area each. Each bunch is represented by 7 words which are the number of Cherenkov photons, the x and y position coordinates at the observation level, direction cosines u and v, arrival time, and height of production above sea level.
To obtain this program version the CERENKOV option has to be selected when extracting the FORTRAN code from the source file. Via the keyword (page 89)

```
CERARY    27   27   1500.   1500.   100. 100.
```

the geometry of your Cherenkov array may be defined. A rotation of the Cherenkov array x-axis relative to North may be respected by the keyword (page 86):

```
ARRANG    0.
```

The bunch size may be selected by the keyword (page 90):

```
CERSIZ    0.
```

The optimal choice of the bunch size depends on the employment of the atmospheric absorption, mirror reflectivity, and photomultiplier quantum efficiency (CEFFIC option, see Sect. 3.4.5 below). Without the CEFFIC option a CERSIZ = 5 is reasonable, as about one photon of such a bunch survives in an off-line treatment of these effects.
By the keyword (page 90)

```
CERFIL    T
```

the Cherenkov output is directed to the separate Cherenkov output file MCETAP or to the particle output file MPATAP. In case of a separate output file the Cherenkov output is structured as the particle output file. It contains the event header and the event end block and in between the data blocks. The data structure of the Cherenkov output data set is given in Table 11 (page 115). In the case the Cherenkov bunches are stored together with the other shower particles on the same particle output file, a Cherenkov bunch is treated like a particle.
The definition of an array of Cherenkov detectors serves to reduce the required disk space for Cherenkov shower. On the other hand one loses the possibility of using an air shower several times during the analysis with different core locations with respect to the detector. Keeping in mind the excessive computation time for Cherenkov showers a possibility is introduced to use Cherenkov showers multiple times with only a tolerable increase of storage space. Therefore, already during the simulation it is defined how often a single shower should be used and where in the array the core locations should be. The core locations for each event are chosen with the Sobol quasi-random number generator [39] and are stored in the event header. Correspondingly, the array of Cherenkov detectors is placed several times in the observation plane and store all Cherenkov bunches that hit one of the detectors. This possibility is selected by the keyword (page 91):

```
CSCAT    ICERML   XSCATT   YSCATT
```

An event is used ICERML times and the core is scattered in the range -XSCATT $\leq x_{core} \leq$ XSCATT and -YSCATT $\leq y_{core} \leq$ YSCATT. For the analysis of such CORSIKA events the user has to use the same core locations in the analysis that have been determined during the simulation. The output will basically scale with the number of times each event is used, but it is still considerably smaller than the output of the complete Cherenkov component would be.
To obtain this program version the CERENKOV option has to be selected when extracting the FORTRAN code from the source file.

3.4.2 Cherenkov Wavelength Option

In the CERWLEN option the index of refraction is made wavelength dependent. As a consequence, photon bunches will carry a specific wavelength. Photons of shorter wavelengths (with larger index of refraction) will result in larger Cherenkov cone opening angles and larger bunch sizes. For very fast particles this will generally have a small effect (less than 0.03° in the opening angle, for example) but near the Cherenkov threshold the effect can be larger.
This option may also require to use a smaller maximum bunch size (see keyword CERSIZ page 90) since all photons in a bunch are of the same wavelength and, therefore, the peak quantum efficiency rather than the average quantum efficiency determines the maximum acceptable bunch size. (In combination with the CEFFIC option (see Sect. 3.4.5 page 39) you should use a maximum bunch size of 1, as usual.)
To obtain this program version the CERWLEN option has to be selected in combination with the CERENKOV option when extracting the FORTRAN code from the source file.

3.4.3 Imaging Atmospheric Cherenkov Telescope Option

The routines treating the Cherenkov radiation for **I**maging **A**tmospheric **C**herenkov **T**elescopes (`IACT` option) have been supplied by K. Bernlöhr [38]. The Cherenkov light production by electrons, positrons, muons, and charged hadrons is considered in the subroutine $cerenk$. The positions of the telescopes are defined by the keyword (page 91)

```
TELESCOPE    0.  0.   0.   0.
```

giving the coordinates relative to the center of the lowest observation level (see Sect. 4.81 page 91). The data set name for the telescope-specific data output is defined by the keyword (page 92):

```
TELFIL    filename
```

For further details of the IACT option see Ref. [38], the comments at the beginning of the *iact*.c routines, and the documentation supplied with the 'bernlohr' package.
With the IACT option by default the TMARGIN flag (keyword TSTART) is set to .true. (see Sect. 4.12 page 64), but it may be overridden. This affects the x and y coordinates of the Cherenkov photons arriving at the observation level if the altitude of the first interaction is fixed by the keyword FIXHEI (page 64).

To obtain this program version the IACT option has to be selected together with the CERENKOV option when extracting the FORTRAN code from the source file. The $Makefile$ will compile the needed **C** files of the 'bernlohr' package in the $libbern$.a library and then link it with your CORSIKA program.

3.4.4 Imaging Atmospheric Cherenkov Telescope Extension Option

With the IACTEXT option the interface to the *telout* function (*iact*.c routines) is extended by parameters describing the emitting particle. This extended information is stored as an additional photon bunch (after the normal one) with mass, charge, energy, and emission time replacing the `cx`, `cy`, `photons`, and `zem` fields, respectively, and are identified by a wavelength of 9999. The compact output format is disabled for making that possible. In addition, all particles arriving at the observation level are included in the `eventio` format output file, in a photon-bunch like block identified by array and detector numbers 999.
The `x`, `y`, `cx`, `cy`, and `ctime` fields keep the normal sense, with coordinates, directions and time counted in the detection level reference frame. The particle momentum is filled into the `zem` field (negative for upward-moving particles) and the particle ID is filled into the `lambda` field. If thinning is used, the particle weight is in the `photons` field.
When compiling *iact*.c manually (instead of taking advantage of the *./coconut* script or the GNU-makefile supplied with the 'bernlohr' package), an additional option `-DIACTEXT` is required to have a consistent interface on both sides.
To obtain this program version the IACTEXT option has to be selected together with the CERENKOV and IACT options when extracting the FORTRAN code from the source file.

3.4.5 Cherenkov Light Reduction Option

The standard simulation of Cherenkov photons does not regard light absorption within the atmosphere, telescope mirror reflectivity, or quantum efficiency of the detecting photomultiplier tubes. In the CEFFIC option these effects are taken into account at an early stage of the Cherenkov photon simulation, thus the computing time is shortened considerably and the requirements for storage of Cherenkov output are reduced additionally.
Data tables containing information on these three effects as function of photon wavelength are needed in this option depending on the status of the flags specified by the keyword (page 90):

```
CERQEF     F    F    F
```

Standard tables for atmospheric absorption (*atmabs*.dat), mirror reflectivity (*mirreff*.dat) (measured for the re-coated mirrors of the Whipple telescope, Sept. 1993), and quantum efficiency (*quanteff*.dat) (measured for Hamatsu R1398HA photo-multipliers with UV-window and 1.125" tube) are supplied with CORSIKA. For other installations the user should establish corresponding tables[42].

[42]The *atmabs*.dat table is composed of 105 wavelength values between 180 and 700 nm in steps of 5 nm; one line for each wavelength, beginning with the wavelength value [nm] as integer, followed by 51 extinction values, starting at sea level up to 50 km height in steps of 1 km. The data format is (105(I4,5(10F10.3),F10.3)).

By early eliminating those Cherenkov photons which are absorbed within the atmosphere, not reflected by the mirror, or not producing photo electrons within the photomultiplier, those suppressed photons are also not counted in the various forms of the longitudinal distributions (see Sect. 3.4.6 below).
As in the CEFFIC option the atmospheric absorption is treated only in a manner suited for planar atmospheres, you should not combine the CEFFIC option with CURVED.
To obtain this program version the CEFFIC option has to be selected together with the CERENKOV option when extracting the FORTRAN code from the source file.

3.4.6 INTCLONG and NOCLONG Options

In the Cherenkov version the longitudinal distribution of photons is given in differential mode (i.e. the number of photons generated within each step) by default. By the preprocessor option INTCLONG the integral mode is selected (i.e. accumulated number of generated Cherenkov photons for each step) which needs additional computing time. If both kinds of longitudinal distribution are of no interest, you may deselect the Cherenkov photon distribution completely by the preprocessor option NOCLONG thus saving computing time.
The option INTCLONG is effective also with the AUGCERLONG option (see Sect. 3.5.3 page 41).
To obtain these program versions the INTCLONG rsp. NOCLONG option has to be selected together with the CERENKOV rsp. AUGCERLONG option when extracting the FORTRAN code from the source file.

3.5 Other Non-standard Options

3.5.1 ANAHIST Option

The ANAHIST option produces a series of histograms generated with HBOOK routines [29]. The histograms are written into the file named '$datnnnnnn$.lhbook' (page 119) onto the output directory DSN specified by the keyword DIRECT (page 87). To suppress the ordinary particle output file the keyword PAROUT (page 87) might be used. This analysis gives a short overview on various shower properties of the particles arriving at the observation level. The histograms are only established for the lowest observation level. Radial thinning is not applied to the particles sorted into the histograms.
Because of its permanent modifications a comprehensive description of the ANAHIST option is not available.
To obtain this program version the ANAHIST option has to be selected when extracting the FORTRAN code from the source file. The HBOOK routines require linking of the CERN library[43] with the program; they are not supplied with the CORSIKA package.

For the same 105 wavelengths the *mirreff*.dat and *quanteff*.dat tables contain reflectivity rsp. quantum efficiency values written in the format (8F6.3). Further details may be taken from the comments in the employed subroutine `tpdini`.

[43] The CERN library is only available for 32bit computers.

3.5.2 ATMEXT Option with External Atmospheres

The ATMEXT option allows to use external tabulated atmospheres of the MODTRAN model documented in Ref. [40]. They are provided together with the 'bernlohr' package as files *atmprof*`i.dat` and read in with special routines written in **C**. Further details may be found in Ref. [38], at the beginning of the *atmo*.c routines and in the documentation supplied with the 'bernlohr' package.
The CURVED option needs the atmospheres in the 5-layer model rather than interpolated from ATMEXT tables. Therefore in the CURVED option AATM, BATM, CATM values are fitted to the tabulated atmospheres and the table interpolation is disabled.
To obtain this program version the ATMEXT option has to be selected when extracting the FORTRAN code from the source file. A linking with the (compiled) *atmo*.c routines of the 'bernlohr' package is done by the $Makefile$.

3.5.3 AUGCERLONG Option

With the AUGCERLONG option it is possible to fill the Cherenkov column in the table of the longitudinal particle distribution without full simulation of the Cherenkov radiation. Because of an elongation of the CPU time the AUGCERLONG option should only be used if the longitudinal Cherenkov distribution is really needed. By the INTCLONG option (page 40) the integrated instead of the differential longitudinal Cherenkov intensity may be selected. The keywords CERSIZ (page 90) and CWAVLG (page 90) are activated with the AUGCERLONG option.
To obtain this program version the AUGCERLONG option has to be selected when extracting the FORTRAN code from the source file.

3.5.4 AUGERHIST Option

The AUGERHIST option produces a series of histograms generated with HBOOK routines [29]. The histograms are written into the file named '$datnnnnnn$.lhbook' (page 119) onto the output directory DSN specified by the keyword DIRECT (page 87). As the HBOOK routines use only lower case characters, the DSN directory name should not contain capitals. These histograms show properties of different particle types at up to 20 horizontal levels (to be defined by keyword OBSLEV, page 84). They enable to study the longitudinal development of various shower parameters. At each defined level a series of histograms is generated e.g. for radial distances of different particle species, for the energy deposit by different particle species as function of distance from the shower axis, for energy spectra (as function of distance) and much more. Examples are given in Ref. [41]. For histograms relating to the emission of Cherenkov radiation the bunch size and wavelength band has to be specified using the keywords CERSIZ (page 90) and CWAVLG (page 90).
Only particles arriving at the lowest observation level are directed to the particle output file (page 110) and/or the table output file (keyword PAROUT page 87) and/or binned into the ANAHIST histograms.

Because of its permanent modifications a comprehensive description of the AUGERHIST option is not available.
The AUGERHIST option works only in combination with the THIN option. To obtain this program version the AUGERHIST option has to be selected when extracting the FORTRAN code from the source file. The HBOOK routines require linking of the CERN library[43] with the program; they are not supplied with the CORSIKA package.

3.5.5 AUGERINFO Option

The AUGERINFO option produces an output file named '$DATnnnnnn$.info' replacing the '$DATnnnnnn$.dbase' file, when activated by the DATBAS keyword (page 94). The output format of the .info file differs from that of the .dbase file to enable the automatic production of a data base for the Auger experiment showing the content of the CORSIKA shower library at the IN2P3 computing center Lyon.
To obtain this program version the AUGERINFO option has to be selected when extracting the FORTRAN code from the source file.

3.5.6 CHARM Option

In the standard CORSIKA program the charmed particles are treated implicitly in the hadronic interaction codes. But several codes (at present DPMJET and QGSJET01d) produce them in a manner that they can be treated with explicitely transporting them respecting their cross-sections calculated by [42] and their lifetimes using the CHARM option [43].
With the SIGMAQ keyword (page 97) the cross-sections for the interaction of the charmed (rsp. bottom) particles are determined. The PROPAQ keyword (page 97) determines whether the interactions are handled by routines [42] which extend the PYTHIA 6.411 package[44], or in combination with QGSJET01d by the routines of $qgsjet01d$.f[45]. The decay of charmed particles is treated by the PYTHIA package [30]. Therefore in combination with QGSJET01d the modified and extended PYTHIA 6.411 package is installed in the `pythia/` subdirectory for automatic compilation and linking. In combination with DPMJET the implicit PYTHIA 6.102 package (= $dpmjet253c$.f) with extensions [42](= $dpmjet253bc$.f) is used and needs no extra linking.
The CHARM option [43] also enables the treatment of the τ-leptons. All possible interactions (bremsstrahlung, pair production, nuclear interactions) of the τ-leptons are respected in a manner analogously to the μ-leptons. For the treatment of primary ν_τ and $\overline{\nu_\tau}$ neutrinos the CHARM option has to be combined with the NUPRIM option (page 49). For tracing back PYTHIA problems in combination with QGSJET01d the PYTHIA keyword is available (page 96).

[43] The CERN library is only available for 32bit computers.

[44] The PYTHIA routines $pyspli$.f and $pypdfu$.f are modified and the routines $chabadif$.f, $chabapar$.f, $chamedif$.f, $chamepar$.f, $bobadif$.f, $bobapar$.f, $bomedfif$.f, and $bomepar$.f are added to the PYTHIA 6.411 package.

[45] Before compiling $qgsjet01d$.f the probability of charmed quark rsp. di-quark formation should be enabled by setting the appropriate values of DC(3) and DC(5) in subroutine XXASET.

To obtain this program version the CHARM option has to be selected when extracting the FORTRAN code from the source file. Please verify that in combination with QGSJET01d you i) have activated the charm production in the $qgsjet01d$.f file and ii) have avaliable the PYTHIA 6.411 package with all modifications and extensions[44].

3.5.7 COASTUSERLIB Option

The COASTUSERLIB option selects **C**++-routines[46] which must be provided by the user before starting *./coconut* by defining the environment variable `$COAST_USER_LIB` pointing to an existing user-defined libCOAST.so library. In this case CORSIKA will transmit the path elements of all tracked particles to the user-defined code. The user-defined code can then be used for many purposes, such as the visualization of particle tracks or the generation of histograms.
The COASTUSERLIB option[47] cannot be combined with the COMPACT option or any other option making use of the COAST interface, i.e. COREAS, INCLINED, or ROOTOUT options, nor iwith histogram producing options ANAHIST, AUGERHIST, or MUONHIST.
To obtain this program version the COASTUSERLIB option has to be selected when extracting the FORTRAN code from the source file and *root* must be installed on your system.

3.5.8 COMPACT Output Option

The standard output of CORSIKA is not adequate when simulating a large number of showers initiated by primaries of so low energies, that only a small percentage of them produces particles arriving at the detector level. As most Data blocks of the MPATAP file would be filled up with zeros, a large amount of useless information for the Data blocks and the unnecessary overhead of the Event Header and Event End blocks would be written in this case. This is avoided in the COMPACT option, which writes out only the Run Header and the full Event Header for the first event. For subsequent events only shortened Event Headers (the first 12 parameters) are written. Event End blocks are omitted completely, the Run End block is written as usual. The Data blocks have a maximal length of 39 particles, trailing zeros are suppressed. Further details are given in Sect. 10.2.3 (page 118).
As the NKG parameters are not written out in the COMPACT version, the NKG flag (keyword ELMFLG page 76) should be disabled. The COMPACT option cannot be combined with the COASTUSERLIB, COREAS, INCLINED, or ROOTOUT options.
To obtain this program version the COMPACT option has to be selected when extracting the FORTRAN code from the source file.

3.5.9 CONEX Option for Cascade Equations

Warning: Despite a lot of successful tests, the authors couldn't test all possible combinations of models and observable types. For the moment the precision expected by the use of the CONEX option should be better than 10% compared to usual CORSIKA for a factor of 10 gain in CPU

[46] Through the enclosed COAST package.
[47] This option compiles only on LINUX and Mac-OSX machines.

time but it may depend on the initial conditions and observables. As a consequence they decline any responsability if an analysis is biased by the use of this option. The user should validate his analysis by the use of CORSIKA without CONEX option for particle based analysis. 1D simulations based on CONEX are already used intensively since many years. The original CONEX program [21, 22] treats the development of showers by numerically solving cascade equations (CE) for the different particle species (hadrons, muons, and electromagnetic particles).

In the CONEX-CORSIKA coupling [44, 45] which resembles the procedure of the SENECA code [46] the shower development starts with a Monte Carlo treatment of the primary particle and also the secondary particles are handled by Monte Carlo techniques as long as their energy E exceeds the preselected value E_{thr}. The secondaries with energies $E < E_{thr}$ are binned into the energy-depth tables which define the "source terms" of the CE. From these source terms the further shower evolution is calculated by solving the CE down to the energy E_{low} which marks the end of the CE treatment. The parameters needed for solving these CE are transmited from the parameter sets which are selected by the steering file *conex_HIGHMODEL_lowmodel.param* (see page 23) in dependence of the selected hadronic interaction model. The solution of the CE can be sampled into individual particles which are saved on the CORSIKA particle stack (hadrons and muons) rsp. on the CORSIKA-EGS4 stack (electromagnetic particles). At this point to each particle with $E < E_{low}$ a weight can be attributed which is used in the THIN option (see Sect. 4.51 page 79) to treat the bulk of the low-energy particles by the standard Monte Carlo method (with thinning) until these particles reach the observation level. For an equivalent precision level a minimum speed-up by a factor 5 can be expected using CONEX instead of standard Monte Carlo methods (including optimized thinning).
The parameters needed for the CONEX code are derived from the settings for the standard CORSIKA Monte Carlo procedure (including the THIN option) and are completed by default values which cover all normal applications. Only in special cases these standard settings may be modified by experts using the keywords CASCADE, CONEX, CX2COR, and CXWMX (Sects. 4.46 - 4.49, pages 77 - 78).
The CONEX option is only avaliable for the high-energy hadronic interaction models EPOS, QGSJET, QGSII, or SIBYLL in combination with all low-energy interaction models. If the CONEX option is selected and none of the special keywords are used, the simulation will include the use of CE at intermediate energy and the weights are defined by the usual THIN, THINH and THINEM keywords.
The CONEX option cannot be combined with any of the CERENKOV options (Sect. 3.4.1 to Sects. 3.4.6, pages 36 - 40).
To obtain this program version the CONEX option has to be selected together with the CURVED +SLANT+THIN+UPWARD options when extracting the FORTRAN code from the source file.

3.5.10 COREAS Option

The COREAS option activates the CoREAS code [47] for the calculation of radio emission from extensive air showers. The emission is calculated by applying the "endpoint formalism" [48] to each individual path element tracked in CORSIKA. CoREAS is the successor of REAS [49] and

is not based on any particular model for the generation mechanism of the radio emission (except of course classical electrodynamics). Please refer to the separate user manual for CoREAS, which is automatically generated when compiling CORSIKA with the COREAS option active. The COREAS option[48] cannot be combined with the COMPACT option or the histogram producing options ANAHIST, AUGERHIST, or MUONHIST.
To obtain this program version the COREAS option has to be selected when extracting the FORTRAN code from the source file.

3.5.11 CURVED Atmosphere Option

The standard CORSIKA program models the Earth's atmosphere as a flat disc where the density of the air decreases with the height. The shower calculations and tracking are using Cartesian coordinates. In a flat atmosphere the thickness increases with $1/\cos\theta$. This is a good approximation for inclined showers if their zenith angles are below $\approx 70^\circ$. Above this value the differences between a flat and a curved atmosphere become more and more important. At 90° eventually the thickness of the flat atmosphere becomes infinite whereas the correct thickness is ≈ 37000 g/cm^2.
Within the CURVED atmosphere option for large zenith angles above $\approx 70^\circ$ the Earth's atmosphere is no longer assumed to be completely flat (as in the standard version for smaller zenith angles). Rather the atmosphere is replaced by a 'sliding plane atmosphere'. Each times the horizontal displacement of a particle exceeds a limit of 6 to 20 km (depending on altitude), a transition to a new, locally plane atmosphere is performed. By these means the advantages of the simpler transport formulas within a planar atmosphere are combined with the faster simulation speed by avoiding the lengthy and more complicated treatment by using a true spherical system. Because of technical reasons only one observation level may be specified in the CURVED option and the X and Y coordinates have a special definition in order to be able to reconstruct the 3 coordinates (x,y,z) in a cartesian frame (see Sect. 4.63 page 85).
In the CURVED option the ionization energy loss, deflection within the Earth's magnetic field, and the generation of Cherenkov photons is enabled for charged hadronic primaries on their path between entering the atmosphere and the first interaction. (The deflection is disabled in the standard version without using keyword TSTART, page 64.) The arrival time refers to the start at the margin of the atmosphere, which is indicated by a negative value of element 7 of the event header block (page 113).
The NKG formulas do not take into account the curvature of the Earth's surface. Therefore the NKG output is suppressed in the CURVED option.
The combination of the CURVED option with the UPWARD option is described in Sect. 3.5.31 (page 57).
As in the CEFFIC option the atmospheric absorption is treated only in a manner suited for planar atmospheres, you should not combine the CURVED option with CEFFIC.
The CURVED option needs the atmospheres in the 5-layer model rather than interpolated from ATMEXT tables. Therefore in the CURVED option AATM, BATM, CATM values are fitted to the tabulated atmospheres and the table interpolation is disabled.

[48] This option compiles only on LINUX and Mac-OSX machines.

To obtain this program version the CURVED option has to be selected when extracting the FORTRAN code from the source file.

3.5.12 EFIELD Option

In the EFIELD option the presence of atmospheric electrical fields acts on the movement of the electrons (rsp. positrons) of the shower. Depending on the direction of the electrostatic field relative to the movement of the electromagnetic particles the electrons (rsp. positrons) are deflected, accelerated, or slowed down. Because of the much higher mass and therefore the much higher stiffness of muons and hadrons the influence on the movement of those particles is disregarded. The shape and functional form of the electrostatic field in the atmosphere must be specified by the user in the C-subroutine $elfield$.c. An $elfield$.c template (with zero electrical field) is available in the `src/` subdirectory. The used variables for input and output with their definitions are explained in the comment lines of this subroutine. Further details on the EFIELD option may be found in Ref. [50].

The EFIELD option should only be used if the electromagnetic shower component is treated with the EGS4 routines. The EFIELD option has no effect on the calculation of the electromagnetic shower component with the analytic NKG formulas.

To obtain this program version the EFIELD option has to be selected when extracting the FORTRAN code from the source file.

3.5.13 EHISTORY Option

With this option additional information on the prehistory of muons, neutrinos, and electromagnetic (EM) particles is given out to the particle output file. This extended additional information is activated for the muons by the keyword MUADDI (see page 83) and for the neutrinos by the keyword NUADDI (see page 84). For EM particles it is activated with the keyword EMADDI (see page 83). The precursor [49] hadrons 'grandmother' and 'mother' of the particles at ground are specified with the penetrated matter between start and end of grandmother's track, the position of mother's birth, and the momenta of grandmother and mother at that point. The generation counter of the mother particle is also given, which might be compared to the generation counter of the muon thus indicating additional decays between mother particle and the muon, which otherwise are not given. Further details are described in [51].

The FORTRAN program *corsikaread_history*.f available in the subsubdirectory `src/utils/` is designed to read binary particle output files produced with the EHISTORY option.

To obtain this program version the EHISTORY option has to be selected when extracting the FORTRAN code from the source file.

[49] For dpmjet2.55 which handles internal decays by PYTHIA the direct parent particle of the decay leading to the final muon rsp. neutrino is taken as mother particle.

3.5.14 INCLINED Observation Plane Option

In the INCLINED option the horizontal observation plane (by default) is replaced by an inclined observation plane. The keyword INCLIN explained in Sec. 4.62 (page 85) is used to specify the geometry of the inclined sampling plane. The output is written as a binary data file to the file named '$DATnnnnnn$.inclined' (see Sect 2.3.2 page 25). The format is identical to the normal binary output data, besides the fact that all coordinates are given within the inclined plane, with the origin at the intersection of the shower axis with the inclined observation plane. Further details on the output file format are given in Sect. 10.3 (page 119).
Technically, the INCLINED option uses a particular COASTUSERLIB library (Sec. 3.5.7 page 43), namely `InclinedPlane`, which is shipped with COAST in the `CorsikaOptions` subdirectory by default.
The INCLINED option[50] cannot be combined with the COASTUSERLIB, COMPACT, COREAS, or ROOTOUT options.
To obtain this program version the INCLINED option has to be selected in *./coconut* when extracting the FORTRAN code from the source file.

3.5.15 INTTEST Interaction Test Option

With this option the interaction model routines can be tested which are used to describe the hadronic collisions. Only the first interactions are simulated and no air showers are developed. Various projectiles (keyword PRMPAR) and targets (keyword INTTST) may be selected. The result of the test is a series of histograms generated with the HBOOK routines [29] and written to the unit HISTDS. The histograms show properties of the secondary particles produced in the first interaction e.g. distributions of transverse momenta, of squared transverse momenta, of longitudinal momenta, of Feynman x-distributions, of rapidity and pseudo-rapidity distributions, and various particle multiplicity distributions.
To obtain this program version the INTTEST option has to be selected when extracting the FORTRAN code from the source file. The HBOOK routines require linking of the CERN library[51] with the program; they are not supplied with the CORSIKA package. Via the keywords with their parameters (pages 97 - 99)

```
INTTST    ITTAR   MCM

INTDEC    LPI0   LETA   LHYP   LK0S

INTSPC    LSPEC

DIFOFF    NDIF

TRIGGER    NTRIG
```

[50] This option compiles only on LINUX and Mac-OSX machines.
[51] The CERN library is only available for 32bit computers.

the conditions of the interaction test run have to be specified.
Because of its permanent modification a comprehensive description of the INTTEST option is not available. It should be noted that the combinations of allowed parameters for the INTTEST option differ from the standard version without explicitly noting this or checking this during the program run.
To obtain this program version the INTTEST option has to be selected when extracting the FORTRAN code from the source file.

3.5.16 LPM Option

The LPM option switches on the **L**andau-**P**omeranchuk-**M**igdal effect (which is added to EGS4) causing an effective reduction of the pair production and bremsstrahlung cross-sections [52] at the highest energies. With this option it is possible to include the LPM-effect without selecting the THIN option (see Sect. 3.5.29 page 54).

3.5.17 MUONHIST Option

The MUONHIST option produces a series of histograms generated with HBOOK routines [29]. The histogram give various properties (energy, transverse momentum, slant depth) of the muons at the position of their origin. An example of the application of this option is given in Ref. [53]. The histograms are written into the file named '$datnnnnnn$.lhbook' (page 119) onto the output directory DSN specified by the keyword DIRECT (page 87). To suppress the ordinary particle output file the keyword PAROUT (page 87) might be used.
To obtain this program version the MUONHIST option has to be selected when extracting the FORTRAN code from the source file. The HBOOK routines require linking of the CERN library[51] with the program; they are not supplied with the CORSIKA package.

3.5.18 MUPROD Option

With this option additional information on the prehistory of muons is given out to the particle output file even if those muons don't reach the observation level. To get this information the MUPROD option has to be selected and activated by the MUADDI keyword (see Sect. 4.59 page 83). This information is written to the particle output file, where the first entry with particle code 85 (for μ^+) rsp. 86 (for μ^-) represents the muon with its momentum and position components at the production point. To get more information the EHISTORY option (see Sect. 3.5.13 page 46) can be combined with the MUPROD option. In the following entries the precursor hadrons 'grandmother' and 'mother' of the decaying muon are specified with the penetrated matter between start and end of grandmother's track, the position of mother's birth, and the momenta of grandmother and mother at that point. The generation counter of the mother particle is also given, which might be compared to the generation counter of the muon thus indicating additional decays between mother particle and the muon, which otherwise are not given. The fourth entry with particle code 95 (for μ^+) rsp. 96 (for μ^-) represents the muon at its decay rsp.

interaction point with its momentum and position components. Further details are described in [51].

3.5.19 NEUTRINO Option

Muonic and electronic neutrinos and anti-neutrinos originate from the decays of $\pi^{\pm}$, the leptonic decays of $K^{\pm}$ and K°_{L}, and the decays of $\mu^{\pm}$. The neutrino formation is simulated with exact kinematics of all two and three body decays taking into account the polarization of the muons. The neutrino trajectories are followed down through the atmosphere disregarding any interaction with the target nuclei of the air. The neutrinos are written to the particle output file using the particle type numbers 66 to 69, if their energy exceeds the cutoff energy ELCUT(1).
To obtain this program version the NEUTRINO option has to be selected when extracting the FORTRAN code from the source file.

3.5.20 NUPRIM Option for Primary Neutrinos

With this program version showers induced by primary neutrinos can be simulated. Possible primary particles are (up to now) ν_e, $\overline{\nu_e}$, ν_μ, and $\overline{\nu_\mu}$. If combining the NUPRIM option with the CHARM or TAULEP option also the ν_τ and the $\overline{\nu_\tau}$ neutrinos can be handled.
Because of the low cross-sections of neutrino-induced interactions it is recommended to fix the height of the primary interaction using the keyword FIXHEI (page 64). As the first (neutrino-induced) interaction is handled by the HERWIG code [18], a link [19] with the HERWIG routines is necessary. The type of interaction (charged current or neutral current) may be selected by the keyword NUSLCT (Sect. 4.41 page 75). HERWIG produces - besides others - secondary particles (with charm). Those particles can be treated by CORSIKA only if you combine the NUPRIM option with the CHARM option (see page 42). Without specifying the CHARM option those charmed secondaries decay at the vertex and are not explicitely transported in CORSIKA. Specifying the TAULEP option the tau leptons emerging from CC-interactions of ν_τ or $\overline{\nu_\tau}$ neutrinos are treated explicitely in CORSIKA. The interactions of secondary particles coming from the primary neutrino reaction are treated by the selected high-energy rsp. low-energy hadronic interaction model or by NKG/EGS4. Secondary neutrinos are explicitly generated only, if the NEUTRINO option (page 49) is selected, but they are not treated by HERWIG. Initiating a shower with non-neutrino primary particles will not call the HERWIG routines.
Sequence 5 of the CORSIKA random generator is foreseen for HERWIG, therefore you should initialize it (see Sect. 4.3 page 61).
For using the NUPRIM version you first have to select the NUPRIM option when extracting the FORTRAN code from the source file. The *Makefile* will compile the HERWIG FORTRAN routines and link them with your CORSIKA program.
A combination of the NUPRIM option with the INTTEST option is not possible.

3.5.21 PARALLEL and PARALLELIB Options

The PARALLEL option enables the CORSIKA program to run a single shower on several cores in parallel to reduce the time to complete the shower [23]. There exist two modes of paralleliza-

tion: a) by shell scripts or b) by **M**essage **P**assing **I**nterface (MPI) system. They are available after having selected the option `"p - PARALLEL treatment of subshowers"` in the additional CORSIKA program options of *./coconut*. Then you have to select the mode of parallelization by answering the following question:

```
Compile CORSIKA as subroutine for parallelization with MPI ?
 1 - Special stack for shell scripts without MPI [DEFAULT]
 2 - Library to be used with MPI system
```

In the first case a set of shell scripts handles the external stack with the secondary particles and starts the subshower processes. These shell scripts are available in the subdirectory `src/parallel/` together with a short description (*userguide-pll-**.txt).
To obtain this program version the PARALLEL option has to be selected when extracting the FORTRAN code from the source file.

In the second case the PARALLELIB option is activated and the parallelization is handled by MPI. In this case the program *mpi_runner*.c is compiled and linked with the subroutine *corsika-compilefile*.f . The first interactions are simulated on a core (MASTER) and produce secondary particles which are used to start subshowers in parallel on many cores (SLAVES) to be run simultaneously. The communication between the MASTER and the SLAVES is established by MPI. Further technical details are explained in the MPI-Runner_GUIDE [54] available in the subdirectory `doc/`.
To obtain this program version the PARALLEL and PARALLELIB options have to be selected when extracting the FORTRAN code from the source file.

3.5.22 PLOTSH Shower Plot Production Option

In the PLOTSH option the start and end points of each particle track are written to extra files (see Table 1 page 24); separate files are used for the electromagnetic, muonic, and hadronic component. Each track is written unformatted as one block consisting of 10 real*4 numbers: Particle identification, Energy (in GeV), Xstart, Ystart, Zstart (all in cm), Tstart (in sec), Xend, Yend, Zend (all in cm), and Tend (in sec). In case of THINning the particle Weight is added at the end of each block.
Plots may be constructed from the content of these files by using the *plottracks* program. The program *plottracks3c*.f (available in `src/utils/`) converts the particle track maps generated by the PLOTSH option into PPM image files (readable by *xv*). *plottracks* is automatically compiled[52] when the PLOTSH option is selected.
The program *plottracks* reads in the '*DATnnnnnn*.track_em', '*DATnnnnnn*.track_mu', and '*DATnnnnnn*.track_hd' files specified on the command line by the parameter run=$nnnnnn$ (integer between 0 and 999999). Then it converts and combines them into 4 PPM images '*tracknnnnnn*.em.ppm', '*tracknnnnnn*.mu.ppm', '*tracknnnnnn*.had.ppm', and '*tracknnnnnn*.all-.ppm' of the same resolution as the maps. By default, the electromagnetic, muonic, and hadronic maps are taken as the red, green, and blue channels of the RGB image, respectively.

[52] `g77 plottracks3c.f -o plottracks`

As with increasing shower energy and decreasing threshold the number of tracks increases drastically you should simulate not more than 1 shower at a time to keep the output on the units 55, 56, and 57 at a tolerable size.
Via the keyword (page 99)

```
PLOTSH     T
```

the PLOTSH option is enabled or disabled.
This option is not recommended for ordinary shower production because of the large output files to be produced.
To obtain this program version the PLOTSH option has to be selected when extracting the FORTRAN code from the source file.

3.5.23 PLOTSH2 Shower Plot Production Option

In the PLOTSH2 option air showers are visualized in a simple way, without having to deal with a (very large) track file. With the PLOTSH2 option map files are generated separately for the electromagnetic particles, muons, and hadrons. For each particle species maps are generated in all three projections (x-y, x-z, and y-z). The map files are written onto the directory DSN via the unit 55 and are named '*DATnnnnnn.*<spec>_<proj>.map', where <spec> stands for 'em', 'mu', or 'hd', and <proj> stands for 'xy', 'xz', or 'yz'. $nnnnnn$ is the run number specified in the keyword RUNNR (see page 60). These map files are, basically, two-dimensional histograms containing the number of tracks in each xy-/xz-/yz-bin. The resolution of the maps is set via the three integer parameters IXRES, IYRES, and IZRES in the COMMON /CRPLOTSH2/. The unformatted map files consist of a two-word header containing the horizontal and vertical resolution of the respective map file, followed by the rows of map values[53]. These files are then easily processed further, for example by *map2png* (see below).
To control the plotting, the keyword PLAXES followed by 6 parameters (see page 100) might be used in the input file:

```
PLAXES    -5.E5  5.E5  -5.E5  5.E5  0.  3.E6
```

The keyword

```
PLCUTS    0.3  0.3  0.003  0.003 1.E5 T
```

(see page 100) serves to define energy cuts in the same order as those for the keyword ECUTS (hadrons, muons, electrons, photons) (see page 81). An additional parameter defines an upper bound on the time (in ns) passed since the first interaction, while the final logical determines whether only track segments inside the three-dimensional box given by the axis ranges should be plotted.
Via the keyword

```
PLOTSH     T
```

[53] Note that on LINUX systems, the header and each row are preceded and followed by an additional word.

the PLOTSH2 option is enabled or disabled (see page 99).

This option is not recommended for ordinary shower production.

To obtain this program version the PLOTSH2 option has to be selected when extracting the FORTRAN code from the source file.

The **C**-program *map2png* (available in `src/utils/`) converts the particle track maps generated by the PLOTSH2 option into PNG image files. The only prerequisite needed by the program is *libpng*, which is included in virtually all recent LINUX distributions. If this library is not available for your system, please use PLOTSH option instead. *map2png* is automatically compiled[54] when the PLOTSH2 option is selected.

The program *map2png* reads in the map files specified on the command line (via the path to *DATnnnnnn*) and combines them into a single PNG image of the same resolution as the maps. By default, the electromagnetic, muonic, and hadronic maps are taken as the red, green, and blue channels of the RGB image, respectively. However, it is possible to specify other colors to be used for the different particle types. The projection to be used as well as whether to use a linear or logarithmic color scale can be specified on the command line. To see a list of options, simply run *map2png* without arguments.

3.5.24 PRESHOWER Option

The PRESHOWER option selects code and **C**-routines [55] to describe the pair production of ultra-high energetic ($E_0 > 10^{19}$ eV) primary photons and bremsstrahlung interactions of the secondary em-particles within the Earth's magnetic field before reaching the top of atmosphere. Within the atmosphere the resulting swarm of em-particles is treated as one shower. To model correctly the field components of the Earth's magnetic dipole field in the outer space additional input is required to specify the location and the time of the experiment. The keyword (page 69)

```
GCOORD    -69.585  -35.463   2003.   1   0
```

gives the longitudinal and lateral position of the experiment on the Earth's globe, the year (the Earth's magnetic field is time dependent), a print indicator, and a stop indicator (for the case no pre-showering occurred). Details of this input are given in page 69. With the stop indicator it is possible to skip those events where no pre-showering occurred. Nevertheless the event header and event end sub-blocks are written to MPATAP.

The appropriate **C**-routines of the preshower2.0 version belonging to the PRESHOWER option are available in the `src/preshower2.0-rev/` subdirectory, compiled and linked by the $Makefile$.

To obtain this program version the PRESHOWER option has to be selected when extracting the FORTRAN code from the source file.

[54] `gcc [-Lpath_to_libpng] -lpng -lm map2png.c -o map2png`
The bracketed option is only needed if the path to 'libpng.so' is not included in the LD_LIBRARY_PATH variable.

3.5.25 ROOTOUT Option

The ROOTOUT option selects code which directly transmits the particle output (normally directed to MPATAP) to **C**++-routines[55] to write an output '$DATnnnnnn$.root' file in *root* format, e.g. for *root* off-line analysis of the particle output data without storing the huge particle output data file MPATAP. The Cherenkov output file MCETAP is not affected by the ROOTOUT selection.
The default COAST ROOTOUT data format is thought for fast and easy off-line analysis of CORSIKA data and is NOT a general replacement for CORSIKA binary data files. Due to the chosen data structure one shower including all particles and all Cherenkov photons is stored in the computer's memory entirely before it is written to disk. This needs a lot of memory for high energy and/or high quality showers.
The ROOTOUT option[56] cannot be combined with the COMPACT option.
To obtain this program version the ROOTOUT option has to be selected when extracting the FORTRAN code from the source file and *root* (see `root.cern.ch`) must be installed on your system.

3.5.26 SLANT Option

With the SLANT option the longitudinal distributions (page 82 and Sect. 10.5 page 119) are given in slant depth bins along the shower axis instead of vertical depth bins used in the standard case. This slant depth scale is more appropriate to investigations of very inclined showers.
In the SLANT option [56, 57] the ionization energy loss, deflection within the Earth's magnetic field, and the generation of Cherenkov photons is enabled for charged hadronic primaries on their path between entering the atmosphere and the first interaction. (The deflection is disabled in the standard version without using keyword TSTART, page 64.) The arrival time refers to the margin of the atmosphere, which is indicated by a negative value of element 7 of the event header block (page 113).
To obtain this program version the SLANT option has to be selected when extracting the FORTRAN code from the source file.

3.5.27 STACKIN Option

With the STACKIN option the parameters of secondary particles will be read into the CORSIKA stack. Thus interactions of very exotic primaries may be treated off-line in a suitable interaction program avoiding a direct coupling of such programs with CORSIKA. The air shower generated by these resulting secondary particles is simulated in CORSIKA, and all options may be combined with STACKIN. To characterize the altitude of the first interaction the keyword FIXHEI[57] (page 64) must be used, the shower axis is defined by the angles THETAP and PHIP

[55] Through the enclosed COAST package.
[56] This option requires a C++ compiler and compiles only on LINUX and Mac-OSX machines.
[57] With the keyword FIXHEI the parameter N1STTR defining the type of the first target should be set to 0 (random selection) as it acts on that secondary particle which is treated first.

(pages 62 and 63). This externally treated first interaction starts the clock by default (the keyword TSTART (page 64) is disabled). The *all-inputs-st* file in the `run/` subdirectory is an example input file for a run with STACKIN option.
To run a series of showers with fixed first hadronic interaction the keyword OUTFILE (page 66) may be used (in a run without STACKIN option) to write out the parameters of the secondary particles to a file (see page 26) which is read-in later with the STACKIN option thus keeping the first interaction independent of the used random seed.
The file containing the parameters of the particles has to be specified by the keyword INFILE (page 65) and is read in via the logical unit LSTCK (by default 23, page 23). The first line is read with free format (with a leading blank character) and contains the number of secondaries and the primary energy. The following lines are read with the format (2I5,4(1X,E15.7)) containing current particle number, particle type (see Table 4 page 104), total energy (GeV), longitudinal momentum (GeV/c), and transverse momenta (GeV/c). The momenta are taken relative to the direction of the shower axis (direction of the exotic particle). The *all-inputs-stackin* file in the `run/` subdirectory is an example input file.
To obtain this program version the STACKIN option has to be selected when extracting the FORTRAN code from the source file.

3.5.28 TAULEP Option

In the standard CORSIKA program the τ-leptons cannot be treated. The TAULEP option [43] (as well as the CHARM option) enables the treatment of the τ-leptons, their decays are treated by the PYTHIA package [30]. Therefore (except in combination with the DPMJET high-energy interaction code) the PYTHIA 6.411 package in the `pythia/` subdirectory is automaticly compiled and linked. In combination with DPMJET the implicit PYTHIA 6.102 package is used and needs no extra linking. The TAULEP option may be combined with all high-energy interaction models.
All possible interactions (bremsstrahlung, pair production, nuclear interactions) of the τ-leptons are respected in a manner analogously to the μ-leptons. For the treatment of primary ν_τ and $\overline{\nu_\tau}$ neutrinos the TAULEP (or CHARM) option has to be combined with the NUPRIM option (see page 49). For tracing back PYTHIA problems the PYTHIA keyword is available (page 96).
To obtain this program version the TAULEP option has to be selected when extracting the FORTRAN code from the source file. Please verify that the PYTHIA 6.411 package in the `pythia/` subdirectory is correctly compiled and linked.

3.5.29 Option for Thinning

For primary energies $E_0 > 10^{16}$ eV the computing times become excessively long (they scale roughly with the primary energy). To reduce the times to tolerable durations the so-called 'thin sampling' mechanism (also named 'variance reduction' [16]) is introduced [52]. When thinning is active all particles below the adjustable fraction of the primary energy (thinning level $\varepsilon_{th} = E/E_0$) which emerge from an interaction are exposed to the thinning algorithm.

Only one of these particles is followed and an appropriate weight is given to it, while the other particles below the thinning level are dropped. Details on this formalism may be found in Refs. [16, 52, 58].
A further improvement [59] to reduce undesired statistical fluctuations of particle densities far from the shower core uses a limitation of the weights. Particles emerging from an interaction which would exceed a specified weight limit are excluded from the thinning algorithm. Using different weight limits for em-particles and hadronic (including muonic) particles enables a drastic reduction of computing time, if the user's interest is focused onto a precise lateral distribution of muons on the expenses of larger fluctuations of the em-part.
A third algorithm to save space on disks reduces the number of particles close to the shower core where anyway the detectors will saturate. Particles arriving at the detector level within a selected core distance $rmax$ are selected at random with a probability $prob \propto (r/rmax)^4$ and, when surviving[58], their weight factor is multiplied with the inverse of this probability, irrespective of exceeding the weight limit. This radial thinning is not effective for the table output '$DATnnnnnn$.tab' (see keyword PAROUT page 87 and Sect. 10.6 page 120), nor for the '$datnnnnnn$.lhbook file generated in the ANAHIST, AUGERHIST and/or MUONHIST versions.
Via the keyword (page 79)

```
THIN    EFRCTHN    WMAX    RMAX
```

you may specify the energy fraction EFRCTHN of the primary energy, below which the thinning process becomes active. Above this energy no thinning will take place. WMAX gives the maximum weight factor[59], which should not be exceeded. The core distance up to which the radial thinning at detector level takes place is specified by RMAX.
Via the keyword (page 80)

```
THINH    THINRAT    WEITRAT
```

or the keyword (page 80)

```
THINEM    THINRAT    WEITRAT
```

differing thin levels and weight limits can be specified for hadronic or electromagnetic particles overwriting the ratios

THINRAT = $\varepsilon_{th_{em}}/\varepsilon_{th_{hadr}}$ and WEITRAT = $\mathrm{WMAX}_{em}/\mathrm{WMAX}_{hadr}$

which are set to 1 by default.
The effect of various thinning levels $\varepsilon_{th} = E/E_0$ on the computing time and the number of particles on MPATAP may be seen from Table 2 which is established with default CORSIKA parameters (QGSJET option, EGS4 activated) for vertical proton induced showers of 10^{15} eV primary energy (DEC 3000/600 AXP with 175 MHz), with $\varepsilon_{th_{em}}/\varepsilon_{th_{hadr}} = 1$, with infinite

[58] To ensure the surviving of enough particles close to the shower axis RMAX should be chosen not too large.

[59] In case of a primary energy spectrum (ULIMIT $>$ LLIMIT) the value of WMAX is used for the low energy end of the energy interval and slides to higher values according with the sliding primary energy.

ε_{th}	none	10^{-6}	10^{-5}	10^{-4}	10^{-3}
Time (min)	98	51	7.2	1.2	0.16
particles	413078	58313	11466	2211	419

Table 2: Computing times and number of particles for various thinning levels, without application of weight limits and radial thinning.

weight limit, and without radial thinning. As to each particle an appropriate weight must be attributed, the output format described in Sect. 10.2 (page 110 ff.) has to be changed to incorporate this additional parameter of each particle. Consequently MPATAP and MCETAP **output data generated with the THIN option differ** from those generated in simulations without this option (see also Sect. 10.2.2 page 118).
An optimum choice of the various thinning parameters depends on the information which should be drawn from the simulations. To minimize the additional fluctuations (caused by the thin sampling algorithm) for muonic particle densities at large distances > 300 m from the shower core - which is one of the problems of the Auger experiment - a suitable setting [59] of WEITRAT would be

$$\text{WEITRAT} \approx 100$$

while the choice of WMAX is optimized for the primary energy E_0 (given in GeV) and the selected thinning level EFRCTHN for em-particles to

$$\text{WMAX} = \text{EFRCTHN} \cdot E_0 \,.$$

The *all-inputs-thin* file in the `run/` subdirectory is an example input file for a run with THIN option.
To obtain this program version the THIN option has to be selected when extracting the FORTRAN code from the source file.

3.5.30 TRAJECT Option

The TRAJECT option selects the zenith and azimuth angles in a manner that a gamma (or neutrino) source is followed along its trajectory in the sky. The angles selected by this option override the angle selection according to the VIEWCONE (see Sect. 3.5.32 page 57), VOLUMEDET, and VOLUMECORR (see Sect. 3.5.33 page 58) options or selected by the keywords THETAP (page 62) and PHIP (page 63). This option is activated/disabled by the keyword TRAFLG (page 92). The keywords SRCPOS (page 92), TRATM (page 93), TLAT (page 93), and TLONG (page 93) specify the position of the source in equatorial coordinates, the time information on the observation, and the latitude and longitude of the observing telescope on the globe. The total observation time given by the parameter TDURATION of the keyword TRATM (page 93) specifies the span of the trajectory. This span contains the number of events defined by the keyword NSHOW (page 61).
As the CORSIKA coordinate system refers to magnetic North, the magnetic declination at the

telesope site must be specified by the keyword GEODEC (page 94) for a correct transformation from the equatorial coordinates to the CORSIKA coordinate system. For extended sources (like a nebula) moving in the sky a broader trajectory may be followed by specifying a spread radius with the keyword TRARAD (page 94).
A detailed description of the TRAJECT option is given in Ref. [60] which gives also an illustrative example.
To obtain this program version the TRAJECT option has to be selected when extracting the FORTRAN code from the source file.

3.5.31 UPWARD Option

The UPWARD option selects code which treats the upward traveling particles. For primary particles the zenith angle is restricted to $0^\circ < \theta < 70^\circ$ and $110^\circ < \theta < 180^\circ$.
No additional keyword has to be specified.
The UPWARD option might be combined with the CURVED option (page 45). This enables to start showers with arbitrary zenith angles $0^\circ < \theta < 180^\circ$ and secondary particles with arbitrary zenith angles are followed. [57].
For showers with skimming incidence ($\mathrm{zenith\ angle} = 90^\circ$) the minimum altitude of the shower axis above sea level is specified by HIMPACT(i) (keyword IMPACT, page 65) and defines the geometry completely. The zenith angle at the entrance into the atmosphere is calculated with it. The angles THETPR(i) (see keyword THETAP page 62) have no meaning and are overridden. The keyword IMPACT may be combined with the keywords FIXHEI (page 64) or FIXCHI (page 64) to start the shower before reaching the minimum altitude of the shower axis.
For showers with zenith angles $> 90^\circ$ (e.g. initiated by neutrinos, page 49) the starting point of the shower rsp. the first interaction must be defined by the keywords FIXHEI (page 64) or FIXCHI (page 64); in this case the observation level (page 84) must be chosen preferentially at the top of the atmosphere, but at minimum above the starting point of the shower. (The shower axis must cross the observation level.)
To obtain the UPWARD program version the UPWARD option has to be selected when extracting the FORTRAN code from the source file.

3.5.32 Viewing Cone Option

The VIEWCONE option enables the generation of showers within the viewing cone of e.g. a Cherenkov telescope. Around the fixed incidence angle defined by THETPR(1) and PHIPR(1) (page 62) a (hollow) cone is defined with its tip pointing towards the detector. The inner and outer limiting angles of this cone are defined by the keyword VIEWCONE (page 63). The zenith angular dependence of the chosen detector geometry (see Sect. 3.5.33 below) is maintained for flat horizontal rsp. spherical detectors, while the VOLUMECORR option is not supported. Showers originating from those portions of the cone which exceed the allowed range of CORSIKA are not simulated, rather they are skipped and a new angle is selected at random out of the range of the cone.

To obtain this program version the VIEWCONE option has to be selected when extracting the FORTRAN code from the source file.

3.5.33 Volume Detector and Vertical String Geometry Options

With this options it is possible to select at random the zenith angle in a manner which respects the geometrical acceptance of the detector.
The **default** primary intensity distribution I goes with the zenith angle θ like

$$I \propto sin\theta \cdot cos\theta$$

The sin term respects the solid angle element of the sky, while the cos term takes the geometrical efficiency of a flat horizontal detector into account[60]. This allows to use each shower several times with the shower axis intersecting the detector array with equal distribution in x and y at random. The area to be covered by randomly scattering the shower axis is independent from the zenith angle and extends horizontally.
Using the VOLUMEDET option, the primary intensity distribution I goes with the zenith angle like

$$I \propto sin\theta$$

respecting only the solid angle elements of the sky. This is appropriate for detectors of approximated spherical geometry, e.g. atmospheric Cherenkov telescopes. To use a shower several times you might scatter it on an area, which has fixed extensions in a plane perpendicular to the shower axis.
Using the VOLUMECORR option, the primary intensity distribution is a more complicated function of the zenith angle, which respects the geometry of a long vertical string detector (e.g. AMANDA [61] or ICECUBE [62] experiment and other neutrino telescopes) with a ratio of $l/d =$*length/diameter* of the sensitive volume. The functional form of the zenith angle distribution becomes

$$I \propto (d/2)^2 \cdot \pi \cdot sin\theta \cdot (cos\theta + 4/\pi \cdot l/d \cdot sin\theta) \quad .$$

The *l/d* ratio (defining the DETector ConFiGuration) has to be read in using the keyword DETCFG (page 86).
The VOLUMECORR option cannot be combined with the VIEWCONE option.
To obtain these program versions the VOLUMEDET rsp. VOLUMECORR option has to be selected when extracting the FORTRAN code from the source file.

3.6 Combination of Options

In principle most options may be combined. Do not combine FLUKA with the present version DPMJET 2.55 because of severe clashes caused by duplicate names of COMMONS and subroutines. DPMJET, EPOS, NEXUS, QGSJET, and SIBYLL are tested only with UNIX work

[60] Selecting in the CURVED version the zenith angle θ at random one should keep in mind that for the default version the probability vanishes at $\theta = 90°$. If the zenith angle range is specified as $\theta_1 < \theta < \theta_2$ with $\theta_1 < 90°$ and $90° < \theta_2$ the zenith angle is selected at random from MIN$(\theta_1, 180° - \theta_2) < \theta <$ MAX$(\theta_1, 180° - \theta_2)$.

stations. You can not select more than one high-energy and one low-energy hadronic interaction model at a time. In principle a combination of NUPRIM with all hadronic interaction models is possible (page 131).
The CONEX option can only be used with the high-energy interaction models EPOS, QGSJET, QGSII, or SIBYLL in combination with any low-energy interaction model.
The CHARM option makes sense only with the interaction models DPMJET or QGSJET01d (with EPOS in preparation), as only these interaction models are producing charmed particles explicitely.
The combination of PRESHOWER with NUPRIM makes no sense as your primary may be either a high-energy gamma ray or a neutrino. Neither PRESHOWER nor NUPRIM may be combined with STACKIN.
The COASTUSERLIB, COREAS, INCLINED, and ROOTOUT options cannot be combined with the COMPACT option, nor with histogram producing options ANAHIST, AUGERHIST, INTTEST or MUONHIST.
The combination of INTTEST with PLOTSH or PLOTSH2 is not reasonable as in the INTTEST option the development of showers is suppressed. Also a combination of INTTEST with ANAHIST, AUGERHIST, COREAS, CURVED, EHISTORY, INCLINED, MUONHIST, PRESHOWER, STACKIN, or THIN gives no sense, as you are simulating just the first interaction without development of a complete shower. A combination with UPWARD makes no sense as in INTTEST anyway all upward going particles are respected. The combination of NUPRIM with INTTEST is not supported. The combination of FLUKA with INTTEST is not foreseen.
IACT is only possible with CERENKOV; IACT and ATMEXT have been tested only with UNIX computers.
VOLUMEDET and VOLUMECORR exclude each other, as you may use only one detector geometry at a time. Similarly INTCLONG and NOCLONG are excluding each other.
As in the CEFFIC option the atmospheric absorption is treated only in a manner suited for planar atmospheres, you should not combine the CURVED option with CEFFIC.
The combination of VOLUMECORR with VIEWCONE is not supported.
The AUGERHIST option cannot be combined with the IACT, CURVED, INTTEST, PLOTSH, or PLOTSH2 options.
Using the *./coconut* script file (see Sect. 2.2.2 page 16) conflicting options are indicated and will be avoided automatically.

4 Steering of the Simulation

The simulation of air showers is steered by commands (keywords) that have to be given on unit MONIIN (MONItor INput) in the card image format. A command consists of a keyword usually up to 6 characters long (left shifted, upper or lower case characters) and one or more arguments in the form:

```
KEYWRD    arg1   arg2   arg3  ...  arg$n$     comments
```

The keyword and the arguments must be separated at minimum by one blank. The last argument may be followed by comments up to column 512. Additional comments may be given on separate lines with the first 6 characters (the keyword) kept blank, with the first character a `c` or a `C` followed by a blank, or with the first character a $*$. In the IACT option (page 38) the lines starting with 'IACT ' are treated additionally as comment lines. Internally all characters including the keywords are converted to upper case characters, except the characters following the keywords EPOPAR, DATDIR, DIRECT, HISTDS, HOST, IACT, INFILE, NEXPAR, TELFIL, and USER. If you want to specify one of these character arguments by a blank, you should include the blank within apostrophes or quotation marks. The sequence of steering keywords is arbitrary. The valid keywords, the internal argument names, their nature (A = character, F = floating, I = integer, or L = logical), their default settings, their descriptions, and their limitations are listed in the following.

As CORSIKA is primarily designed to simulate EAS in the energy range 10^{11} eV to some 10^{20} eV the code contains parameterizations and approximations which are valid only for a limited range of some arguments. Leaving the recommended range might cause incorrect results or even end the execution of the program with an error message.

4.1 Run Number

```
RUNNR     NRRUN
```

Format = (A5, I), Default = 1
NRRUN : Run number of this simulation. This number is used to form part of the name of the various output files.
Limit is: $0 \leq$ NRRUN ≤ 999999

4.2 First Event Number

```
EVTNR     SHOWNO
```

Format = (A5, I), Default = 1
SHOWNO : Event number of first shower. The second shower will get number SHOWNO+1 and so on.
Limit is: $1 \leq$ SHOWNO ≤ 999999

4.3 Random Number Generator Initialization

`SEED  ISEED(i,k),i=1... 3`

Format = (A4, 3I), Defaults = k, 0, 0
ISEED($1, k$) : Contains the seed of the random number sequence k.
ISEED($2..3, k$) : Contain the number of calls N_{in} to the generator that are performed for initialization such that N_{in} =ISEED($2, k$) + 10^9·ISEED($3, k$).
At present at most $k = 7$ sequences are used: Sequence 1 for the hadron shower, 2 for the EGS4 part, 3 for the simulation of Cherenkov photons (only for CERENKOV option), 4 for the random offset of Cherenkov telescope systems with respect of their nominal positions (only for IACT option), 5 for the HERWIG routines in the NUPRIM option, 6 for the PARALLEL option, and 7 for the CONEX option. Their activation follows the sequence of occurrence of the keyword lines.
At minimum 2 seeds should be activated. If not sufficient seeds are activated, the default values are taken.
The use of ISEED($2, k$) > 0 and especially of ISEED($3, k$) > 0 should be avoided as presetting the random number generator by billions of calls needs considerable computing time. To get different random sequences it is sufficient to modify ISEED($1, k$).
When the `eventio` and other separate functions are enabled in the IACT option, an external random generator may be used.
Limit (to get independent sequences of random numbers) is: $1 \leq$ ISEED($1, k$) $\leq$ 900 000 000

4.4 Number of Showers

`NSHOW  NSHOW`

Format = (A5, I), Default = 10
NSHOW : Number of showers to be generated in a run.
Limit is: NSHOW ≥ 1

4.5 Primary Particle Definition

`PRMPAR  PRMPAR(0)`

Format = (A6, I), Default = 14
PRMPAR(0) : Particle type of the primary particle. See Table 4 (page 104) for the particle codes.
Limits are: $1 \leq$ PRMPAR(0) ≤ 5656 . Vector mesons, resonances, and charmed particles are excluded because of their short life time. Primary neutrinos can only be used in the NUPRIM option (page 49). Instable nuclei with mass number 5 or 8 may be used only with SIBYLL or QGSJET, for other models the cross-sections are not defined.

This keyword is not available in the STACKIN option.

4.6 Energy Range

```
ERANGE    LLIMIT   ULIMIT
```

Format = (A6, 2F), Defaults = 1.E4, 1.E4
LLIMIT : Lower limit and
ULIMIT : Upper limit of the primary particle energy range (in GeV). The primary energy is selected at random out of this interval. If LLIMIT = ULIMIT, the primary energy is fixed at this value.
The energies are total energies and include the particle rest mass.
Limits are: For primary nuclei LLIMIT $>$ HILOW (by default 80 GeV/nucleon for nuclei, see page 75); below this limit a simple superposition model is used.
For primary photons and electrons ULIMIT $\leq$ 1.E11 GeV (but keep in mind that no LPM-effect is included in NKG!). For primary hadrons and nuclei no upper limit is recommended, but **the user should take care not to over-stretch the selected hadronic interaction model**. See also Ref. [25]. It is recommended for HDPM: ULIMIT $\leq\ 1 \cdot 10^8$ GeV and for VENUS: ULIMIT $\leq\ 2 \cdot 10^7$ GeV.
This keyword is not available in the STACKIN option.

4.7 Slope of Spectrum

```
ESLOPE    PSLOPE
```

Format = (A6, F), Default = 0.
PSLOPE : Exponent γ of differential primary energy spectrum. The primary energy is taken at random from an exponential energy spectrum of the form $dN/dE_0 \ \propto \ E_0^{\gamma}$. PSLOPE has no meaning in case of fixed primary energy. The energies are total energies and include the particle rest mass.
This keyword is not available in the STACKIN option.

4.8 Zenith Angle Definition

```
THETAP    THETPR(1)   THETPR(2)
```

Format = (A6, 2F), Defaults = 0., 0.
THETPR(1) : Low edge of zenith angle range of primary particle (in $^\circ$).
THETPR(2) : High edge of zenith angle range of primary particle (in $^\circ$).
The zenith angle is selected at random out of this interval in a manner which respects equal particle fluxes from all solid angle elements of the sky and a registration by a horizontal flat

detector arrangement[61]. THETPR is the angle of incidence at a horizontal detector. THETPR(i) = 0. is vertical. If THETPR(1) = THETPR(2), the zenith angle is fixed at this value.
Limits[62] are: 0.$\leq$ THETPR(i) $\leq$ 70.

4.9 Azimuth Angle Definition

```
PHIP    PHIPR(1)   PHIPR(2)
```

Format = (A4, 2F), Defaults = 0., 0.
PHIPR(1) : Low edge of azimuth angle range of primary particle (in $^\circ$).
PHIPR(2) : High edge of azimuth angle range of primary particle (in $^\circ$).
The azimuth angle is selected at random out of this interval.
If PHIPR(1) = PHIPR(2), the azimuth angle is fixed at this value. For $\phi = 0^\circ$ the shower axis points to magnetic North, for $\phi = 90^\circ$ it points to West, see Fig. 1 (page 102).
Limits are: -360. $\leq$ PHIPR(i) $\leq$ 360 [63].

4.10 Viewing Cone Specifications

```
VIEWCONE    VUECON(1)   VUECON(2)
```

Format = (A8, 2F), Defaults = 0., 0.
VUECON(1) : Inner limiting angle of viewing cone (in $^\circ$).
VUECON(2) : Outer limiting angle of viewing cone (in $^\circ$).
The VIEWCONE option (see Sect. 3.5.32 page 57) selects the direction of primaries in a circular cone around the fixed primary direction THETPR(1) and PHIPR(1) (page 62) with the inner opening VUECON(1) and the outer opening VUECON(2). The zenith angular dependence of the selected detector geometry is maintained for flat horizontal rsp. spherical detectors (see Sect. 3.5.33 page 58).

[61] In the case you use a volume detector (sphere) or a vertical long string detector instead of a flat horizontal detector, you should respect this by selecting the preprocessor option VOLUMEDET (for sphere) or VOLUMECORR (for vertical long string) to get the angular dependence of the shower intensity as observed with such detectors (see Sect. 3.5.33 page 58).

[62] The zenith angle limitation is recommended for the standard CORSIKA version because of some approximations made in subroutine *NKG*. At $\theta > 70^\circ$ also the curvature of the Earth's surface must be taken into account. For large zenith angles you should use the CURVED option (page 45).
For the CURVED option the limit is 0.$\leq$ THETPR(i) $<$ 90.; for the CURVED option combined with the CERENKOV option the limit is 0.$\leq$ THETPR(i)$\leq$ 88.
For the UPWARD option with upward going primary the limits are 110. $\leq$ THETPR(i) $\leq$ 180.
For the CURVED option combined with the UPWARD option THETPR(i) has to be chosen in a manner excluding the range 90. $<$ THETPR(i) $<$ 90. + δ. The angle δ <90° is spanned between the upward going shower axis and the horizontal detector plane above the shower starting point. Further details are given in [57, 63]. The keyword IMPACT (page 65) for skimming horizontal showers overrides THETPR(i).

[63] In the output file the corresponding ϕ of each shower is given in the range [$-\pi$,π] radian.

Limits: 0. $\leq$ VUECON(1) $\leq$ VUECON(2) $<$ 90. The generation of showers with angles beyond the range of the program validity is skipped.
This keyword is only available in the VIEWCONE option.

4.11 Starting Altitude

`FIXCHI  THICK0`

Format = (A6, F), Default = 0.
THICK0 : The starting altitude (in g/cm^2 mass overburden) of the primary particle is set for all showers. This choice is not effective if the height of the first interaction is set by FIXHEI $>$ 0. (see Sect. 4.13 below). With this keyword the development of sub-showers starting at the chosen altitude within the atmosphere may be followed. The starting altitude must be above the lowest observation level.
In the UPWARD option with an upward primary particle the starting altitude must be below the observation level (page 57).
Limit is: 0. $\leq$ THICK0.

4.12 Starting Point of Arrival Timing

`TSTART  TMARGIN`

Format = (A6, L), Default = F
TMARGIN : Flag indicating the starting point of the arrival time scale. If TMARGIN = .false., the first interaction starts the clock. If TMARGIN = .true., the entrance into the atmosphere (rsp. THICK0, see above) is taken for starting the internal clock. Additionally, the deflection within the Earth's magnetic field is enabled for charged hadronic or muonic primaries on their path between entering the atmosphere and the first interaction, which otherwise is disabled in the standard version (for the CURVED and SLANT options see page 45 rsp. 53). For TMARGIN = .true. the height of the first interaction is written negative to element 7 of the event header block. This keyword is not available in the CONEX, CURVED, SLANT, or STACKIN options. TMARGIN is set .true. by default in the CONEX, CURVED, SLANT, and IACT options, but it may be overridden in the IACT option.

4.13 First Interaction Definition

`FIXHEI  FIXHEI  N1STTR`

Format = (A6, F, I), Defaults = 0., 0
FIXHEI : Fixes the height (in cm) of the first interaction of hadronic primaries (rsp. the starting altitude for em-particles) for all showers in a run. If FIXHEI = 0., the height of the first

interaction is varied at random according to the appropriate mean free path. In case of unstable hadronic primaries and fixed height the first interaction will not be a decay. The fixed height must be above the lowest observation level. If FIXHEI $> 0.$ is set, the starting altitude of the primary is not effective (see Sect. 4.11 above).
In the CURVED option the keyword FIXHEI cannot be used for em-primary particles.
In the UPWARD option with an upward primary particle the starting altitude must be below the observation level (page 57).
In the STACKIN option FIXHEI is needed to specify the altitude of the first, externally treated interaction.
N1STTR : Fixes the target of the first interaction: 1 = Nitrogen, 2 = Oxygen, 3 = Argon, $else$ = random selection according to the atmospheric abundances. This option is only applicable for high-energy hadronic primaries, i.e. primaries with an energy per nucleon of $E_{lab} \geq$ HILOW (see page 75). Also in the NUPRIM option it may be used.
With the STACKIN option one should select N1STTR = 0 as in this case it acts on that secondary particle which is treated first.
In case of unstable hadronic primaries and predetermined target the first interaction will not be a decay.
Limits are: 0. $\leq$ FIXHEI $<$ border of atmosphere (at 112.8E5 cm for atmospheric models $1 \leq$ MODATM ≤ 9 or MODATM ≥ 17) .

4.14 Skimming Incidence

```
IMPACT    HIMPACT(1)   HIMPACT(2)
```

Format = (A6, 2F), Default = 0., 0.
HIMPACT(1) : Lower value (in cm) for minimum altitude of horizontal shower axis.
HIMPACT(2) : Upper value (in cm) for minimum altitude of horizontal shower axis. The actual minimum altitude is selected at random out of this interval with uniform distribution. Zenith angles given by the keyword THETAP (page 62) are overridden by a calculation from the actual minimum altitude. See UPWARD option page 57.
Limits are: OBSLEV(1) $\leq$ HIMPACT(i) $\leq$ min(FIXHEI, FIXCHI, border of atmosphere [at 112.8E5 cm for atmospheric models $1 \leq$ MODATM ≤ 9 or MODATM ≥ 17]).
This keyword is only available in the combination of the CURVED option with the UPWARD option.

4.15 Stack Input File Name

```
INFILE    FILINP
```

Format = (A6, A132), Defaults = ‘ ’
FILINP : File name to define the name and directory of the input file containing the parameters of secondary particles (see Sect. 3.5.27 page 53). Lower case characters of FILINP are not

converted to capitals. Please keep in mind that in FORTRAN an automatic expansion of UNIX names like `'$HOME'` is not possible, rather you should give the fully expanded name of the directory and file.
Limit is: FILINP must not begin with a ~ (tilde) character.
This keyword is only available in the STACKIN option.

4.16 Stack Output File Name

`OUTFILE  FILOUT`

Format = (A7, A132), Defaults = ' '
FILOUT : File name to define the name and directory of the output file which will contain the parameters of the secondary particles produced in the first interaction. These particles are written only if the file name is defined. These particles may be read in in a later corsika-run established with the STACKIN option (see Sect. 3.5.27 page 53). Lower case characters of FILINP are not converted to capitals. Please keep in mind that in FORTRAN an automatic expansion of UNIX names like `'$HOME'` is not possible, rather you should give the fully expanded name of the directory and file.
Limit is: FILOUT must not begin with a ~ (tilde) character.
This keyword is not available in the CONEX or STACKIN options.

4.17 Atmospheric Model Selection

`ATMOD  MODATM`

Format = (A5, I), Default = 1
MODATM : Gives the number of the atmospheric parameterization.

MODATM = 0: Atmosphere as read in by keywords ATMA, ATMB, ATMC, and ATMLAY (uppermost layer unchanged).
MODATM = 1: U.S. standard atmosphere as parameterized by Linsley.
MODATM = 2: AT115 Central European atmosphere for Jan. 15, 1993.
MODATM = 3: AT223 Central European atmosphere for Feb. 23, 1993.
MODATM = 4: AT511 Central European atmosphere for May 11, 1993.
MODATM = 5: AT616 Central European atmosphere for June 16, 1993.
MODATM = 6: AT822 Central European atmosphere for Aug. 22, 1993.
MODATM = 7: AT1014 Central European atmosphere for Oct. 14, 1993.
MODATM = 8: AT1224 Central European atmosphere for Dec. 24, 1993.
MODATM = 9: Atmosphere as read in by keywords ATMA, ATMB, ATMC. Layers as in MODATM = 1 . . . 8.
MODATM = 10: Atmosphere as read in by keywords ATMA, ATMB, ATMC, and ATMLAY (uppermost layer also read in).

MODATM = 11: South pole atmosphere for March 31, 1997 (MSIS-90-E).
MODATM = 12: South pole atmosphere for July 01, 1997 (MSIS-90-E).
MODATM = 13: South pole atmosphere for Oct. 01, 1997 (MSIS-90-E).
MODATM = 14: South pole atmosphere for Dec. 31, 1997 (MSIS-90-E).
MODATM = 15: South pole atmosphere for January after Lipari.
MODATM = 16: South pole atmosphere for August after Lipari.
MODATM = 17: U.S. standard atmosphere as parameterized by Keilhauer.
MODATM = 18: Malargüe GDAS model for January after Will/Keilhauer.
MODATM = 19: Malargüe GDAS model for February after Will/Keilhauer.
MODATM = 20: Malargüe GDAS model for March after Will/Keilhauer.
MODATM = 21: Malargüe GDAS model for April after Will/Keilhauer.
MODATM = 22: Malargüe GDAS model for May after Will/Keilhauer.
MODATM = 23: Malargüe GDAS model for June after Will/Keilhauer.
MODATM = 24: Malargüe GDAS model for July after Will/Keilhauer.
MODATM = 25: Malargüe GDAS model for August after Will/Keilhauer.
MODATM = 26: Malargüe GDAS model for September after Will/Keilhauer.
MODATM = 27: Malargüe GDAS model for October after Will/Keilhauer.
MODATM = 28: Malargüe GDAS model for November after Will/Keilhauer.
MODATM = 29: Malargüe GDAS model for December after Will/Keilhauer.
The various atmospheric models are described in Appendix D (page 135 ff.).
Limits are: $0 \leq$MODATM≤ 29 .

4.18 Atmospheric Parameters A(i)

`ATMA  AATM1  AATM2  AATM3  AATM4  (AATM5)`

Format = (A4, 4F), Defaults = 0., 0., 0., 0. (for ATMOD 0)
Format = (A4, 5F), Defaults = 0., 0., 0., 0., 0. (for ATMOD 10)
AATMi : A-parameters for 4 layers of atmospheric model # 0 (or for 5 layers of atmospheric model # 10). For the 5^{th} layer a linear decrease is assumed, which in case of ATMOD = 0 uses the same parameters as the U.S. standard atmosphere. To be used with ATMOD = 0 or 10 .
Limit is: $0. <$ AATM5

4.19 Atmospheric Parameters B(i)

`ATMB  BATM1  BATM2  BATM3  BATM4`

Format = (A4, 4F), Defaults = 0., 0., 0., 0.
BATMi : B-parameters for 4 layers of atmospheric model # 0. For the 5^{th} layer a linear decrease is assumed with the same parameters as for the U.S. standard atmosphere. To be used with ATMOD = 0 or 10.

Limits are: BATM$i \neq 0$.

4.20 Atmospheric Parameters C(i)

`ATMC  CATM1  CATM2  CATM3  CATM4  (CATM5)`

Format = (A4, 4F), Defaults = 0., 0., 0., 0. (for ATMOD 0)
Format = (A4, 5F), Defaults = 0., 0., 0., 0., 0. (for ATMOD 10)
CATMi : C-parameters for 4 layers of atmospheric model # 0 (or for 5 layers of atmospheric model # 10). For the 5^{th} layer a linear decrease is assumed, which in case of ATMOD = 0 uses the same parameters as the U.S. standard atmosphere. To be used with ATMOD = 0 or 10.
Limits are: CATM$i > 0$.

4.21 Atmospheric Layer Boundaries

`ATMLAY  HLAY2  HLAY3  HLAY4  HLAY5`

Format = (A6, 4F), Defaults = 4.D5, 10.D5, 40.D5, 100.D5
HLAYi : Layer lower boundaries (in cm) for the layers of atmospheric model # 0 and # 10. A value of 0. is adopted for the HLAY1. If not specified, the default values of MODATM = 1 are used for MODATM = 0 and 10. For other models (MODATM $\neq 0$ and $\neq 10$), the default values correspond with the selected model MODATM. Should only be used with ATMOD = 0 or 10.
Limits are: $0. <$ HLAYi.

4.22 External Tabulated Atmosphere

`ATMOSPHERE  IATMOX  FREFRX`

Format = (A10, I, L), Defaults = 0, F
IATMOX : Use MODTRAN [40] atmospheric model `IATMOX = i` (in terms of density and refractive index) instead of CORSIKA built-in model. This requires a file named *atmprof*`i`.dat. MODTRAN model atmospheres supplied with the 'bernlohr' package include tropical (`i` = 1), mid-latitude summer (2), mid-latitude winter (3), sub-arctic summer (4), sub-arctic winter (5), and U.S. standard atmosphere 1976 (6). Additionally for the MAGIC Cherenkov telescope on La Palma the summer (7) and winter (8) atmospheres[64] are supplied. The atmosphere (9) gives the winter atmosphere at the South pole. User supplied models are possible (`i`≥ 9).
FREFRX : If .true., the atmospheric refraction for Cherenkov photons is taken into account (for plane-parallel atmosphere); if .false., refraction is ignored. The value of this second argument is ignored if the CERENKOV option is not selected.
This keyword is only available in the ATMEXT option and needs linking with the (compiled) *atmo*.c routines of the 'bernlohr' package.

4.23 Earth's Magnetic Field

```
MAGNET    BX    BZ
```

Format = (A6, 2F), Defaults = 20.40, 43.23
BX : Is the horizontal component of the Earth's magnetic field (in μT) to the x-direction of the detector (North) and
BZ : Is the vertical component of the Earth's magnetic field (in μT) downwards.
The default values represent the magnetic field for the Karlsruhe location. The values of other locations may be obtained from the program $Geomag$ which is available on-line in the world wide web [65]. The value H of $Geomag$ corresponds with our BX, the value Z with our BZ. For orientation see also Fig. 1 (page 102).
Limits are: BX, BZ $\neq$ 0. .

4.24 Experiment Coordinates for Pre-Showering and CONEX

```
GCOORD    GLONG    GLATI    GRFYEAR    IPREPR    IPRSTP
```

Format = (A6, 3F, 2I), Defaults = -69.585, -35.463, 2003., 1, 0
GLONG : Gives the geographical longitude (in °, West length is negative) of the experiment.
GLATI : Gives the geographical latitude (in °, South latitude is negative) of the experiment.
GRFYEAR : Gives the year of the experiment (the magnetic field is varying with time). These coordinates are used to calculate the magnetic dipole field of the Earth's globe above the atmosphere of the experiment's position in the case of a pre-shower induced by ultra-high energetic primary photons. The default values give the position coordinates of the southern Pierre Auger Observatory at Malargüe (Argentina) for the year 2003.
IPREPR : Print indicator[64]: IPREPR $\leq$ 0 disables pre-shower printing; IPREPR = 1 prints details of pre-shower in case of MAXPRT (page 86) or DEBUG (page 95); IPREPR $\geq$ 2 always prints details of pre-shower.
IPRSTP : If IPRSTP $\neq$ 0 events without pre-showering are skipped.
Limits are: -180. $\leq$ GLONG $\leq$ 180.; -90. $\leq$ GLATI $\leq$ 90.; 1965. $\leq$ GRFYEAR $\leq$ 2015.
This keyword is only available in the CONEX or PRESHOWER options.

4.25 DPMJET Selection Flag

```
DPMJET    FDPMJT    LEVLDB
```

Format = (A6, L, I), Defaults = T, 0
FDPMJT : If .true., the DPMJET routines are used to treat the high-energy hadronic interactions. If .false., the HDPM routines are used to treat the high-energy hadronic interactions.

[64]The print statements within the PRESHOWER C-routines write only to the standard output. A redirecting of the standard output to a logical unit differing from 6 (see keyword OUTPUT page 88) usually will not affect the output of the PRESHOWER C-routines.

LEVLDB : Gives amount of debug output for the DPMJET code in case of DEBUG. With increasing value up to 8 the debug output becomes more and more detailed. This output cannot be redirected and always appears on unit 6.
Limits are: $0 \leq$ LEVLDB ≤ 8 .
This keyword is only available in the DPMJET option.

4.26 DPJSIG Selection Flag

`DPJSIG` `FDPJSG`

Format = (A6, L), Default = T
FDPJSG : If .true, the DPMJET high-energy hadronic cross-sections are used. If .false., the default cross-sections as described in Ref. [3] are used.
This keyword is only available in the DPMJET option.

4.27 EPOS Selection Flag

`EPOS` `FNEXUS` `ISH0N`

Format = (A5, L, I), Defaults = T, 0
FNEXUS : If .true., the EPOS routines are used to treat the high-energy hadronic interactions. If .false., the HDPM routines are used to treat the high-energy hadronic interactions.
ISH0N : Determines amount of debug output for the EPOS routines. With increasing number ISH0N > 0 the output becomes more and more detailed. This output appears on the unit MDEBUG.
For more information look into the EPOS documentation. Additional debugging is effective by setting print parameters using EPOPAR *print* .... This debug output is written to the `ifch` file (see Table 1 page 24).
Limits are: $0 \leq$ ISH0N ≤ 9 .
This keyword is only available in the EPOS option.

4.28 EPOS Parameters

`EPOPAR` *parcha*

Format = (A6, A74), Defaults = ‘ ’
parcha : Command line to be read by subroutine *aread* of program block *epos*-`bas-lhc.f`.
The possible command lines are described in the EPOS documentation. Use lower case characters. Lower case characters of *parcha* are not converted to capitals. Do not use the commands

application ..., *set nevent ...*, *run*, or *stop* within your input parameters, these will cause unpredictable results or crashes. Only $epos$.inixx names might be changed by standard users (note that $.lhc$ is automatically added at the end of relevant file names)[65].
This keyword is only available in the EPOS option.

4.29 EPOSIG Selection Flag

```
EPOSIG    FNEXSG
```

Format = (A6, L), Default = T
FNEXSG : If .true., the EPOS high-energy hadronic cross-sections are used. If .false., the default cross-sections as described in Ref. [3] are used.
This keyword is only available in the EPOS option.

4.30 NEXUS Selection Flag

```
NEXUS    FNEXUS   ISH0N
```

Format = (A5, L, I), Defaults = T, 0
FNEXUS : If .true., the NEXUS routines are used to treat the high-energy hadronic interactions. If .false., the HDPM routines are used to treat the high-energy hadronic interactions.
ISH0N : Determines amount of debug output for the NEXUS routines. With increasing number ISH0N > 0 the output becomes more and more detailed. This output appears on the unit MDEBUG.
For more information look into the NEXUS documentation. Additional debugging is effective by setting print parameters using NEXPAR *print* .... This debug output is written to the `ifch` file (see Table 1 page 24).
Limits are: $0 \leq$ ISH0N ≤ 9 .
This keyword is only available in the NEXUS option.

[65] A typical EPOPAR input looks like:
EPOPAR input ~corsika-74xxx/epos/epos.param
EPOPAR fname inics ~corsika-74xxx/epos/epos.inics
EPOPAR fname iniev ~corsika-74xxx/epos/epos.iniev
EPOPAR fname inihy ~corsika-74xxx/epos/epos.ini1b
EPOPAR fname inirj ~corsika-74xxx/epos/epos.inirj
EPOPAR fname initl ~corsika-74xxx/epos/epos.initl
EPOPAR fname check ~corsika-74xxx/epos/epos.check
EPOPAR fname histo ~corsika-74xxx/epos/epos.histo ! for interaction test only
EPOPAR fname data ~corsika-74xxx/epos/epos.data ! for debugging only
EPOPAR fname copy ~corsika-74xxx/epos/epos.copy ! for debugging only
EPOPAR fname log ~corsika-74xxx/epos/epos.log ! for debugging only
EPOPAR printcheck screen ! for debugging only.

4.31 NEXUS Parameters

NEXPAR *parcha*

Format = (A6, A74), Defaults = ‘ ’
parcha : Command line to be read by subroutine *aread* of program block *nexus*-bas.f. The possible command lines are described in the NEXUS documentation. Use lower case characters. Lower case characters of *parcha* are not converted to capitals. Do not use the commands *application ...*, *set nevent ...*, *run*, or *stop* within your input parameters, these will cause unpredictable results or crashes. Only $nexus$.inixx names might be changed by standard users[66].
This keyword is only available in the NEXUS option.

4.32 NEXSIG Selection Flag

NEXSIG FNEXSG

Format = (A6, L), Default = T
FNEXSG : If .true., the NEXUS high-energy hadronic cross-sections are used. If .false., the default cross-sections as described in Ref. [3] are used.
This keyword is only available in the NEXUS option.

4.33 QGSJET Selection Flag

QGSJET FQGS LEVLDQ

Format = (A6, L, I), Defaults = T, 0
FQGS : If .true., the qgsjet-II-04 (rsp. QGSJET01d) routines are used to treat the high-energy hadronic interactions. If .false., the HDPM routines are used to treat the high-energy hadronic interactions.
LEVLDQ : Gives amount of debug output for the qgsjet-II-04 (rsp. QGSJET01d) code in case of DEBUG. With increasing value up to 4 the debug output becomes more and more detailed. This output cannot be redirected and always appears on unit 6.

[66] A typical NEXPAR input looks like:
NEXPAR fname inics ~corsika-74xxx/nexus/nexus.inics
NEXPAR fname iniev ~corsika-74xxx/nexus/nexus.iniev
NEXPAR fname inirj ~corsika-74xxx/nexus/nexus.inirj
NEXPAR fname initl ~corsika-74xxx/nexus/nexus.initl
NEXPAR fname check ~corsika-74xxx/nexus/nexus.check
NEXPAR fname histo ~corsika-74xxx/nexus/nexus.histo ! for interaction test only
NEXPAR fname data ~corsika-74xxx/nexus/nexus.data ! for debugging only
NEXPAR fname copy ~corsika-74xxx/nexus/nexus.copy ! for debugging only
NEXPAR fname log ~corsika-74xxx/nexus/nexus.log ! for debugging only
NEXPAR printcheck screen ! for debugging only.

Limits are: $0 \leq$ LEVLDQ ≤ 4 .
This keyword is only available in the QGSJET option.

4.34 QGSSIG Selection Flag

```
QGSSIG    FQGSSG
```

Format = (A6, L), Default = T
FQGSSG : If .true., the qgsjet-II-04 (rsp. QGSJET01d) high-energy hadronic cross-sections are used. If .false., the default cross-sections as described in Ref. [3] are used.
This keyword is only available in the QGSJET option.

4.35 SIBYLL Selection Flag

```
SIBYLL    FSIBYL   ISDEBUG
```

Format = (A6, L), Default = T, 0
FSIBYL : If .true., the SIBYLL routines are used to treat the high-energy hadronic interactions. If .false., the HDPM routines are used to treat the high-energy hadronic interactions.
ISDEBUG : Debug level; with increasing level the SIBYLL 2.1 debug output becomes more and more detailed. This output cannot be redirected and always appears on unit 6.
This keyword is only available in the SIBYLL option.

4.36 SIBSIG Selection Flag

```
SIBSIG    FSIBSG
```

Format = (A6, L), Default = T
FSIBSG : If .true., the SIBYLL high-energy hadronic cross-sections are used. If .false., the default cross-sections as described in Ref. [3] are used.
This keyword is only available in the SIBYLL option.

4.37 VENUS Selection Flag

```
VENUS     FVENUS   ISH00
```

Format = (A5, L, I), Defaults = T, 0
FVENUS : If .true., the VENUS routines are used to treat the high-energy hadronic interactions. If .false., the HDPM routines are used to treat the high-energy hadronic interactions.
ISH00 : Determines the amount of debug output for VENUS routines. With increasing number ISH00 ≥ 90 the output becomes more and more detailed. This output appears on the unit

MDEBUG. For more information look into the listing of subroutine *venini*.
Limits are: $0 \leq$ ISH00 ≤ 98 .
This keyword is only available in the VENUS option.

4.38 VENUS Parameters

```
VENPAR    PARCHA(i)   PARVAL(i)
```

Format = (A6, A6, F), Defaults = ‘ ’, 0.
PARCHA(i) : Name of VENUS parameter to be changed.
PARVAL(i) : New value of VENUS parameter to be changed.
A maximum of $i = 100$ VENUS parameters may be set by the user in arbitrary sequence. The available names and their meaning may be taken from the listing of subroutine $venini$.
The VENUS parameters should not be changed by standard users.
This keyword is only available in the VENUS option.

4.39 VENSIG Selection Flag

```
VENSIG    FVENSG
```

Format = (A6, L), Default = T
FVENSG : If .true., the VENUS high-energy hadronic cross-sections are used. If .false., the default cross-sections as described in Ref. [3] are used.
This keyword is only available in the VENUS option.

4.40 HDPM Interaction Parameters & Fragmentation

```
HADFLG    NFLAIN  NFLDIF  NFLPI0  NFLPIF  NFLCHE  NFRAGM
```

Format = (A6, 6I), Defaults = 0, 0, 0, 0, 0, 2
Steering flags of the high-energy hadronic interaction model HDPM and of the projectile nucleus fragmentation of all hadronic interaction models.

NFLAIN : The number of interactions of a projectile in a target nucleus may fluctuate (NFLAIN = 0) or is calculated as an average value (NFLAIN $\neq$ 0).
NFLDIF : No diffractive interactions are allowed in case of more than 1 interaction in the target (NFLDIF = 0) or diffractive interactions are possible (NFLDIF $\neq$ 0).
NFLPI0 : The rapidity distribution of π° is taken different from that of charged pions as indicated by collider data (NFLPI0 = 0) or is taken as for charged pions (NFLPI0 $\neq$ 0).
NFLPIF : The number of π° fluctuates in the same way as the number of charged pions (NFLPIF = 0) or fluctuates independently as parameterized from collider data (NFLPIF $\neq$ 0).

NFLCHE : Charge exchange reactions for the proj. and target particles are allowed (NFLCHE = 0) or inhibited (NFLCHE $\neq$ 0).

NFRAGM : A primary nucleus fragments at the first interaction completely into free nucleons (NFRAGM = 0) or successively by assuming that the non-interacting nucleons proceed as one new nucleus (NFRAGM = 1). This new nucleus may evaporate nucleons or alpha-particles with a transverse momentum distribution according to experimental data [66] (NFRAGM = 2, default) or with a transverse momentum distribution according to Goldhaber's theory [67] using 0.090 GeV/nucleon as the average transverse momentum (NFRAGM = 3). NFRAGM = 4 gives identical fragments as NFRAGM = 2 or 3, but without transverse momenta.

The NFRAGM flag is used also to steer the fragmentation in the various interaction models as described for the HDPM routines. EPOS, NEXUS and VENUS use the same evaporation model as HDPM with the same meaning of NFRAGM, while SIBYLL and QGSJET deliver themselves realistic nuclear fragments with according transverse momenta; they are selected by NFRAGM $\geq$ 2. In principle DPMJET offers a very detailed nuclear fragmentation model with evaporation. But there is no allowance to distribute it. Therefore the nuclear evaporation as used for HDPM, EPOS, NEXUS, and VENUS is coupled with DPMJET and the meaning of NFRAGM follows HDPM. Additionally NFRAGM = 5 is used to activate the DPMJET evaporation module if it exists.
Limits are: $0 \leq all\ flags < 100$

4.41 Neutrino Interaction Type Selection

`NUSLCT  NUSLCT`

Format = (A6, I), Default = 2
NUSLCT : Selects the type of the primary neutrino interaction: 0 = neutral current interaction, 1 = charged current interaction, $else$ = type of interaction is selected at random according to the interaction cross sections for the two processes.
This keyword is only available in the NUPRIM option.

4.42 Transition Energy between Models

`HILOW  HILOELB`

Format = (A5, F), Default = 80.
HILOELB : Allows to define the transition energy[67] (E_{lab} in GeV) between high and low-energy hadronic interaction model.

[67] With the INTTEST option the default value of this border is at 49 GeV rsp. 101 GeV, depending whether a high- or low-energy hadronic interaction model should be tested. For testing of DPMJET, EPOS, NEXUS, QGSJET, and VENUS the default value is set to 49 GeV, for SIBYLL to 60 GeV. If none of those models is selected, the default value is set to 101 GeV to test the models FLUKA, GHEISHA, or UrQMD.

Limits depend on the used interaction model, for most high-energy hadronic interaction models the low-energy limit is in the range of $\approx$ 80 GeV, for SIBYLL $\approx$ 60 GeV, while most low energy models enable a limit as high as several 100 GeV.

4.43 Electromagnetic Interaction Steering Flags

ELMFLG FNKG FEGS

Format = (A6, 2L), Defaults = T, F
FNKG : If .true., the NKG option is switched on for calculating the electromagnetic subcascades analytically. For the electron kinetic energy threshold the value of ELCUT(3) is taken (keyword ECUTS page 81). If .false., the NKG option is disabled[68].
FEGS : If .true., the EGS4 option is selected to calculate all interactions of e^+, e^-, and photons in the atmosphere explicitly. (The second random number sequence should be initialized for use in the EGS4 part. Otherwise the default initialization is taken.) If .false., the EGS4 option is disabled.
In the CERENKOV option this flag is obsolete as EGS4 is selected automatically.
The two options may be selected or disabled independently at the same time.

4.44 Electron Multiple Scattering Length Factor

STEPFC STEPFC

Format = (A6, F), Default = 1.
STEPFC : Factor by which the multiple scattering length for electrons and positrons in EGS4 simulations is elongated relative to the value given in [16]. A detailed discussion on the use of the step length is given in [34]. An enlargement of this factor may be tolerated to reduce computing time, but simultaneously the electron lateral distribution on ground becomes slightly narrower. With STEPFC = 10. the CPU-time is reduced by a factor of ≈ 1.7 (relative to the default value). A reduction of STEPFC will increase the computing time considerably, e.g with STEPFC = 0.1 by a factor of ≈ 5.
Limits are: 0. $<$ STEPFC $\leq$ 10.0

4.45 Radius of NKG Lateral Range

RADNKG RADNKG

[68] In the CURVED option the NKG formulas are no longer valid, therefore the NKG flag is disabled automatically in this option. The NKG flag should be disabled in the COMPACT option, as the resulting NKG parameters cannot be written out onto the particle output file.

Format = (A6, F), Default = 200.E2
RADNKG : Gives the outer range radius (in cm) within which the lateral NKG distribution is calculated for 10 radii equidistant in logarithmic scale. The inner radius is always kept at 100 cm.
Limit is: RADNKG $>$ 100.

4.46 Flags for Simplified CONEX Threshold Management

CASCADE FCXCAS FCXLCE FCXGHE

Format = (A7, 3L), Defaults = T, F, F
FCXCAS : If .true. optimized parameters are used to run in hybrid mode (MC+CE+MC) with full 3D information. Muons are fully tracked in MC. If .false. the cascade equations are disabled below the high-energy thresholds and only MC is used.
FCXLCE : If .true. default parameters are used to run in hybrid mode (MC+CE) with full 3D information for muons only. For hadrons and e/m particles, only 1D informations are relevant (longitudinal profile and energy distribution of particles at ground). Muons are fully tracked in 3D in MC. If .false. optimized sampling weights are used to run MC below the default low-energy thresholds. Weights defined by THIN keyword are used for thinning in CONEX and sampling weights are used as thinning weights in CORSIKA (see FCXCE in Sect. 4.49 page 78).
FCXGHE : If .true. default parameters are used to run in hybrid mode (MC+CE) for all particles. Only 1D informations are relevant (longitudinal profile and energy distribution of particles at ground). If .false. optimized sampling weights are used to run MC for muons below the default low-energy thresholds. Weights defined by THIN keyword are used for thinning in CONEX and sampling weights are used as thinning weights in CORSIKA (see FCXCE in Sect. 4.49 page 78).
Limits are: Only the combinations of the three flags T T T, T T F, T F F, or F F F are allowed.
Notes: This keyword should be used if the user wants to select easily different hybrid mode. In particulars the first MC part of the shower is not changed by changing the value of CASCADE (or even by introducing this command), thus a shower with the same longitudinal profile can be calculated with different speed and precision (1D or 3D) if the 7^{th} SEED sequence is not changed of course.
The sampling weights are adjusted to get a good compromise between precision and speed. If ECUTS are defined below CONEX limits (1 GeV for hadrons and muons and 1 MeV for e-m particles), low energy MC (T F F but with larger weight and lower threshold) is automatically switched on for the relevant part of the shower and to get reasonable precision in 1D, but it will be necessary slower than default T T F or T T T mode. To be able to reproduce a given shower, ECUTS should not be changed while changing CASCADE configuration.
To get the fastest simulation for any ECUTS see the keyword CORSIKA in Sect. 4.50 page 79.
It is recommended not to use this keyword together with the CX2COR or CXWMX keywords since it redefines some of the parameters.

This keyword must not be called after the energy cuts are defined (see keyword ECUTS Sect. 4.54 page 81).
This keyword is only available in the CONEX option.

4.47 Thresholds for CONEX Cascade Equations

`CONEX  CXTHR(1)  CXTHR(2)  CXTHR(3)`

Format = (A5, 3F), Defaults = 1.E-3, 1., 1.E-4
CXTHR(i) : Fractions of primary energy above which the shower particles are treated individually by Monte Carlo methods. Below these thresholds cascade equations treat the particles. These thresholds apply to hadrons (i = 1), muons (i = 2), and electrons/photons (i = 3).
Limits are: 0. $\leq$ CXTHR(i) $\leq$ 1.
This keyword is only available in the CONEX option.

4.48 Thresholds for Transition from CONEX to CORSIKA

`CX2COR  CXMCT(1)  CXMCT(2)  CXMCT(3)  CXMCS`

Format = (A6, 4F), Defaults = 3.E2, 1.E20, 1.E1, 4.E2
CXMCT(i) : Thresholds (in GeV) below which the particles are transferred from CONEX back to the CORSIKA stack. These thresholds apply to hadrons (i = 1), muons (i = 2), and electrons/photons (i = 3).
CXMCS : Minimal vertical depth (g/cm^2) above observation level below which the transfer of particles from CONEX to CORSIKA treatment is started.
Limits are: 0. $\leq$ CXMCT(i) $\leq$ 1.E20; 0. $\leq$ CXMCS $\leq$ 1.E20
In the CONEX option this keyword has to be called after the keyword CASCADE T F F (see Sect. 4.46 page 77) is used to be effective, but it is better not to use it with CASCADE keyword.
This keyword is only available in the CONEX option.

4.49 Weight Sampling for CONEX

`CXWMX  CXWMT(1)  CXWMT(2)  CXWMT(3)  FCXWMX  FCXCE`

Format = (A5, 3F, L), Defaults = -1., -1., -1., F, F
CXWMT(i) : Factors (GeV^{-1}) to be multiplied with the primary energy to get maximal sampling weights for particles transferred from CONEX to CORSIKA. These factors apply to hadrons (i = 1), muons (i = 2), and electrons/photons (i = 3). For negative or zero values of CXWMT(i) the CORSIKA thinning weight limits (see Sects. 4.51, 4.52, and 4.53) are taken as sampling weights too.
FCXWMX : If .true. forces the usage of CONEX sampling weight limits also in thinning (in

CONEX and CORSIKA) and these weight limits are determined with the CXWMT(i) factors. If .false. and FCXCE = .false. the CORSIKA weight limits are used as thinning weights for both CONEX and CORSIKA (see Sect. 4.46) and forces the usage of the CONEX weight limits also in CORSIKA.
FCXCE : If .true. forces the usage of CONEX sampling weight limits also in thinning for CORSIKA only and these weight limits are determined with the CXWMT(i) factors. The CORSIKA weight limits are used as thinning weight for CONEX only. If FCXCAS = .true. (see Sect. 4.46) forces FCXCE to be .true.
Limits are: -1. $\leq$ CXWMT(i) $\leq$ 1.
In the CONEX option this keyword has to be called after the keyword CASCADE (see page 77) is used to be effective, but it is better not to use it with CASCADE keyword.
This keyword is only available in the CONEX option.

4.50 CONEX without CORSIKA

```
CORSIKA   FCORS
```

Format = (A7, 1L), Defaults = T
FCORS : If .true. allows CORSIKA to run low energy MC. If .false. all particles which are supposed to go into the CORSIKA stack are lost and only CONEX is run with CE going to the minimum energy. This is equivalent to CASCADE T T T (see Sect. 4.46) but with no influence of ECUTS.
This keyword should be used with care because only the longitudinal energy deposit profile is valid. Even the X_{max} fit based on the number of electrons could be biased because ECUTS below CONEX limits are not taken into account. This keyword is useful to get very fast 1D energy deposit profile. The change of FCORS doesn't change the first interactions of the shower so with the same SEED the same X_{max} for the energy deposit will be obtained.
This keyword is only available in the CONEX option.

4.51 Thinning Definition

```
THIN    EFRCTHN   WMAX   RMAX
```

Format = (A4, 3F), Defaults = 1.E-4, 1.E30, 0.E0
EFRCTHN : Factor ε_{th} which defines the energy fraction of the primary energy below which the thinning algorithm becomes active. If the fraction is selected in a manner that this energy is below the lowest energy threshold of ELCUT(i), $i = 1...4$ (keyword ECUTS page 81), thinning will not become active, but the particle output data structure will contain the weight (= 1.) for each particle.
WMAX : Weight limit for thinning. If the weight of a particle exceeds WMAX, no further

thinning is performed[69].
RMAX : Maximum radius (in cm) at observation level within which all particles are subject to inner radius thinning. Particles are selected with probability $(r/rmax)^4$. The weight of surviving particles is multiplied by the appropriate factor (inverse of probability). This thinning neither affects the shower development nor the table output nor the histogram output of the ANAHIST, AUGERHIST, or MUONHIST options, rather only the particle output file written onto MPATAP (and the Cherenkov output file written onto MCETAB). For RMAX $\leq 0.$ no radial thinning is applied.
Limits are: ULIMIT·EFRCTHN $\leq 1 \cdot 10^7$ GeV (for ULIMIT see keyword ERANGE Sect. 4.6 page 62); $0.1 \leq$ WMAX $\leq 1 \cdot 10^{20}$.
This keyword is only available in the THIN option.

4.52 Hadronic Thinning Definition

```
THINH    THINRAT   WEITRAT
```

Format = (A5, 2F), Defaults = 1., 1.
THINRAT : Defines hadronic thinning limit differing from em-thinning limit EFRCTHN by the ratio of $\varepsilon_{th_{em}}/\varepsilon_{th_{hadr}}$ which gives the ratio between the energy of the em-particles (specified by keyword THIN, see above) and the energy of the hadronic particles below which the thinning algorithm becomes active for these particle species (see also Sect. 4.51 above).
WEITRAT : Defines hadronic weight limit differing from em-weight limit WMAX by the ratio[70] of weight limit of em-particles to weight limit of hadronic particles in case of thinning (see also Sect. 4.51 above).
A simultaneous use of the keyword THINH together with THINEM is not tolerated and will lead to an error stop.
Limits are: ULIMIT·EFRCTHN/THINRAT $\leq 1{\cdot}10^7$ GeV (for ULIMIT see keyword ERANGE Sect. 4.6 page 62); $1 \cdot 10^{-4} \leq$ WEITRAT $\leq 1 \cdot 10^6$.
This keyword is only available in the THIN option.

4.53 Electromagnetic Thinning Definition

```
THINEM    THINRAT   WEITRAT
```

Format = (A6, 2F), Defaults = 1., 1.
THINRAT : Defines em-thinning limit differing from hadronic thinning limit EFRCTHN by the ratio of $\varepsilon_{th_{em}}/\varepsilon_{th_{hadr}}$ which gives the ratio between the energy of the em-particles and the energy of the hadronic particles (specified by keyword THIN, see above) below which the thinning algorithm becomes active for these particle species (see also Sect. 4.51 above).

[69] See footnote page 55.

[70] In the Slowenian thinning [59] an ϵ is defined which gives the inverse of WEITRAT. There the weight limit w_{max} is defined for em-particles and from this the weight limit for hadrons and muons is derived by $\epsilon \cdot w_{max}$.

WEITRAT : Defines em-weight limit differing from hadronic weight limit WMAX by the ratio[70] of weight limit of em-particles to weight limit of hadronic particles in case of thinning (see also Sect. 4.51 above).
A simultaneous use of the keyword THINEM together with THINH is not tolerated and will lead to an error stop.
Limits are: ULIMIT·EFRCTHN·THINRAT $\leq 1 \cdot 10^7$ GeV (for ULIMIT see keyword ERANGE Sect. 4.6 page 62); $1 \cdot 10^{-4} \leq$ WEITRAT $\leq 1 \cdot 10^6$.
This keyword is only available in the THIN option.

4.54 Energy Cut-Offs

```
ECUTS    ELCUT(i), i=1... 4
```

Format = (A5, 4F), Defaults[71] = 0.3, 0.3, 0.003, 0.003
ELCUT(i) : The low energy cut-off (in GeV) of the particle kinetic energy may be chosen differently for hadrons (without π^0's) ($i = 1$), muons ($i = 2$), electrons ($i = 3$), and photons (including π^0's) ($i = 4$). For nuclei ELCUT(1) is applied to the energy per nucleon.
It is in the responsibility of the user to choose the cut-off values in a reasonable way not to eliminate those parent particles which might decay to secondaries which you are looking for in the investigated problem (e.g. decay of muons to electrons).
Limits are: ELCUT(1)[72] ≥ 0.05 ; ELCUT(2)[73] ≥ 0.01; ELCUT(3), ELCUT(4) ≥ 0.00005
The value of ELCUT(3) is also taken as threshold value for the NKG calculation. In this case an upper limit of ELCUT(3) < 0.08 is recommended.
In the CONEX option this keyword has to be called before the keyword CASCADE (see page 77) is used.

4.55 Time Cut-Off

```
TIMLIM    DSTLIM   LTMLMPR
```

Format = (A6, F, L), Defaults = 1.D8, F
DSTLIM : Gives the distance (in cm) a particle would travel with velocity of light downstream the detector before cut away by the time limit. An additional security time of 20 μsec (corresponding with ≈ 6 km) is taken into account.
LTMLMPR : Flag which enables (T) or disables (F) printing of particles which exceed the time limit.
Limit is: DSTLIM > 0.

[71] For the INTTEST option: Defaults = 0., 0., 0., 0. and all limits are ELCUT(i) ≥ 0.
[72] ELCUT(1) is used also for neutrinos in the NEUTRINO option. In the FLUKA version ELCUT(1) should be ≥ 0.02 GeV; in the URQMD option ELCUT(1) should be ≥ 0.3 GeV.
[73] ELCUT(2) is used also for τ-leptons in the CHARM or TAULEP option.

This keyword is only available in the CURVED option.

4.56 Longitudinal Shower Development

`LONGI  LLONGI  THSTEP  FLGFIT  FLONGOUT`

Format = (A5, L, F, 2L), Defaults = F, 20.0, F, F
LLONGI : If .true., the longitudinal development of particle numbers for gammas (EGS4), positrons (EGS4), electrons (EGS4), positive and negative muons, hadrons, all charged, nuclei, and Cherenkov photons (CERENKOV) is sampled. Moreover the longitudinal development of the energy content in the various particle species (same order as before, but without Cherenkov photons) is sampled. Additionally the longitudinal development of energy deposit by ionization energy loss and by angular or energy cuts is sampled. See also Sect. 10.1 page 108. To get the sampling in slant depth instead of the (default) vertical depth you should use the SLANT option (page 53). If .false., the longitudinal development is not sampled.
THSTEP : Vertical step width (rsp. slant step width in the SLANT option) for sampling of the longitudinal development (in g/cm^2). The sampling is done in vertical (rsp. slant) depth. The altitudes are not depending on the zenith angle of the primary particle (except the preprocessor option SLANT has been selected). In the CURVED option the minimum step size has to be selected in a manner that no more than 1875 steps are needed to pass through the complete atmosphere.
FLGFIT : If .true. and LLONGI also .true., the longitudinal development of all charged particles number is fitted. If .false., the fit is suppressed.
FLONGOUT : If .true. and LLONGI also .true., the longitudinal distributions of particle numbers and energy deposit for the various particle groups are written to the '$DATnnnnnn$.long' file (see Sect. 10.5 page 119).
If .false. and LLONGI .true., the longitudinal distributions only of the particle numbers for the various particle species are written out to the particle output file '$DATnnnnnn$' in extra 'LONG' sub-blocks (see Sect. 10.2, Table 6 page 110 and Table 12 page 116).
Limits are: 1. $\leq$ THSTEP $\leq$ 1875.
20. $\leq$ THSTEP $\leq$ 1875 for the SLANT option and horizontal incidence.

Normally only to the number distribution of all charged particles a function of the Gaisser-Hillas type [68]

$$N(t) = N_{max} \cdot \left(\frac{t - t_0}{t_{max} - t_0}\right)^{\frac{t_{max}-t_0}{a+bt+ct^2}} \cdot exp\left(\frac{t_{max} - t}{a + bt + ct^2}\right)$$

is fitted to describe the dependence on the atmospheric depth t and the resulting 6 parameters $N_{max}, t_0, t_{max}, a, b,$ and c and the χ^2/dof are stored in the event end block. The longitudinal development of the electromagnetic particles is only sampled if EGS4 is selected (see ELMFLG). If only NKG is activated the fit is applied to the NKG longitudinal distribution which consists of particle numbers from only ≤ 10 levels. If neither EGS4 nor NKG is selected the charged particle distribution contains only muons and charged hadrons. In the AUGERHIST

option also a Gaisser-Hillas type function is fitted to the longitudinal energy deposit, if EGS4 is selected.
In the Cherenkov versions the longitudinal distribution of photons is given in differential mode (i.e. the number of photons generated within each step) as default. By the preprocessor option INTCLONG the integral mode is selected (i.e. accumulated number of generated Cherenkov photons for each step) which needs additional computing time. If both kinds of the distribution are of no interest, you may deselect the Cherenkov photon distribution completely by the preprocessor option NOCLONG thus saving computing time.

4.57 Muon Multiple Scattering Treatment

`MUMULT  FMOLI`

Format = (A6, L), Default = T
FMOLI : If .false., the muon multiple scattering angle is selected by Gauss approximation. If .true., the muon multiple scattering angle is selected for large steps by Molière's theory and for small steps by adding many single Coulomb scattering events.

4.58 Additional Electromagnetic Particle Information

`EMADDI  FEMADD`

Format = (A6, L), Default = F
FEMADD : If .false., no additional information on electromagnetic (EM) particles is written to particle output file. If .true., additional information on mother and grandmother hadrons of EM particles at the origin of the EM subshower is written to the particle output file. Details are similar to those given in Ref. [51] for muons in the EHISTORY option (Sect. 3.5.13 page 46). In case of em-particle primaries the additional particle information is suppressed (by automatically setting FEMADD = .false.) as for the bulk of the em-particles no hadronic mother or grandmother particles exists.
This keyword is only available in the EHISTORY option and not in the INTTEST option.

4.59 Additional Muon Information

`MUADDI  FMUADD`

Format = (A6, L), Default = F
FMUADD : If .false., no additional muon information is written to particle output file. If .true., additional information on muons at their origin is written to the particle output file. This additional muon information consists of 7 data words according to Table 10 (page 115) and precedes

the corresponding muon particle on the particle output file. The first data word contains the particle identification 75 (for μ^+) or 76 (for μ^-) combined with the hadronic generation counter, which in this case may differentiate between muons originating from K-decay (normal generation counter) and $\pi^\pm$-decay (generation counter incremented by 50). The 7^{th} data word contains the altitude (in cm) of the muon birth instead of time (see also Table 10 page 115).
In combination with the EHISTORY option (see Sect. 3.5.13 page 46) an extended additional muon information is written to the particle output file. In the combination of the MUPROD option (see Sect. 3.5.18 page 48) with the EHISTORY option the extended muon information is written also for muons which decay or interact before reaching the lowest observation level. Details are given in Ref. [51].
This keyword is not available in the INTTEST option.

4.60 Additional Neutrino Information

`NUADDI  FNUADD`

Format = (A6, L), Default = F
FMUADD : If .false., no additional neutrino information is written to particle output file. If .true., additional information on mother and grandmother hadrons of neutrinos at their origin is written to the particle output file. Details are similar to those given in Ref. [51] for muons in the EHISTORY option (Sect. 3.5.13 page 46). An extended additional neutrino information is written to the particle output file.
This keyword is only available in the EHISTORY option and not in the INTTEST option.

4.61 Observation Level Definition

`OBSLEV  OBSLEV(i)`

Format = (A6, F), Default = 110.E2
OBSLEV(i) : Observation level i above sea level (in cm). This keyword has to appear once for each level to be defined. At maximum up to 10 observation levels are possible[74]. Their sequence is arbitrary.
In the UPWARD option (page 57) for upward going primaries the observation level should be chosen preferentially at the top of atmosphere, but at minimum above the starting point of the shower. The value of OBSLEV has to be selected in a manner that the shower axis crosses the observation level.
Limits are[75]: $0 \leq$ OBSLEV(i) $<$ *top of atmosphere*

[74] Only one observation level is possible in the CURVED option. Up to 20 levels might be specified for the production of histograms in the AUGERHIST option, the lowest observation level must be at minimum 1 g/cm^2 above sea level.

[75] For atmospheric models $1 \leq$ MODATM ≤ 9 and MODATM ≥ 17 limits are: $-$1.E5 $<$ OBSLEV(i) $<$ 112.8E5

In the CURVED option only one observation lavel can be defined.

4.62 Inclined Observation Plane

`INCLIN  XPINCL  YPINCL  ZPINCL  THINCL  PHINCL`

Format = (A6, 5F), Defaults = 0., 0., $OBSLEV(1)$, 0., 0.
XPINCL : X-coordinate (in cm) of reference point in inclined observation plane.
YPINCL : Y-coordinate (in cm) of reference point in inclined observation plane.
ZPINCL : Z-coordinate (in cm) of reference point in inclined observation plane.
THINCL : θ-angle (in deg) of normal vector of inclined observation plane. θ = 0. points to the zenith (opposite to Fig. 1 page 102) and defines a horizontal plane.
PHINCL : ϕ-angle (in deg) of normal vector of inclined observation plane.
All coordinates are defined in the standard CORSIKA output coordinate system (see Fig. 1 page 102) with its origin at the point where the shower axis hits the sea level.
ATTENTION: CORSIKA discards all particles below the height of the lowest standard observation level (see Sect. 4.61 page 84 for definition of OBSLEV), so as soon as the inclined observation plane drops below the lowest standard (horizontal) observation level, there will be no particles recorded any more.
Particles on the inclined plane will be stored in a coordinate system within the inclined plane, with the origin at the core (point where the shower axis intersects the observation level).
This keyword uses the COAST package and is available *only* when the INCLINED option (see page 25) in *./coconut* has been selected.
Technically, the INCLINED option uses a particular COASTUSERLIB library (Sec. 3.5.7 page 43), namely `InclinedPlane`, which is shipped with COAST in the `CorsikaOptions` directory by default.
This keyword is only available in the INCLINED option.

4.63 Observation Level Curvature

`CURVOUT  FCURVOUT`

Format = (A7, L), Default = T
FCURVOUT : If .true. the observation level is a sphere following the Earth curvature at an altitude $H = OBSLEV(1)$ and (X, Y) in the output file are replaced by (X', Y') which can be used to calculate the angles necessary to obtain (X, Y, Z) in a Cartesian frame: defining θ and ϕ the spherical coordinates of the particle as

$$\begin{aligned} \theta &= \frac{\sqrt{X'^2 + Y'^2}}{R_{Earth} + H} \\ \phi &= atan2(Y', X') \end{aligned}$$

with $R_{Earth} = 637131500$ cm. At the observation level ($H = OBSLEV(1)$) the Cartesian coordinates can be obtained using $D = (R_{Earth} + H) \cdot \sin\theta$ as

$$\begin{aligned} X &= D \cdot \cos\phi \\ Y &= D \cdot \sin\phi \\ Z &= (R_{Earth} + H) \cdot \cos\theta - R_{Earth} \end{aligned}$$

If θ is small (close to shower core) then (X', Y') = (X, Y) . If .false. the observation level is considered as flat and particle positions (X, Y) are defined in a Cartesian frame with the origin (0, 0) at the core position. Particles are tracked only until $Z = OBSLEV(1)$ in this Cartesian frame (apparent height). As a consequence, far from the core, some particles can be discarded before reaching the ground making a bias in the longitudinal profile. To avoid such a problem, FCURVOUT = .false. is forbidden by default if the zenith angle is between 85° and 95°.
The keyword FLATOUT = .not.FCURVOUT can still be used for backward compatibility.
These keywords are only available in the CURVED option and not in the ANAHIST or AUGERHIST options.

4.64 Array Rotation

ARRANG ARRANG

Format = (A6, F), Default = 0.
ARRANG : Defines a rotation angle (in °) between the detector array x-direction and magnetic north direction; positive if detector array x-direction points to the West.
Limits are: -180. $\leq$ ARRANG $\leq$ 180.
This keyword is not available in the EHISTORY option.

4.65 String Detector Configuration

DETCFG DETCFG

Format = (A6, F), Default = 0.
DETCFG : Gives the geometry configuration of a long vertical string detector as the ratio *height/diameter*.
Limit is: DETCFG $>$ 0.
This keyword is only available in the VOLUMECORR option.

4.66 Event Printout

MAXPRT MAXPRT

Format = (A6, I), Default = 10
MAXPRT : Is the maximum number of events that produce a detailed printout during the simulation run.
Limit is: MAXPRT $>$ 0

4.67 Particle Printout

`ECTMAP  ECTMAP`

Format = (A6, F), Default = 1.E4
ECTMAP : Defines a cut in the particle γ factor (or energy in GeV for em-particles and neutrinos) above which they are printed out on the logical unit MONIOU when passing an observation level.

4.68 Output Directory

`DIRECT  DSN`

Format = (A6, A239), Defaults = 'anynameupto239characters/'
DSN : May be used to define a name of an output directory. Lower case characters of DSN are not converted to capitals. Do not use capitals with the ANAHIST, AUGERHIST, MUONHIST, or INTTEST options as the HBOOK routines use only lower case characters. To suppress the output you might give `/dev/null` [76] or use the keyword PAROUT.
If you want to write into the directory from where you are starting your CORSIKA run, you should give: `DIRECT ' '` or `DIRECT " "` (a blank enclosed in apostrophes or quotation marks). Please keep in mind that in FORTRAN an automatic expansion of UNIX names like `'$HOME'` is not possible, rather you should give the fully expanded name of the directory ending with a '/' character.
Limit is: DSN must not begin with a ~ (tilde) character.

4.69 Table Output

`PAROUT  FPAROUT  FTABOUT`

Format = (A6, 2L) Defaults = T, F
FPAROUT : If .false., the particle output onto MPATAP is suppressed. This might be of advantage with the CERENKOV option to suppress the particle output file but keeping the Cherenkov output file (see Sect. 4.78).
FTABOUT : If .true., the tabular output of the charged particle development is written out to the file '$DATnnnnnn$.tab' onto the output directory DSN (see Sect. 4.68 above).

[76] A dummy directory named `/dev/null` must be mounted in the system.

4.70 Compact Output

`COMOUT  COMOUT`

Format = (A6, F), Default = T
COMOUT : If .true., the particle output is written in COMPACT form (see page 43). If .false., the standard CORSIKA particle output is written.
This keyword is only available in the COMPACT option.

4.71 Printer Output Unit

`OUTPUT  MONNEW`

Format = (A6, I), Default = 6
MONNEW : Logical unit of simulation control output on line printer. Make sure that your selection of MONNEW is not conflicting with existing definitions (see Table 1 page 24).
In the PARALLEL + PARALLELIB option the default value of MONNEW is changed to 89 for writing individual output files with names as the *cutfile* seeds, see Ref. [54].

4.72 Data Table Directory

`DATDIR  DATDIR`

Format = (A6, A132), Default = './'
DATDIR : Can be used to specify a common directory, where CORSIKA will try to find all required input data tables (except those belonging to FLUKA). Lower case characters of DATDIR are not converted to capitals.
Limit is: DATDIR must not begin with a ~ (tilde) character.

4.73 Parameters for Parallel Treatment

`PARALLEL  ECTCUT  ECTMAX  MPIID  FECTOUT`

Format = (A8, 2F, I, L), Defaults = 1.E4, 1.E7, 1, F
ECTCUT : Threshold energy (in GeV) for subshowers. All particles with energies above ECTCUT will have a seed from the 6^{th} sequence of random numbers (see keyword SEED page 61).
ECTMAX : Maximum energy (in GeV) for a complete subshower.
MPIID : Identification number of the parallel task which will be changed internally by MPI.
FECTOUT : If .true., particles with energy above ECTCUT are written additionally in an external *DATnnnnnn-iiiiiiii-kkkkkkkkk*.cut file. This is needed when parallel runs are performed via shell scripting. These cut-files are written in binary mode. For the case of parallel runs via

MPI treatment this is optional and could be used for debugging or for additional re-simulations of secondary subshowers to escape re-running the whole simulation from start/primary particle. Those cut-files are written in ASCII.
This keyword has to be specified before the keyword CUTFILE (see Sect. 4.74 below) is used. This keyword is only available in the PARALLEL option [77].

4.74 Reading of Cutted Particle File for Parallel Treatment

```
CUTFILE    CFILINP    I1CUTPAR    I2CUTPAR
```

Format = (A7, A255, 2I), Defaults = ‘ ’, 0, 0
CFILINP : Input file name for cut particles to be read into second stack.
I1CUTPAR : Index of first particle to be used for the actual run.
I2CUTPAR : Index of last particle to be used for the actual run.
This keyword has to be used when compilation is made with the PARALLEL option (see Sect. 3.5.21) to run parallel CORSIKA simulations distributed with the help of shell scripts. It must be used after the keyword PARALLEL (see Sect. 4.73 above) has been specified.
This keyword is only available in the PARALLEL option, but not interpreted by CORSIKA in the PARALLELIB option with MPI.

4.75 Cherenkov Detector Array Definition

```
CERARY    NCERX    NCERY    DCERX    DCERY    ACERX    ACERY
```

Format = (A6, 2I, 4F), Defaults = 27, 27, 1500., 1500., 100., 100.
NCERX : Number of Cherenkov detectors in X-direction.
NCERY : Number of Cherenkov detectors in Y-direction.
DCERX : Grid spacing (in cm) of Cherenkov detectors in X-direction. The DCERX value has no relevance in case of NCERX = 1.
DCERY : Grid spacing (in cm) of Cherenkov detectors in Y-direction. The DCERY value has no relevance in case of NCERY = 1.
ACERX : Length (in cm) of each Cherenkov detector in X-direction.
ACERY : Length (in cm) of each Cherenkov detector in Y-direction.
The altitude of this array is at the lowest observation level. For the definition of the X and Y-directions see Fig. 1 (page 102) and keyword ARRANG (page 86).
Limits are: NCERX, NCERY ≥ 1 ; DCERX, DCERY, ACERX, ACERY ≥ 1.
This keyword is only available in the CERENKOV option, but not in the IACT option for Cherenkov telescopes.

[77] The MPI-Runner [54] reads this keyword and transmits the arguments to the subroutine *corsika*.

4.76 Cherenkov Wavelength Band

`CWAVLG` `WAVLGL` `WAVLGU`

Format = (A6, 2F), Defaults = 300., 450.
WAVLGL : Lower limit (in nm) of the wavelength band for Cherenkov radiation production.
WAVLGU : Upper limit (in nm) of the wavelength band for Cherenkov radiation production.
Limits are: 100. $\leq$ WAVLGL $<$ WAVLGU $\leq$ 700.
This keyword is only available in the CERENKOV, AUGCERLONG, or AUGERHIST options.

4.77 Cherenkov Bunch Size Definition

`CERSIZ` `CERSIZ`

Format = (A6, F), Default = 0.
CERSIZ : Defines the maximal bunch size of Cherenkov photons that are treated together. If set to 0., by the subroutine $getbus$ the program calculates a bunch size which is found to be appropriate for the HEGRA-array.
Limit is: CERSIZ $\geq$ 0.
This keyword is only available in the CERENKOV, AUGCERLONG, or AUGERHIST options.

4.78 Cherenkov Output Steering

`CERFIL` `LCERFI`

Format = (A6, L), Default = T
LCERFI : If .true., Cherenkov output is written to the Cherenkov output file MCETAP. If .false., Cherenkov output is written to the particle output file MPATAP.
In the IACT option (Cherenkov telescopes) with LCERFI .true., the output file name DSN (specified by keyword DIRECT) should be set to `/dev/null` [77] to suppress the normal Cherenkov output file, as the Cherenkov telescope output will be written to the `eventio` output file[78].
LCERFI automatically will be set .true. in the COMPACT option to prevent a writing of Cherenkov photons to the COMPACT output.
This keyword is only available in the CERENKOV option.

4.79 Cherenkov Quantum Efficiency

`CERQEF` `CERQEF` `CERATA` `CERMIR`

[77] The existence of `/dev/null` is assumed, see footnote of Sect. 4.68.
[78] Details on the `eventio` format may be found in the documentation supplied with the 'bernlohr' package.

Format = (A6, 3L), Defaults = F, F, F
CERQEF : If .true., quantum efficiency of detector photomultiplier is taken into account. It needs reading in the *quanteff*.dat file.
CERATA : If .true., the atmospheric absorption of Cherenkov photons is taken into account. It needs reading in the *atmabs*.dat file.
CERMIR : If .true., the mirror reflectivity of Cherenkov telescopes is taken into account. It needs reading in the *mirreff*.dat file.
Respecting these effects at an early stage of the Cherenkov photon simulation drastically reduces computing time and storage requirements for Cherenkov photon output. For the influence onto the longitudinal distribution of Cherenkov photons see Sect. 3.4.5 page 39 and keyword LONGI page 83.
This keyword is only available in the CEFFIC option together with the CERENKOV option.

4.80 Multiple Use of Cherenkov Events

`CSCAT  ICERML  XSCATT  YSCATT`

Format = (A5, I, 2F), Defaults = 1, 0., 0.
ICERML : Number of uses of each event.
XSCATT : Maximum scattering of core location in $\pm X$ direction (in cm). See Sect. 3.4.1 page 36 ff.
YSCATT : Maximum scattering of core location in $\pm Y$ direction (in cm). See Sect. 3.4.1 page 36 ff.
Limits are: $0 \leq$ ICERML ≤ 20 ; XSCATT, YSCATT ≥ 0.
In case of IACT option (Cherenkov telescopes) ICERML telescope arrays are simulated randomly (see keyword SEED page 61) in the specified area which is a circle of radius XSCATT, if YSCATT $= 0.$, or within a rectangle of area 2 XSCATT $\cdot$ 2 YSCATT.
This keyword is only available in the CERENKOV option.

4.81 Cherenkov Telescope Dimensions

`TELESCOPE  X  Y  Z  R`

Format = (A9, 4F)
X, Y, Z : Coordinates of Cherenkov telescope (in cm) relative to the center of the observation level. This keyword adds a new telescope at position X, Y, Z with radius R, within which the telescope is fully contained. At least one telescope has to be specified. For the definition of the X and Y-directions see Fig. 1 (page 102) and keyword ARRANG (page 86).
Limits are: $0 < R$; $1 \leq$ number of telescopes < 1000.
This keyword is only available in the CERENKOV option together with the IACT option for Cherenkov telescopes.

4.82 Cherenkov Telescope Data File Name

`TELFIL  TELFNM`

Format = (A6, A100)
TELFNM : The telescope-specific data are to be written to a file named TELFNM in `eventio` format[79]. Lower case characters of TELFNM are not converted to capitals. If this file exists and is write-enabled, new data are appended. After ending the run the file will be set read-only to avoid accidental overwriting. The file name `/dev/null` [80] suppresses the output file.
If you want to write into the directory from where you are starting your CORSIKA run, you should give: `TELFIL ' '` or `TELFIL " "` (a blank enclosed in apostrophes or quotation marks). Please keep in mind that in FORTRAN an automatic expansion of UNIX names like `'$HOME'` is not possible, rather you should give the fully expanded name of the directory ending with a '/' character.
This keyword is only available in the CERENKOV option together with the IACT option for Cherenkov telescopes.

4.83 Trajectory Selection Flag

`TRAFLG  TLOGIC`

Format = (A6, L), Default = T
TLOGIC : If .true., the zenith and azimuth angles of each shower event are determined by the TRAJECT option to follow a source object movement in the sky. The determination of the angles by other options (VIEWCONE, VOLUMECORR, or VOLUMEDET) or by the keywords THETAP and PHIP is disregarded.
This keyword is only available in the TRAJECT option.

4.84 Source Position Parameters

`SRCPOS  RA  DEC`

Format = (A6, 2F), Defaults = 5.57, 22.0
RA : Defines the right ascension (in hours) of the simulated source in equatorial coordinates.
DEC : Defines the declination (in °) of the simulated source in equatorial coordinates.
The default values give the Crab Nebula.
Limits are: $0. <$ RA $< 24.$; $-90. <$ DEC $< +90.$.
This keyword is only available in the TRAJECT option.

[79] Details on the `eventio` format may be found in the documentation supplied with the 'bernlohr' package.
[80] The existence of `/dev/null` is assumed, see footnote of Sect. 4.68.

4.85 Trajectory Time Parameters

```
TRATM   TYEAR TMONTH TDAY THOUR TMINUTE TSECOND TDURATION
```

Format = (A5, 7I), Defaults = 2000, 1, 1, 21, 0, 0, 3600
TYEAR, TMONTH, TDAY, THOUR, TMINUTE, TSECOND : Define the start time of the observation (in year, month, day, hour, min, sec).
TDURATION : Duration of observation (in sec).
Limits are: Only valid values for date and time are admitted; $0 <$ TDURATON.
This keyword is only available in the TRAJECT option.

4.86 Lateral Telescope Site Parameters

```
TLAT    TLATDGR   TLATMIN   TLATSEC   TLATDIR
```

Format = (A4, 3F, A1), Defaults = 28., 45., 42.462, 'N'
TLATDGR : Latitude of the telescope site (in °).
TLATMIN : Latitude of the telescope site (in min).
TLATSEC : Latitude of the telescope site (in sec).
TLATDIR : Direction North = 'N', South = 'S' of the latitude of the telescope site.
The default values give the site of the MAGIC telescope.
Limits are: 0. $\leq$ TLATDGR $\leq$ 90.; 0. $\leq$ TLATMIN $\leq$ 60.; 0. $\leq$ TLATSEC $\leq$ 60.; TLATDIR = 'N' or 'S'.
This keyword is only available in the TRAJECT option.

4.87 Longitudinal Telescope Site Parameters

```
TLONG    TLONGDGR   TLONGMIN   TLONGSEC   TLONGDIR
```

Format = (A5, 3F, A1), Defaults = 17., 53., 26.525, 'W'
TLONGDGR : Longitude of the telescope site (in °).
TLONGMIN : Longitude of the telescope site (in min).
TLONGSEC : Longitude of the telescope site (in sec).
TLONGDIR : Direction East = 'E', West = 'W' of the longitude of the telescope site.
The default values give the site of the MAGIC telescope.
Limits are: 0. $\leq$ TLONGDGR $\leq$ 180.; 0. $\leq$ TLONGMIN $\leq$ 60.; 0. $\leq$ TLONGSEC $\leq$ 60.; TLONGDIR = 'E' or 'W'.
This keyword is only available in the TRAJECT option.

4.88 Geomagnetic Declination of Telescope

GEODEC GEODECL

Format = (A6, F), Default = -6.35
GEODECL : Defines the geomagnetic declination [81], i.e. the directional deviation of the magnetic North from the geographic North (in °).
The default value corresponds with the site of the MAGIC telescope.
Limit is: -45. $\leq$ GEODECL $\leq$ +45. .
This keyword is only available in the TRAJECT option.

4.89 Trajectory Broadening Parameter

TRARAD TRAD

Format = (A6, F), Default = 0.
TRAD : Defines the radius of a spread around the calculated trajectory (in arcmin).
Limit is: 0. $\leq$ TRAD $\leq$ 3600. .
This keyword is only available in the TRAJECT option.

4.90 Write Data Base File

DATBAS FDBASE

Format = (A6, L), Default = F
FDBASE : If .true., all essential run parameters are written to the file '*DATnnnnnn*.dbase' (rsp. '$DATnnnnnn$.info' in the AUGERINFO option) onto the output directory DSN (keyword DIRECT page 87). This file may be used to build a data base for examining the content of an air shower library (page 121).
This keyword is only effective in the UNIX options.

4.91 User Name

USER USER

Format = (A4, A20), Defaults = ' '
USER : A user name is read in to be written to the '$DATnnnnnn$.dbase' file. Lower case characters of USER are not converted to capitals.

[81] The sign of the declination is defined positive for eastward declination, negative for westward declination, see also Ref. [65]

4.92 Host Name

HOST HOST

Format = (A4, A20), Defaults = ‘ ’
HOST : A host name is read in to be written to the ‘$DATnnnnnn$.dbase’ file. Lower case characters of HOST are not converted to capitals.

4.93 Debugging

DEBUG DEBUG MDEBUG DEBDEL NDEBDL

Format = (A5, L, I, L, I), Defaults = F, MONIOU, F, 100000
DEBUG : If .false., debugging is disabled. If .true., additional output for debugging purposes is given on logical unit MDEBUG.
MDEBUG : Logical unit where to write debugging information. Make sure that your selection of MDEBUG is not conflicting with existing definitions (see Table 1 page 24).
DEBDEL : If .true., the debugging printouts are activated after NDEBDL particles above the ECTMAP energy have been printed. If .false., delayed debugging is disabled. This feature helps to trace run time errors that have occurred in long simulation runs.
NDEBDL : See DEBDEL

4.94 Debugging EGS

EGSDEB JCLOCK

Format = (A6, I), Default = 2147483647
JCLOCK : Counter for delayed start of EGS4 debugging. After activation of debug by DEBUG or by NDEBDL (see Sect. 4.93 above) each pass of subroutine $electr$ or $photon$ is counted. If the counter exceeds JCLOCK, the debug statements within the EGS4 portion are activated. This output appears on the unit MDEBUG.

4.95 FLUKA Printing

FLUDBG FFLUDB

Format = (A6, L), Default = F
FFLUDB : If .true. the two files ‘$DATnnnnnn$.flout’ for additional information on the parameters used by FLUKA and ‘$DATnnnnnn$.flerr’ on possible FLUKA error messages are written onto the output directory DSN (keyword DIRECT page 87). If by the keyword DIRECT the directory `/dev/null` has been specified, the two files are opened within the directory from

where the job has been started.
If .false. in the LINUX option the two files are written to `fort`.11 rsp. `fort`.15 and may be redirected at runtime to `/dev/null` using shell commands like `setenv FORT.11 /dev/null` rsp. `setenv FORT.15 /dev/null`. Without the LINUX option the files are opened directly to the directory `/dev/null` [82].
This keyword is only available in the FLUKA option.

4.96 GHEISHA Debugging

`GHEIDB` `GHEISDB`

Format = (A6, L), Default = F
GHEISDB : If .true., in the DEBUG case also the GHEISHA routines produce debug output. This output appears on the unit MDEBUG.
This keyword is only available in the GHEISHA option.

4.97 URQMD Debugging

`URQMD` `FURQMD` `IUDEBUG`

Format = (A5, L, I), Default = T, 0
FURQMD : If .true., the UrQMD routines are used for the low-energy hadronic interactions. If .false., the program will stop.
IUDEBUG : If > 0, in the DEBUG case also the routines of UrQMD produce some output. With increasing value of IUDEBUG this printout becomes more and more detailed. This output appears on the unit MDEBUG.
Limit is: $0 \leq$ IUDEBUG ≤ 3 .
This keyword is only available in the URQMD option.

4.98 PYTHIA Printing

`PYTHIA` `IFLGPYW` `IFLGPYE`

Format = (A6, 2I), Default = 0, 0
IFLGPYW : If set > 0 the printing of PYTHIA warnings is enabled. IFLGPYW gives the number of warnings after which PYTHIA becomes silent (see MSTU(26) of Pythia manual [30]).
IFLGPYE : If set > 0 the printing of PYTHIA errors is enabled. IFLGPYE gives the number of warnings after which PYTHIA becomes silent (see MSTU(22) of Pythia manual [30]).
Limits are: $0 \leq$ IFLGPYW; $0 \leq$ IFLGPYE.
This keyword is only available in the DPMJET, CHARM, or TAULEP options.

[82] The existence of `/dev/null` is assumed, see footnote of Sect. 4.68 [page 87.

4.99 Charm Interaction Cross Section

```
SIGMAQ    SIGMAQ(i), i=1... 4
```

Format = (A6, F4), Defaults = 0., 0., 0., 0.
SIGMAQ(i) : The interaction cross-sections for charmed mesons ($i = 1$), charmed baryons ($i = 2$), bottom mesons ($i = 3$), and bottom baryons ($i = 4$) are specified. For SIGMAQ(i) = 0., the parameterizations of Ref. [42] are used; for SIGMAQ(i) $\neq$ 0., the energy independent cross-section of SIGMAQ(i) is used for the corresponding projectile.
Limits are: $0. \leq$ SIGMAQ(i).
This keyword is only available in the CHARM option.

4.100 Charm Interaction Selection

```
PROPAQ    PROPMOD
```

Format = (A6, I), Default = 1
PROPMOD : If set to 1, the PYTHIA extension [42] is used to treat the interactions of charmed projectiles.
If set to 0, the QGSJET01d interaction model is used to treat the interactions of charmed projectiles.
Limits are: $0 \leq$ PROPMOD ≤ 1.
This keyword is only available in the CHARM option in combination with the QGSJET option.

4.101 Cherenkov Debugging

```
CDEBUG    LCERDB
```

Format = (L), Default = F
LCERDB : If .false., Cherenkov debug output is disabled. If .true., the Cherenkov routines produce debug output. This output appears on the unit MDEBUG.
This keyword is only available in the CERENKOV option.

4.102 Interaction Test Target Definition

```
INTTST    ITTAR    MCM
```

Format = (A6, 2I), Defaults = 0, 0
ITTAR : Defines the target for the interaction test option[83]: 1 = proton; 2 = neutron; 9 = Beryllium; 12 = Carbon; 14 = Nitrogen; 16 = Oxygen; 40 = Argon; 99 = air.
MCM : Defines the reference system for which the interaction products are plotted. 1 = rest

[83] For the DPMJET high-energy model and ITTAR $<$ 14 the Glauber data file $glaubint$.glb has to be used instead of the $glaubtar$.glb file.

system of 1 target nucleon and 1 projectile nucleon[84]; 2 = laboratory system; 3 = rest system of all secondary particles (but not the spectators).
This keyword is only available in the INTTEST option.

4.103 Interaction Test Decay

```
INTDEC    LPI0   LETA   LHYP   LK0S
```

Format = (A6, 4L), Defaults = T, T, T, T
LPI0 : If .true. the π° particles decay before gathering them in the interaction test.
LETA : If .true. the η particles decay before gathering them in the interaction test.
LHYP : If .true. all hyperons decay before gathering them in the interaction test.
LK0S : If .true. the K_s° particles decay before gathering them in the interaction test.
This keyword is only available in the INTTEST option.

4.104 Interaction Test Spectator Definition

```
INTSPC    LSPEC
```

Format = (A6, L), Default = F
LSPEC : If .true. spectators are plotted, if .false. spectators are not plotted in the interaction test.
This keyword is only available in the INTTEST option.

4.105 Interaction Test Diffraction Flag

```
DIFOFF    NDIF
```

Format = (A6, I), Default = 0
NDIF : Allows to select diffractive or non-diffractive interactions only[85]. 0 = diffractive and non-diffractive interactions mixed; 1 = non-diffractive interactions only; 2 = diffractive interactions only. With the QGSJET01d interaction model NDIF = 2 cannot be selected. With the EPOS, NEXUS and QGSJET-II models only NDIF = 0 is possible.
This keyword is only available in the INTTEST option.

[84] In photo-nuclear interactions the cm-system of the photon with one nucleon.

[85] For photo-nuclear interactions the meaning is: 0 = multi-hadron production and vector meson production mixed; 1 = no vector meson production, only multi-hadron production; 2 = only vector meson production, no multi-hadron production.

4.106 Interaction Test Trigger Condition

`TRIGGER NTRIG`

Format = (A7, I), Default = 0
NTRIG : Allows to select various trigger conditions for comparison with experimental data:
0 = accepts all events;
1 = accepts only events according to the UA5-experiment [69] trigger;
2 = accepts only events according to the CDF-experiment [70] trigger.
3 = accepts only events according to the P238-experiment [71] trigger.
NTRIG $\neq$ 0 may be combined only with NDIF = 0 .
This keyword is only available in the INTTEST option.

4.107 Interaction Test Histogram Output

`HISTDS HISTDS`

Format = (A6, A120), Defaults = 'histo.corsika.inttest'
HISTDS : May be used to specify a name of the histogram output directory and data file. Lower case characters of HISTDS are not converted to capitals. Do not use capitals as the HBOOK routines use only lower case characters. The data file name is extended by a string containing information about projectile, target, energy, and the type of interaction which has been selected. At the end of the data file name .hbook is appended such that the total data file name would look like
histo.corsika.inttest.p0014t14e100e3.diffractive.hbook for a proton projectile on nitrogen target with a lab energy of 100E3 GeV including diffractive events.
If you want to write into the directory from where you are starting your CORSIKA run, you should give: `HISTDS ' '` or `HISTDS " "` (a blank enclosed in apostrophes or quotation marks). Please keep in mind that in FORTRAN an automatic expansion of UNIX names like `'$HOME'` is not possible, rather you should give the fully expanded name of the directory ending with a '/' character.
Limit is: HISTDS must not begin with a ~ (tilde) character.
This keyword is only available in the INTTEST option.

4.108 Plot Output

`PLOTSH PLOTSH`

Format = (A6, L), Default = F
PLOTSH : If .true., the track start- and endpoints of the electromagnetic, muonic, and hadronic component of the shower are given out separately and may be used to plot the shower development.

This keyword is only available in the PLOTSH and PLOTSH2 option.

4.109 Plot Axes Definition

```
PLAXES    X1    X2    Y1    Y2    Z1    Z2
```

Format=(A6,6F), Defaults = -500000., 500000., -500000., 500000., 0., 3000000.
X1, X2 : They denote the X-axis range (in cm) to be plotted in the map.
Y1, Y2 : They denote the Y-axis range (in cm) to be plotted in the map.
Z1, Z2 : They denote the Z-axis range (in cm) to be plotted in the map.
The point of first interaction determines the zero point of the X and Y axes (see Fig. 1 page 102). Depending on the choice of these parameters, the whole shower may be visualized, or one can 'zoom in' on interesting regions of the shower.
Limits are: X1 $<$ X2, Y1 $<$ Y2, Z1 $<$ Z2 .
This keyword is only available in the PLOTSH2 option.

4.110 Plot Energy Cut Definition

```
PLCUTS    ELCUTS(1...4)    TCUT    FBOXCUT
```

Format=(A6,5F,L), Defaults 0.3, 0.3, 0.003, 0.003, 100000., F
ELCUTS(1...4) : ELCUTS denote the energy cuts in the same order as those for the keyword ECUTS (hadrons, muons, electrons, photons) (see page 81).
TCUT : This is an upper bound on the time (in ns) passed since the first interaction. If, at the end point of a track, the time is above TCUT, the track is **not** plotted. This cut allows a visualization of the shower development.
FBOXCUT : This flag determines whether only track segments inside the three-dimensional box given by the axis ranges should be plotted. If .true., all track segments whose endpoints both fall outside this box are not plotted.
This keyword is only available in the PLOTSH2 option.

4.111 End of Steering

```
EXIT
```

Format = (A4)
This keyword ends the keyword input.

5 Input Example

The keyword input file for a CORSIKA run including QGSJET and CERENKOV options may look like the following list.

```
RUNNR  1                        number of run
EVTNR  100400                   no of first shower event
SEED   100401  0  0             seed for hadronic part
SEED   100402  0  0             seed for EGS4 part
SEED   100403  0  0             seed for Cherenkov part
NSHOW  10                       no of showers to simulate
PRMPAR 5626                     primary particle code (iron)
ERANGE 2.00E4  4.00E4           energy range of primary (GeV)
ESLOPE -2.7                     slope of energy spectrum
THETAP 0.  10.                  range zenith angle (deg)
PHIP   -180.  180.              range azimuth angle (deg)
QGSJET T  0                     QGSJET for high energy & debug level
QGSSIG T                        QGSJET cross-sections enabled
HADFLG 0  0  0  0  0  2         HDPM interact.flags & fragmentation flag
ELMFLG T  T                     elmag. interaction flags NKG, EGS4
STEPFC 1.                       multiple scattering step length factor
RADNKG 200.E2                   outer radius (cm) of NKG elect. distrib.
MAGNET 20.4  43.23              magnetic field central Europe (/uT)
ECUTS  .3  .3  .015  .015       energy cuts: hadr. muon elec. phot. (GeV)
LONGI  T  20.  T   T            longitud, stepsize(g/cm^2), fit, out
MUMULT T                        muon multiple scattering by Moliere
MUADDI T                        additional muon information
OBSLEV 110.E2                   observation level (cm)
ARRANG 18.25                    angle between north to array-grid (deg)
MAXPRT 10                       max. no of printed events
ECTMAP 1.E2                     printout gamma factor cut
DIRECT /home/user/corsika/run/  directory of particle output
CERARY 10 8 1200. 1500. 80. 50. Cherenkov detector grid (cm)
CWAVLG 300.  450.               Cherenkov wavelength band (nm)
CERSIZ 5.                       bunch size Cherenkov photons
CERFIL F                        Cherenkov output file
CSCAT  5  1000.  1000.          scatter Cherenkov events (cm)
DATBAS T                        write data base file
USER   you                      user name for data base file
HOST   your_host                host name for data base file
DEBUG  F  6  F  999999999       debug flag, log. unit, delayed debug
EXIT
```

6 Coordinate System

The coordinates in CORSIKA are defined with respect to a Cartesian coordinate system with the positive X-axis pointing to the magnetic North, the positive Y-axis to the West, and the Z-axis upwards. The origin is located at sea level. This definition is necessary, because the Earth's magnetic field is taken into account. By default the magnetic field is implemented for the location of Karlsruhe (49° N, 8° E) as described at the keyword MAGNET (page 69). The

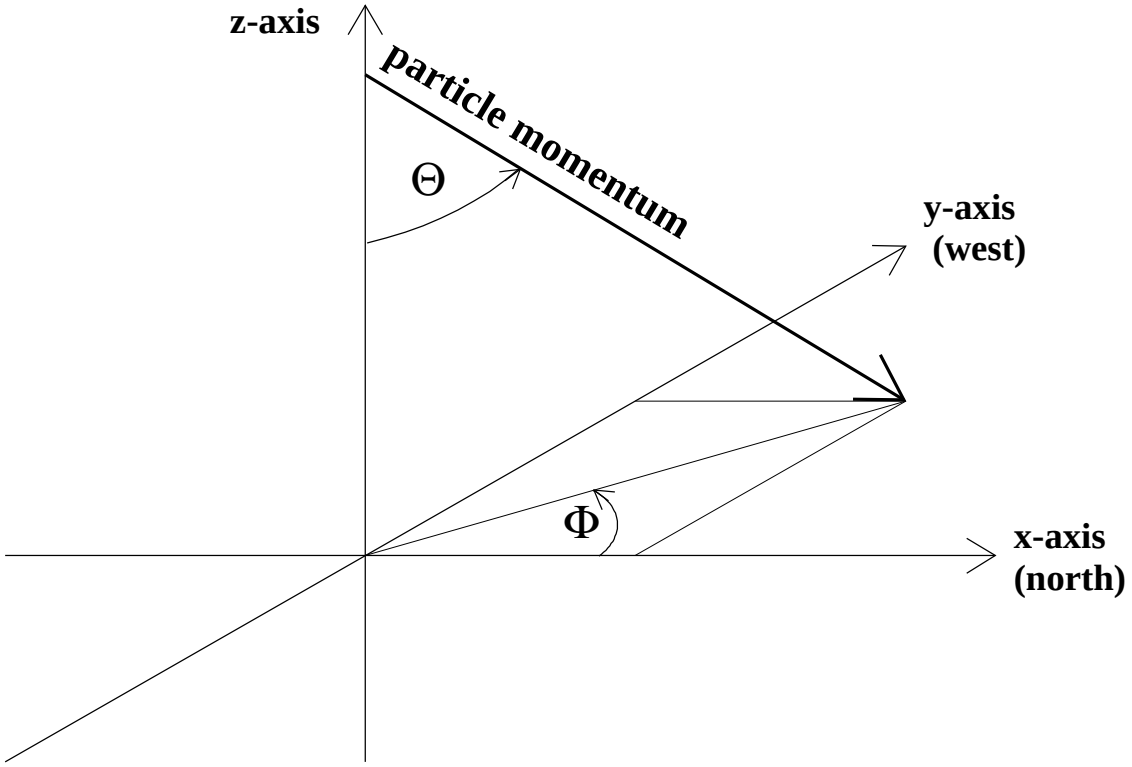

Figure 1: Coordinate system in CORSIKA.

zenith angle θ of a particle trajectory is measured between the particle momentum vector and the negative Z-axis, and the azimuthal angle ϕ between the positive X-axis and the horizontal component of the particle momentum vector (i.e. with respect to North) proceeding counter-clockwise. This is shown in Fig. 1.
In the EGS4 part and in CONEX the definitions of the coordinate system differ from that used in CORSIKA (see the in-line comments in these program parts). For EGS4 the positive Z-axis points downwards, the positive Y-axis points to East, the X-axis remains unchanged. For CONEX the positive X-axis points to East, the positive Y-axis points to North, the Z-axis remains unchanged.
Attention: The definition of the coordinate system in CORSIKA disagrees from definitions of other air-shower simulation programs (e.g. AIRES [72]) and from the conventions of the Auger experiment !!

7 Units in CORSIKA

Within CORSIKA uniform units for the various dimensions are used as far as possible. But there are deviations at that program parts which are coupled to CORSIKA. Table 3 gives an overview on the used units.

Quantity	CORSIKA	EGS4	FLUKA GHEISHA UrQMD HERWIG PYTHIA	SIBYLL	DPMJET EPOS NEXUS QGSJET VENUS	CONEX
length	cm	cm				m
energy	GeV	MeV	GeV	GeV [3])	GeV	GeV
mass	GeV	MeV	GeV	GeV	GeV	GeV
time	sec [1])	sec				sec
magn. field	μT					[4])
density	g/cm^3	g/cm^3				g/cm^3
mass overburden	g/cm^2					g/cm^2
angle	rad [2])	rad				rad
wavelength	nm					
	[1]) For output files also nsec is used. [2]) For in- and output files also $^\circ$ is used. [3]) In some subroutines also TeV is used. [4]) No Earth magnetic field considered.					

Table 3: Units used in CORSIKA and the coupled programs.

8 Particles in CORSIKA

CORSIKA is able to treat the particles that are listed in Table 4. The particle codes have in general been chosen according to the convention in the GEANT detector simulation code [32]. Exceptions are vector mesons (ω and ρ) and resonances (K^* and Δ), the η particles which are split in 4 types according to their decays in the HDPM routines, the different types of neutrinos (to be generated in the NEUTRINO option), the nuclei, and the Cherenkov photons. The codes of the charmed particles correspond with those of DPMJET. The particles with codes between 116 and 173 are only available in the CHARM or TAULEP option. Cherenkov photons can not be a primary particle for an air shower simulation.

[86] Only available in MUPROD option.

Particle identifications			
Identification	Particle	Identification	Particle
1	γ	51	ρ°
2	e^+	52	ρ^+
3	e^-	53	ρ^-
		54	Δ^{++}
5	μ^+	55	Δ^+
6	μ^-	56	Δ°
7	π°	57	Δ^-
8	π^+	58	$\overline{\Delta}^{--}$
9	π^-	59	$\overline{\Delta}^-$
10	K_L°	60	$\overline{\Delta}^\circ$
11	K^+	61	$\overline{\Delta}^+$
12	K^-	62	$K^{*\circ}$
13	n	63	K^{*+}
14	p	64	K^{*-}
15	$\overline{p}$	65	$\overline{K^*}^\circ$
16	K_S°	66	ν_e
17	η	67	$\overline{\nu}_e$
18	Λ	68	ν_μ
19	Σ^+	69	$\overline{\nu}_\mu$
20	Σ°		
21	Σ^-	71	$\eta \to \gamma\gamma$
22	Ξ°	72	$\eta \to 3\pi^\circ$
23	Ξ^-	73	$\eta \to \pi^+\pi^-\pi^\circ$
24	Ω^-	74	$\eta \to \pi^+\pi^-\gamma$
25	$\overline{n}$	75	μ^+ add. info.
26	$\overline{\Lambda}$	76	μ^- add. info.
27	$\overline{\Sigma}^-$		
28	$\overline{\Sigma}^\circ$	85	decaying μ^+ at start[86]
29	$\overline{\Sigma}^+$	86	decaying μ^- at start[86]
30	$\overline{\Xi}^\circ$		
31	$\overline{\Xi}^+$	95	decaying μ^+ at end[86]
32	$\overline{\Omega}^+$	96	decaying μ^- at end[86]
50	ω		

Table 4: Particle identifications as used in CORSIKA (to be continued).

Particle identifications (continued)			
Identification	Particle	Identification	Particle
116	D°	155	$\overline{\Xi_c'}^{-}$
117	D^{+}	156	$\overline{\Xi_c'}^{\circ}$
118	$\overline{D}^{-}$	157	$\overline{\Omega_c}^{\circ}$
119	$\overline{D}^{\circ}$		
120	D_s^{+}	161	Σ_c^{*++}
121	$\overline{D_s}^{-}$	162	Σ_c^{*+}
122	η_c	163	$\Sigma_c^{*\circ}$
123	$D^{*\circ}$		
124	D^{*+}	171	$\overline{\Sigma_c^{*}}^{--}$
125	$\overline{D^{*}}^{-}$	172	$\overline{\Sigma_c^{*}}^{-}$
126	$\overline{D^{*}}^{\circ}$	173	$\overline{\Sigma_c^{*}}^{\circ}$
127	D_s^{*+}		
128	$\overline{D_s^{*}}^{-}$	176	B°
		177	B^{+}
130	J/ψ	178	$\overline{B}^{-}$
131	τ^{+}	179	$\overline{B}^{\circ}$
132	τ^{-}	180	B_s°
133	ν_τ	181	$\overline{B_s}^{\circ}$
134	$\overline{\nu}_\tau$	182	B_c^{+}
		183	$\overline{B_c}^{-}$
137	Λ_c^{+}	184	Λ_b°
138	Ξ_c^{+}	185	Σ_b^{-}
139	Ξ_c°	186	Σ_b^{+}
140	Σ_c^{++}	187	Ξ_b°
141	Σ_c^{+}	188	Ξ_b^{-}
142	Σ_c°	189	Ω_b^{-}
143	$\Xi_c'^{+}$	190	$\overline{\Lambda_b}^{\circ}$
144	$\Xi_c'^{\circ}$	191	$\overline{\Sigma_b}^{+}$
145	Ω_c°	192	$\overline{\Sigma_b}^{-}$
		193	$\overline{\Xi_b}^{\circ}$
149	$\overline{\Lambda_c}^{-}$	194	$\overline{\Xi_b}^{+}$
150	$\overline{\Xi_c}^{-}$	195	$\overline{\Omega_b}^{+}$
151	$\overline{\Xi_c}^{\circ}$		
152	$\overline{\Sigma_c}^{--}$		
153	$\overline{\Sigma_c}^{-}$		
154	$\overline{\Sigma_c}^{\circ}$		
$\mathrm{A}\times 100+\mathrm{Z}$	nucleus of Z protons and $\mathrm{A}-\mathrm{Z}$ neutrons ($2\leq \mathrm{A}\leq 56$)		
9900	Cherenkov photons on particle output file		

Table 4: (continued) Particle identifications as used in CORSIKA.

9 Running the CORSIKA Program

Depending on the program options CORSIKA needs at minimum $\approx$ 92 Mbyte memory for SIBYLL with GHEISHA without other options. Other high-energy or low-energy models need additionally memory. Especially EPOS, QGSJET-II, and FLUKA need a huge amount of memory. Empirical (approximated) values of required memory for 64-bit LINUX computers are given in Table 5.

Option	SIBYLL + GHEISHA	DPMJET	EPOS	QGSJET01	QGSJET-II	FLUKA	UrQMD	CONEX
approx. Memory (Mbyte)	92	+ 47	+ 551	+ 4	+ 256	+ 431	+ 5	+ 657

Table 5: Required memory for various CORSIKA options (64-bit LINUX).

A simplified flow diagram of CORSIKA is given in Appendix B (page 133).

The sequence of the initializing procedures is given in Appendix C (page 134).

CORSIKA runs fastest (full simulation adopted without THINning) when using no EGS4, no DPMJET, no EPOS, no NEXUS, no VENUS, no FLUKA, no UrQMD, no NEUTRINO, no Cherenkov light generation, no CURVED, and no options producing plots, histograms or ROOT files. For this program version the computing time on an Intel 6600 Core2Duo (with 2.4 GHz) is $\approx$ 7.3 sec per shower for primary protons of energy 10^6 GeV, vertical incidence, NKG enabled, with one observation level at 110 m a.s.l. and with the hadron and muon energy cut at 0.3 GeV. Under the same conditions an iron induced shower consumes $\approx$ 10.6 sec. The computing time scales roughly with the primary energy. The full EGS4 option with longitudinal profile is roughly 200 times slower than the fastest version mentioned above (with ELCUT(3,4) at 3 MeV). There is not much experience what the time consumption is for the CERENKOV option, but it will be much higher. SIBYLL and QGSJET01 run fastest with similar short computing times. EPOS or NEXUS (NKG, no EGS) need $\approx$ 50 times the computing time of SIBYLL or QGSJET01. FLUKA needs $\approx$ 7 times computing time of GHEISHA (+ QGSJET01/SIBYLL), UrQMD needs $\approx$ 40 times that of GHEISHA.
In case of THINning (Sect. 3.5.29 page 54) the computing time strongly depends on the energy fraction below which thinning becomes active, see Table 2 page 56.
The interested CORSIKA user may find CPU-times for various interaction models and options in Ref. [26].

The particle output format is described in Sect. 10.2 (page 110 ff.). For each particle that reaches an observation level 7 words (for THINning 8 words) with 4 bytes each are stored on the particle output file. Proton showers at 10^6 GeV deliver at sea level roughly 0.8 MB particle output when calculated with the NKG option ($E_h, E_\mu > 0.3$ GeV). With EGS4 about

30 MB output are written due to the huge amount of photons and electrons that are explicitly stored ($E_e, E_\gamma > 0.003$ GeV)[87]. The particle output in the CERENKOV option is additionally increased as the Cherenkov photons are stored, either separately on the Cherenkov output file, or together with the particles on the particle output file.
While running, the interactions produce plenty of secondary particles which are stored in an intermediate stack which is foreseen for 200000 particles. After completing an interaction all particles (in case of THINning only a fraction of them) are moved to the internal stack. The size of this internal stack is large enough to keep 680 (in case of thinning 624) particles. If the size of the internal stack is full, 680 (624) particles are temporarily added to the external stack. If the internal stack is empty 340 (312) particles are re-read from the external stack. Only half of the internal stack is read back to avoid a permanent shifting of data from and to the internal stack if it is just full and the number of secondaries oscillates with a small amplitude around 680 (624).

[87] These energy cuts correspond with the sensitivities of the KASCADE array detectors.

10 Outputs

There are two major output files produced by a simulation run. The control printout (.lst file) allows to survey the simulation run. The particle output file is written to the data file '$DATnnnnnn$' with $nnnnnn$ being the run number specified in the keyword RUNNR (page 60). This file becomes very large when simulating showers in great detail (EGS4, low thresholds, ...). In the CERENKOV version an additional file '$CERnnnnnn$' might be written. Optionally a tabular output (γ, $e^{\pm}$, and $\mu^{\pm}$ particles at ground and longitudinal development of charged particles) is written out to the file '$DATnnnnnn$.tab'. Further on the longitudinal distributions of particle numbers and energy deposits may be written to file '$DATnnnnnn$.long'. The output file '$DATnnnnnn$.dbase' (rsp. '$DATnnnnnn$.info') is destinated to be used in a data base for examining the content of an air shower library.
If the INCLINED option (Sect. 3.5.14 page 47) is used in combination with the INCLIN keyword (page 85) an output file '$DATnnnnnn$.inclined' is written, which has the same binary structure as a standard CORSIKA particle output file '$DATnnnnnn$'. The INCLINED option internally makes use of the COAST package.

10.1 Control Printout (.lst File Output)

The simulation run produces a printout (.lst file, unit MONIOU, by default standard output)[88] that allows to control the simulation and informs about the general run, the program version with interaction model, the selected options, steering keywords, physical constants, the used atmospheric model, and the primary particle (about 200 lines). For each shower it prints roughly 400 lines containing the random number generator status, time at beginning of a shower, the primary particle at the place of the first interaction, the number of secondaries reaching the observation levels with energies above ECTMAP (page 87), the stack statistics, internal and external stack usage, energy-multiplicity and energy-elasticity relations, interaction statistics for nucleons, pions, kaons, and strange baryons per kinetic energy interval, an interaction length statistics for the above particles and a decay statistics for muons, summaries of secondaries for each observation level, NKG electromagnetic shower information, and the longitudinal shower development.
The NKG output (keyword ELMFLG, page 76) comprises a table on the longitudinal development of the electromagnetic shower component giving every 100 g/cm^2 the number of electrons and the longitudinal pseudo-age parameter[89]. For the lateral electron distribution the densities (in $electrons$/cm^2) are calculated on a grid of 80 points (8 directions separated by 45° with 10 distances between 1 m and RADNKG (page 76) for each direction) and the lateral pseudo-ages for those various distances are determined. The lateral distribution is calculated only for the lowest two observation levels.
For the longitudinal development (keyword LONGI page 82) three tables are generated:

[88] Renaming the standard (log) output to '$DATnnnnnn$.lst' and redirecting it to the directory specified by the keyword DIRECT (page 87) is convenient (page 109) as by the shell commands 'dir' or 'ls -l' all files belonging to one run are displayed consecutively which facilitates book-keeping.

[89] See footnote page 35.

- The first one gives the particle numbers of $\gamma, e^+, e^-, \mu^+, \mu^-$, hadrons, all charged, nuclei, neutrinos, and Cherenkov photons as function of atmospheric depth.
- The second one reflects the energy content within the various particle species γ, e^+, e^-, μ^+, μ^-, hadrons, charged particles, nuclei, neutrinos, and the energy sum as function of atmospheric depth. For all particle species the particle rest mass is included within the energy.
- The third table gives the energy dissipated within the atmosphere specified for various processes: Energy contained within γ's falling below (energy or angular) cut, ionization energy loss of $e^{\pm}$, energy contained within $e^{\pm}$ falling below (energy or angular) cuts, ionization energy loss of $\mu^{\pm}$, energy contained within $\mu^{\pm}$ falling below (energy or angular) cuts, ionization energy loss of hadrons, energy contained within hadrons falling below (energy or angular) cuts, energy taken away by ν's, and the energy sum for each depth bin. For $\pi^{\pm}$ and $\mathrm{K}^{\pm}$ falling below the energy or angular cuts 1/4 of the energy is attributed to the hadronic energy, while 3/4 is attributed to neutrinos. For the neutral $\mathrm{K}^{\circ}_{\mathrm{L,S}}$ this ratios are 1/2.

  The last bin of the cut energies reflects the energy content of particles arriving at detector level. In this table the rest mass of nucleons and electrons is **not** counted, while the rest mass of the corresponding anti-particles (which might undergo annihilation) is counted **twice**. Energies of unstable hadrons and muons are added up including their rest mass. This counting is necessary to respect the target nucleons or electrons involved into the shower development, thus enabling a correct energy balancing.

For writing out the longitudinal tables to the '$DATnnnnnn$.long' file or as 'LONG' blocks to the particle output file '$DATnnnnnn$' see the FLONGOUT flag (page 82).
The control printout contains as well all kinds of warnings and error messages. System errors may be redirected in UNIX systems[90] to the standard output[91] (.lst file) by the & character following immediately the $>$ character as given in the example:

```
./corsika <inputs >& /home/user/corsika/run/DATnnnnnn.lst
```

assuming that all output should go to the directory /home/user/corsika/run/ as given in the example on page 101. Users are advised to check this printout carefully for any indications of problems during the run time and keeping it together with the particle output for later consultation.
When errors occur the DEBUG option may help in localizing the bug. This option entails a very detailed printout of the simulation process that easily fills large disks when enabled for many showers.
All these informations are printed per event. For low energy primaries and high statistics this printout per event may not be necessary and can be suppressed by selecting a maximum number of showers to be printed (keyword MAXPRT page 86). At the end of each run a short run summary is printed with similar tables as for single showers but averaged over all showers of the actual run.

[90] The usage of the korne-shell is assumed.
[91] For the naming of standard output see the footnotes page 25 and 108.

10.2 Normal Particle Output

The particle and Cherenkov photon output files contain the information about the simulation run and about all particles reaching observation levels. This is what has to be analyzed for detailed energy spectra and distributions. These files are written to the directory DSN (as defined by keyword DIRECT, page 87) as 'sequential' 'unformatted' FORTRAN files. They may be read by the utility programs *corsikaread*.f rsp. *corsikaread_thin*.f available in the subsubdirectory `src/utils/`. The particle output file and the Cherenkov photon output file are structured as shown in Table 6, with the sub-blocks as given in Tables 7 to 14. All quantities are written as single precision real numbers by the subroutine $tobuf$.c (independently of 32-bit or 64-bit processors).

Block structure
RUN HEADER nrun
EVENT HEADER 1
DATABLOCK
DATABLOCK
...
...
(LONG 1:1)
...
(LONG 1:n)
EVENT END 1
EVENT HEADER 2
DATABLOCK
DATABLOCK
...
...
(LONG 2:1)
...
(LONG 2:n)
EVENT END 2
...
...
EVENT HEADER nevt
DATABLOCK
DATABLOCK
...
...
(LONG nevt:1)
...
(LONG nevt:n)
EVENT END nevt
RUN END nrun

Table 6: Block structure of the particle and Cherenkov photon output files. (The LONG blocks eventually occur only in the particle output file.)

Run header sub-block: (once per run)	
No. of word	Contents of word (as real numbers R*4)
1	'RUNH'
2	run number
3	date of begin run (yymmdd)
4	version of program
5	number of observation levels (maximum 10)
$5+i$	height of observation level i in cm
16	slope of energy spectrum
17	lower limit of energy range
18	upper limit of energy range
19	flag for EGS4 treatment of em. component
20	flag for NKG treatment of em. component
21	kin. energy cutoff for hadrons in GeV
22	kin. energy cutoff for muons in GeV
23	kin. energy cutoff for electrons in GeV
24	energy cutoff for photons in GeV
	physical constants and interaction flags:
$24+i$	C(i), $i = 1, 50$
75	XPINCL X-displacement of inclined observation plane
76	YPINCL Y-displacement of inclined observation plane
77	ZPINCL Z-displacement of inclined observation plane
78	THINCL θ angle of normal vector of inclined observation plane
79	PHINCL ϕ angle of normal vector of inclined observation plane
$79+i$	0, $i = 1, 14$ (no longer used)
93	NSHOW number of showers to be generated
$94+i$	CKA(i), $i = 1, 40$
$134+i$	CETA(i), $i = 1, 5$
$139+i$	CSTRBA(i), $i = 1, 11$
$150+i$	0, $i = 1, 97$ (no longer used)
248	XSCATT scatter range in x direction for Cherenkov
249	YSCATT scatter range in y direction for Cherenkov
$249+i$	HLAY(i), $i = 1, 5$
$254+i$	AATM(i), $i = 1, 5$
$259+i$	BATM(i), $i = 1, 5$
$264+i$	CATM(i), $i = 1, 5$
270	NFLAIN
271	NFLDIF
272	NFLPI0$+100\times$NFLPIF
273	NFLCHE$+100\times$NFRAGM

Table 7: Structure of the run header sub-block.

Event header sub-block: (once per event)	
No. of word	Contents of word (as real numbers R*4)
1	'EVTH'
2	event number
3	particle id (particle code or $A \times 100 + Z$ for nuclei)
4	total energy in GeV
5	starting altitude in g/cm^2
6	number of first target if fixed
7	z coordinate (height) of first interaction in cm (negative, if tracking starts at margin of atmosphere, see TSTART)
8	px momentum in x direction in GeV/c
9	py momentum in y direction in GeV/c
10	pz momentum in -z direction in GeV/c (pz is positive for downward going particles)
11	zenith angle θ in radian
12	azimuth angle ϕ in radian
13	number of different random number sequences (max. 10)
$11 + 3 \times i$	integer seed of sequence i
$12 + 3 \times i$	number of offset random calls (mod 10^6) of sequence i
$13 + 3 \times i$	number of offset random calls (/ 10^6) of sequence i
44	run number
45	date of begin run (yymmdd)
46	version of program
47	number of observation levels
$47 + i$	height of level i in cm
58	slope of energy spectrum
59	lower limit of energy range in GeV
60	upper limit of energy range in GeV
61	cutoff for hadrons kinetic energy in GeV
62	cutoff for muons kinetic energy in GeV
63	cutoff for electrons kinetic energy in GeV
64	cutoff for photons energy in GeV
65	NFLAIN
66	NFLDIF
67	NFLPI0
68	NFLPIF
69	NFLCHE
70	NFRAGM

Table 8: Structure of event header sub-block (to be continued).

Event header sub-block: (continued)	
No. of word	Contents of word (as real numbers R*4)
71	x component of Earth's magnetic field in μT
72	z component of Earth's magnetic field in μT
73	flag for activating EGS4
74	flag for activating NKG
75	low-energy hadr. model flag (1.=GHEISHA, 2.=UrQMD, 3.=FLUKA)
76	high-energy hadr. model flag (0.=HDPM,1.=VENUS, 2.=SIBYLL, 3.=QGSJET, 4.=DPMJET, 5.=NEXUS, 6.=EPOS)
77	CERENKOV flag [92]
78	NEUTRINO flag
79	CURVED flag (0=standard, 2=CURVED)
80	computer flag (3=UNIX, 4=Macintosh)
81	lower edge of θ interval (in $^\circ$)
82	upper edge of θ interval (in $^\circ$)
83	lower edge of ϕ interval (in $^\circ$)
84	upper edge of ϕ interval (in $^\circ$)
85	Cherenkov bunch size in the case of Cherenkov calculations
86	number of Cherenkov detectors in x-direction
87	number of Cherenkov detectors in y-direction
88	grid spacing of Cherenkov detectors in x-direction in cm
89	grid spacing of Cherenkov detectors in y-direction in cm
90	length of each Cherenkov detector in x-direction in cm
91	length of each Cherenkov detector in y-direction in cm
92	Cherenkov output directed to particle output file ($= 0.$) or Cherenkov output file ($= 1.$)

Table 9: (continued) Structure of event header sub-block (to be continued).

[92] EVTH(77) has the following contents if converted to an integer with suitable rounding applied:
bit 1 CERENKOV option compiled in
2 IACT option compiled in
3 CEFFIC option compiled in
4 ATMEXT option compiled in
5 ATMEXT option used with refraction enabled
6 VOLUMEDET option compiled in
7 CURVED option compiled in (see also EVTH(79))
9 SLANT option compiled in
11-21 table number for external atmosphere table (but limited to 1023 if the number is larger).

Event header sub-block: (continued)	
No. of word	Contents of word (as real numbers R*4)
93	angle (in rad) between array x-direction and magnetic north
94	flag for additional muon information on particle output file
95	step length factor for multiple scattering step length in EGS4
96	Cherenkov wavelength lower end (in nm)
97	Cherenkov wavelength upper end (in nm)
98	number i of uses of each Cherenkov event
$98+i$	x coordinate of i^{th} core location for scattered events in cm
$118+i$	y coordinate of i^{th} core location for scattered events in cm
139	SIBYLL interaction flag (0.= no SIBYLL, 1.=vers.1.6; 2.=vers.2.1)
140	SIBYLL cross-section flag (0.= no SIBYLL, 1.=vers.1.6; 2.=vers.2.1)
141	QGSJET interact. flag (0.=no QGSJET, 1.=QGSJETOLD, 2.=QGSJET01d, 3.=QGSJET-II)
142	QGSJET X-sect. flag (0.=no QGSJET, 1.=QGSJETOLD, 2.=QGSJET01d, 3.=QGSJET-II)
143	DPMJET interaction flag (0.=no DPMJET, 1.=DPMJET)
144	DPMJET cross-section flag (0.=no DPMJET, 1.=DPMJET)
145	VENUS/NEXUS/EPOS cross-section flag (0=neither, 1.=VENUSSIG, 2./3.=NEXUSSIG, 4.=EPOSSIG)
146	muon multiple scattering flag (1.=Molière, 0.=Gauss)
147	NKG radial distribution range in cm
148	EFRCTHN energy fraction of thinning level hadronic
149	EFRCTHN·THINRAT energy fraction of thinning level em-particles
150	actual weight limit WMAX for thinning hadronic
151	actual weight limit WMAX·WEITRAT for thinning em-particles
152	max. radius (in cm) for radial thinning
153	inner angle of viewing cone VIEWCONE (in °)
154	outer angle of viewing cone VIEWCONE (in °)
155	transition energy high-energy/low-energy model (in GeV)
156	skimming incidence flag (0.=standard, 1.=skimming)
157	altitude (cm) of horizontal shower axis (skimming incidence)
158	starting height (cm)
159	flag indicating that explicite charm generation is switched on
160	flag for hadron origin of electromagnetic subshower on particle tape
161	minimal vertical depth for transfer of particles from CONEX to CORSIKA
162	high-energy threshold for treatment of hadrons by cascade equations in CONEX
163	high-energy threshold for treatment of muons by cascade equations in CONEX
164	high-energy threshold for treatment of em-particles by cascade equations in CONEX
165	low-energy threshold for treatment of hadrons by cascade equations in CONEX
166	low-energy threshold for treatment of muons by cascade equations in CONEX
167	low-energy threshold for treatment of em-particles by cascade equations in CONEX
168	flag for observation level curvature (CURVOUT) (0.=flat, 1.=curved)
169	actual weight limit whmax for thinning hadronic in CONEX
170	actual weight limit wtmax for thinning em-particles in CONEX
171	actual weight limit whmax for sampling hadronic in CONEX
172	actual weight limit wtmax for sampling muons in CONEX
173	actual weight limit wtmax for sampling em-particles in CONEX
174...273	not used

Table 9: (continued) Structure of event header sub-block.

Particle data sub-block : (up to 39 particles, 7 words each)	
No. of word	Contents of word (as real numbers R*4)
$7 \times (n-1)+1$	particle description encoded as: part. id$\times 1000$ + hadr. generation[93] $\times 10$ + no. of obs. level
$7 \times (n-1)+2$	px, momentum in x direction in GeV/c
$7 \times (n-1)+3$	py, momentum in y direction in GeV/c
$7 \times (n-1)+4$	pz, momentum in -z direction in GeV/c
$7 \times (n-1)+5$	x position coordinate in cm
$7 \times (n-1)+6$	y position coordinate in cm
$7 \times (n-1)+7$	t time since first interaction (or since entrance into atmosphere)[94] in nsec [for additional muon information: z coordinate in cm]
	for $n = 1 \ldots 39$ if last block is not completely filled, trailing zeros are added

Table 10: Structure of particle data sub-block.

Cherenkov photon data sub-block : (up to 39 bunches, 7 words each)	
No. of words	Contents of word (as real numbers R*4)
$7 \times (n-1)+1$	number of Cherenkov photons in bunch [in case of output on the particle output file: $99.E5 + 10\times$NINT(number of Cherenkov photons in bunch) + 1]
$7 \times (n-1)+2$	x position coordinate in cm
$7 \times (n-1)+3$	y position coordinate in cm
$7 \times (n-1)+4$	u direction cosine[95] to x-axis
$7 \times (n-1)+5$	v direction cosine[95] to y-axis
$7 \times (n-1)+6$	t time since first interaction (or since entrance into atmosphere)[94] in nsec
$7 \times (n-1)+7$	height of production of bunch in cm
	for $n = 1 \ldots 39$ if last block is not completely filled, trailing zeros are added

Table 11: Structure of Cherenkov photon data sub-block.

[93] The generation counter is set to 0 before the first interaction and augmented by each hadronic interaction or decay. The decay of $\pi^{\pm}$-mesons increases this counter by 51, thus the muons (and neutrinos) coming from $\pi^{\pm}$-decays may be discriminated from those originating in K-decays or other reactions.

[94] See keyword TSTART page 64.

[95] The direction cosines u and v are $u = sin(\theta) \cdot cos(\phi)$ and $v = sin(\theta) \cdot sin(\phi)$, see Fig. 1 page 102.

'Longitudinal' sub-block: (up to 26 depth steps/block)	
No. of word	Contents of word (as real numbers R*4)
1	'LONG'
2	event number
3	particle id (particle code or $A \times 100 + Z$ for nuclei)
4	total energy in GeV
5	(total number of longitudinal steps)$\times$ 100 + number of longitudinal blocks/shower
6	current number m of longitudinal block
7	altitude of first interaction in g/cm^2
8	zenith angle θ in radian
9	azimuth angle ϕ in radian
10	cutoff for hadron kinetic energy in GeV
11	cutoff for muon kinetic energy in GeV
12	cutoff for electron kinetic energy in GeV
13	cutoff for photon energy in GeV
$10 \times n + 4$	vertical (rsp. slant) depth of step j in g/cm^2
$10 \times n + 5$	number of γ-rays at step j
$10 \times n + 6$	number of e$^+$ particles at step j
$10 \times n + 7$	number of e$^-$ particles at step j
$10 \times n + 8$	number of μ^+ particles at step j
$10 \times n + 9$	number of μ^- particles at step j
$10 \times n + 10$	number of hadronic particles at step j
$10 \times n + 11$	number of all charged particles at step j
$10 \times n + 12$	number of nuclei[96]at step j
$10 \times n + 13$	number of Cherenkov photons at step j
	for $n = 1, 26$ and for j longitudinal steps for 1^{st} 'LONG' block: 1 ... j ... 26 for 2^{nd} 'LONG' block: 27 ... j ... 52 ..... for m^{th} 'LONG' block: $(m-1) \cdot 26 + 1$ ... j ... $m \cdot 26$ if last block is not completely filled, trailing zeros are added

Table 12: Structure of 'longitudinal' sub-block. (These type of blocks are written only if 'LONGI' is enabled and 'FLONGOUT' is disabled, page 82.)

[96]Nuclei (A $>$ 1) are not counted with the 'hadron' species. They are assumed to be completely stripped and therefore counted with their charge Z in the 'all charged' species.

Event end sub-block : (once per event)	
No. of word	Contents of word (as real numbers R*4)
1	'EVTE'
2	event number
	statistics for one shower :
3	weighted number of photons arriving at observation level(s)
4	weighted number of electrons arriving at observation level(s)
5	weighted number of hadrons arriving at observation level(s)
6	weighted number of muons arriving at observation level(s)
7	number of weighted particles written to particle output file MPATAP. (This number includes also Cherenkov bunches, if Cherenkov output is directed to MPATAP, but excludes additional muon information.)
	NKG output (if selected) :
$7+i$	$i=1,21$ lateral distribution in x direction for 1. level in cm^{-2}
$28+i$	$i=1,21$ lateral distribution in y direction for 1. level in cm^{-2}
$49+i$	$i=1,21$ lateral distribution in xy direction for 1. level in cm^{-2}
$70+i$	$i=1,21$ lateral distribution in yx direction for 1. level in cm^{-2}
$91+i$	$i=1,21$ lateral distribution in x direction for 2. level in cm^{-2}
$112+i$	$i=1,21$ lateral distribution in y direction for 2. level in cm^{-2}
$133+i$	$i=1,21$ lateral distribution in xy direction for 2. level in cm^{-2}
$154+i$	$i=1,21$ lateral distribution in yx direction for 2. level in cm^{-2}
$175+i$	$i=1,10$ electron number in steps of 100 g/cm^2
$185+i$	$i=1,10$ pseudo-age in steps of 100 g/cm^2
$195+i$	$i=1,10$ distances for electron distribution in cm
$205+i$	$i=1,10$ local pseudo-age 1. level
$215+i$	$i=1,10$ height of levels for electron numbers in g/cm^2
$225+i$	$i=1,10$ height of levels for electron numbers in cm
$235+i$	$i=1,10$ distance bins for local pseudo-age in cm
$245+i$	$i=1,10$ local pseudo-age 2. level
$255+i$	$i=1,6$ parameters of longitudinal distribution of charged particles
262	χ^2 per degree of freedom of fit to longitudinal distribution
263	weighted number of photons written to particle output file
264	weighted number of electrons written to particle output file
265	weighted number of hadrons written to particle output file
266	weighted number of muons written to particle output file
267	number of em-particles emerging from pre-shower
268..273	not used

Table 13: Structure of event end sub-block.

Run end sub-block : (once per run)	
No. of word	Contents of word (as real numbers R*4)
1	'RUNE'
2	run number
3	number of events processed
4...273	not used yet

Table 14: Structure of run end sub-block.

10.2.1 Normal Particle Output: Block Structure without Thinning

The information is stored unformatted in a fixed block structure with a block length of 22932 bytes. A block consists of 5733 words each 4 bytes long[97]. Each block consists of 21 sub-blocks of 273 words. These sub-blocks can be a RUN HEADER, EVENT HEADER, DATA BLOCK, LONGitudinal, EVENT END, or a RUN END sub-block (see Table 6). The contents of the sub-blocks are listed in Tables 7 to 14.

10.2.2 Normal Particle Output: Block Structure with Thinning

To take the weight parameter for each particle the data structure of the version without thinning has to be extended for the THIN option. The structure of the output (see Table 6 page 110) remains unchanged, but the blocksize is enlarged to a length of 26208 bytes. Now a block has 6552 words each 4 bytes long[97], consisting of 21 sub-blocks of 312 words. The ends of the sub-blocks RUN HEADER (see Table 7 page 111), EVENT HEADER (see Table 9 page 113), LONGitudinal (see Table 12 page 116), EVENT END (see Table 13 page 117), and RUN END (see Table 14) are filled up with zeros, while the data blocks (Tables 10 and 11 page 115) contain 8 words for each particle rsp. Cherenkov bunch, the last one being the weight[98]. Again 39 particles are collected within one data sub-block.

10.2.3 Normal Particle Output: Compact Option

The compact particle output available in the COMPACT option (Sect. 3.5.8 page 43) is organized similar to the normal particle output (see Sect. 10.2 page 110 ff.) with the following modifications:

- The block size of the records has variable length.
- In the block structure (Table 6 page 110) the 'event end' blocks are omitted completely.
- Only the first event header block has its full length (273 rsp. 312 4-byte words) and starts with the characters 'EVTH'. The headers of subsequent events are shortened and contain only the first 12 parameters of Table 9 (page 113) and start with the characters 'EVHW'.

[97] On UNIX installations using the GNU g77 compiler, on HP UNIX stations, and some other machines the blocks comprise two additional words giving the record length.

[98] Concerning the weight calculation see Sect. 3.5.29 page 54 ff.

- The data blocks have variable length, trailing zeros are omitted. They are headed by a 4-byte integer which defines the length (in 4 byte words) of the appended data block. The maximal length is to 1+273 4-byte words (rsp. 1+312 4-byte words for thinning).

Therefore a special reading routine is necessary for this output. The Cherenkov output is not affected by the COMPACT option, but it must be directed to the Cherenkov output file MCETAP (page 90, LCERFI = .true.).

10.3 .inclined File Output

This file is produced in the INCLINED option (Sect. 3.5.14 page 47) for inclined observation planes, using the keyword INCLIN (page 85). The format and structure is identical to the normal binary output data file as described in Sect. 10.2 (page 110), besides the fact that all coordinates are given within the inclined plane, with the origin at the intersection of the shower axis with the inclined observation plane.

10.4 .lhbook File Output

The optional file '$datnnnnnn$.lhbook' written to the directory DSN (see keyword DIRECT, page 87) contains histograms produced by the ANAHIST (page 40), AUGERHIST (page 41), and/or the MUONHIST (page 48) options. Do not use capitals in the DSN directory name as the HBOOK routines tolerate only lower case characters. Table 15 gives the numbering of the hbook histograms. The histograms with numbers $<$1000 are produced by ANAHIST, those with numbers between 8000 and 9200 by MUONHIST, while those with numbers above 100000 come from the AUGERHIST option. If more than one of those options are selected simultaneously, the series of numbers appear in parallel without interferences. All series of histograms are added up for many showers. In case of ANAHIST and AUGERHIST they are normalized correctly before writing them to output. The AUGERHIST histograms are produced for up to 20 levels (to be specified by the keyword OBSLEV, page 84) which are denoted by ii with $01 \leq \mathrm{ii} \leq 20$, and for various particle types denoted by jj with $00 \leq \mathrm{jj} \leq 04$ with the meaning: 00 = *gamma*, 01 = *electron*, 02 = *positron*, 03 = *muon*, 04 = *hadron*.

10.5 .long File Output

The optional file '$DATnnnnnn$.long' written to the directory DSN (keyword DIRECT page 87) contains a table of the longitudinal distribution of various particle numbers, arranged in the columns:
$depth$, γ, e^+, e^-, μ^+, μ^-, $hadrons$, $charged\ particles$, $nuclei$, $Cherenkov\ photons$[99,100].
In a second table the longitudinal distributions of energy deposit (in GeV) by various particle species are given in columns: $depth$, $\gamma\ energy\ cut$, $e^{\pm}\ ionization$, $e^{\pm}\ energy\ cut$,

[99] The Cherenkov photon longitudinal distributions are influenced by selecting the preprocessor options INTCLONG and NOCLONG (page 40, see also page 83) and AUGCERLONG (page 41).
[100] In case of the NEUTRINO or NUPRIM options the $neutrinos$ are given instead of the $Cherenkov\ photons$.

histo number	dimension	content of histograms
1 - 28	1-dim	longitudinal distributions
99	2-dim	particle codes vs. log10(r)
101 - 331	1-dim	ground particle distributions
341 - 451	2-dim	ground particle distributions
999	1-dim	density normalization
9111	1-dim	slant depth of muons
9112	2-dim	slant depth vs. log10(energy) of muons
9114	2-dim	slant depth vs. log10(p_t) of muons
9115	2-dim	log10(energy) vs. log10(p_t) of muons
9121	1-dim	mass overlay of muons
9122	2-dim	mass overlay vs. log10(energy) of muons
9123	2-dim	mass overlay vs. log10(p_t) of muons
9127	2-dim	mass overlay vs. log10(y-coordinate) of muons
8000-8200	2-dim	log10(energy) vs. log10(p_t)for slant depths between 0 and 2000 g/cm^2 in steps of 10 g/cm^2
10iijj	1-dim	lateral particle distributions
20ii10	1-dim	Cherenkov photons vs. emission angle
30iijj	1-dim	energy spectra
40iijj	1-dim	deposited energy vs. distance
41iijj	1-dim	releasable energy vs. distance

Table 15: Histogram numbering of the '$datnnnnnn$.lhbook' file.

$\mu^{\pm}\, ionization$, $\mu^{\pm}\, energy\, cut$, hadron ionization, $hadron\, energy\, cut$, $neutrino$[101], $sum\, of\, all$. For both tables the binning is in vertical depth (in g/cm^2) as specified by the keyword LONGI, rsp. slant depth (in g/cm^2) if the SLANT option (page 53) has been selected. This table output is activated by the keyword LONGI (page 82).

10.6 .tab File Output

The optional file '$DATnnnnnn$.tab' written to the directory DSN (keyword DIRECT page 87) contains information on the particles arriving at the lowest detector level. Activation is done by the keyword PAROUT (page 87). There are 3 tables containing separately the number of γ, $e^{\pm}$, and $\mu^{\pm}$ particles binned into energy (40 bins ranging from 100 keV to 10 TeV in logarithmic steps), time delay relative to a spherical shower front (30 bins ranging from 10 nsec to 10 μsec in logarithmic steps), and core distance (20 bins ranging from 50 m to 5 km in logarithmic steps).
Additionally a fourth table is written containing the longitudinal development of γ, e^+, e^-, μ^+, μ^-, $hadrons$, and $charged$ particles (see Sect. 10.1 page 108) in steps as defined by keyword

[101] The counting of energy going into neutrinos is described in the third item of Sect. 10.1 (page 109).

LONGI (page 82). This fourth table output is activated by the parameter LLONGI of keyword LONGI.

10.7 .dbase File and .info File Output

To build up a data base as a directory of a shower library which enables a computer aided search for specific shower events, the '$DATnnnnnn$.dbase' file (page 94) may be used. The content of the '$DATnnnnnn$.dbase' file consists of parameter words enclosed within # marks, followed by the information on the corresponding parameter. The list of parameter words and their contents is given in Table 16 (page 122). In the AUGERINFO version this file is named '$DATnnnnnn$.info' and for each parameter a new line is started omitting the # mark separators. It should be noted, that some of the parameters listed in Table 16 are present only with the selected options e.g. for thinning (page 54), for external atmospheres (page 41), for viewing cone (page 57), or for Cherenkov telescopes (page 38).

Content of $DATnnnnnn$.dbase (rsp. $DATnnnnnn$.info) file	
Keyword	Content (Format)
#version#	version number (F6.3)
#versiondate#	date of version YYYYMMDD (I9)
#modelversion#	version number of high-energy hadronic interaction program (F8.3)
#rundate#	date of run YYYYMMDD (I9)
#computer#	computer option (I2) (3=UNIX/LINUX, 4=Macintosh)
#curved#	number indicating CURVED option (I2)(2=curved, 0=else)
#neutrino#	number indicating NEUTRINO option (I2)
#cerenkov#	number indicating CERENKOV option (I2)
#runnumber#	run number (I7)
#primary#	particle code of primary particle (I5)
#e_range_l#	lower end of primary energy range (E14.7)
#e_range_u#	upper end of primary energy range (E14.7)
#slope#	slope of primary energy spectrum (E15.7)
#nkg#	number indicating use of NKG option (I2)
#egs#	number indicating use of EGS4 option (I2)
#model#	high-energy hadr. int. model (0=HDPM, 1=VENUS, 2=SIBYLL, 3=QGSJET, 4=DPMJET, 5=NEXUS/EPOS) (I2)
#gheisha#	low-energy hadr. model(1=GHEISHA, 2=UrQMD, 3=FLUKA) (I2)
#isobar#	low-energy hadr. model(1=GHEISHA, 2=UrQMD, 3=FLUKA) (I2)
#model+crossect#	high-energy hadr. model and cross-sections (0=HDPM, 11=VENUS, 22=SIBYLL, 33=QGSJET, 44=DPMJET, 55=NEXUS, 66=EPOS (I2);
#hadflag1#	number indicating NFLAIN (I2)
#hadflag2#	number indicating NFLDIF (I2)
#hadflag3#	number indicating NFLPI0 (I2)
#hadflag4#	number indicating NFLPIF (I2)
#hadflag5#	number indicating NFLCHE (I2)
#hadflag6#	number indicating NFRAGM (I2)
#longi#	number indicating use of longitudinal sampling (I2)
#longistep#	step width for longitudinal sampling (E14.7)
#magnetx#	horizontal component of Earth's magnetic field (E15.7)
#magnetz#	vertical component of Earth's magnetic field (E15.7)
#nobslev#	number of observation levels (I3)
#obslev1#	height of highest observation level (E15.7)
#obslev2#	height of 2nd observation level (E15.7)
#obslev3#	height of 3rd observation level (E15.7)
#obslev4#	height of 4th observation level (E15.7)
#obslev5#	height of 5th observation level (E15.7)
#obslev6#	height of 6th observation level (E15.7)
#obslev7#	height of 7th observation level (E15.7)
#obslev8#	height of 8th observation level (E15.7)
#obslev9#	height of 9th observation level (E15.7)
#obslev10#	height of 10th observation level (E15.7)
#hcut#	energy for hadron cut (E14.7)
#mcut#	energy for muon cut (E14.7)
#ecut#	energy for electron cut (E14.7)

Table 16: Content of $DATnnnnnn$.dbase (rsp. $DATnnnnnn$.info) file (to be continued).

Content of $DATnnnnnn$.dbase (rsp. $DATnnnnnn$.info) file (continued)	
Keyword	Content (Format)
#gcut#	energy for gamma cut (E14.7)
#theta_l#	lower end of θ range (E14.7)
#theta_u#	upper end of θ range (E14.7)
#phi_l#	lower end of ϕ range (E15.7)
#phi_u#	upper end of ϕ range (E15.7)
#fixhei#	fixed height of first interaction (E14.7)
#n1sttr#	first target (I3) (0=random air, 1=Nitrogen, 2=Oxygen, 3=Argon)
#fixchi#	starting altitude of primary particle (E14.7)
#stepfc#	multiple scattering step length factor (E14.7)
#arrang#	array rotation angle (E15.7)
#himpact1#	lower limit of horizont. shower axis (skimming incid.) (E14.7)
#himpact2#	upper limit of horizont. shower axis (skimming incid.) (E14.7)
#muaddi#	number indicating use of additional muon information (I2)
#nseq#	number of used sequences for random generator (I2)
#seq1seed1#	seed of sequence 1 (I9)
#seq1seed2#	number of calls of sequence 1 (I9)
#seq1seed3#	billions of calls of sequence 1 (I9)
#seq2seed1#	seed of sequence 2 (I9)
#seq2seed2#	number of calls of sequence 2 (I9)
#seq2seed3#	billions of calls of sequence 2 (I9)
#seq3seed1#	seed of sequence 3 (I9)
#seq3seed2#	number of calls of sequence 3 (I9)
#seq3seed3#	billions of calls of sequence 3 (I9)
#size#	size of particle tape output (I10)
#dsn_events#	data set name of particle tape output (A59)
#dsn_prtout#	data set name of .lst file output (A9)
#tape_name#	name of data tape (A10)
#backup#	name of backup tape (A10)
#howmanyshowers#	number of showers to generate (I10)
#host#	host computer name (A20)
#user#	user name (A20)
#atmosphere#	Modtran atmosphere model number (I3)
#refract#	number indicating use of refractive index (I2)
#viewcon_l#	inner limiting angle of viewing cone (E14.7)
#viewcon_u#	outer limiting angle of viewing cone (E14.7)
#telescope i #	coordinates x, y, z, r of telescope i (4F11.1)
#cscat #	number and range of scattering in x, y (2F10.1)
#thinning#	number indicating use of thinning (I2)
#thinnlev_had#	thinning level hadronic (E14.7)
#thinnlev_em#	thinning level em (E14.7)
#maxweight_had#	weight limit hadronic (E14.7)
#maxweight_em#	weight limit em (E14.7)
#rad_max#	maximum radius for radial thinning [m] (E14.7)
#energy_prim#	primary energy of first shower (E14.7)
#theta_prim#	primary's θ of first shower (E14.7)
#phi_prim#	primary's ϕ of first shower (E14.7)

Table 16: (continued) Content of $DATnnnnnn$.dbase (rsp. $DATnnnnnn$.info) file.

11 Hints for Programmers

If you need any option, addition, or other extension which is not (yet) covered by CORSIKA, feel free to program it. (Please contact T. Pierog or D. Heck before doing so.) If your addition is of general interest, it might be worth to take it over into the next official CORSIKA release. At the beginning of the $corsika$.h file you find a rather complete list of all global variables used in /COMMONS/. This enables to check the names which you give your variables for conflicts with names already used within CORSIKA. The names of the CORSIKA commons start with CR.... to prevent conflicts with common or subroutine names of linked interaction program packages.
As the largest fraction of the CORSIKA routines is written in CAPITAL letters, you are advised to use lower case characters for your private program extensions. This facilitates to distinguish your programming from the official code.

We advise you to use:
`./coconut [help] [dev] [options] [configure options]`
to handle your compilation/installation processes.
It is recommended to run *./coconut* (without arguments) to configure and install CORSIKA.
Various options are available in *./coconut* for expert use only:

- `--*`: Any additional option with trailing `--` will be passed on to `./configure` directly (try e.g. `--help`).
- `-h, --help`: Display a help file and stop.
- `-b, --batch`: No user interaction, just take DEFAULT and CACHED configuration.
- `-c, --clean`: Execute `make clean` (if CORSIKA is already installed once). Remove all object files and stop.
- `-d, --distclean`: Execute `make distclean` (if CORSIKA is already installed once). Remove all files produced by the installation (but not the binaries in `run/`) and stop.
- `-e, --expert`: Activate expert mode with additional configuration steps.
- `-i, --install`: Execute `make install` (if CORSIKA is already installed once). Compile, link and copy binaries into `run/`, but *./configure* is not called at all (except if it was never done before. If $Makefile$s are not yet installed, *./configure* will be called once).
- `-m, --make`: Execute `make all` (if CORSIKA is already installed once). Compile and link, but do not copy binaries to `run/`, and *./configure* is not called at all (no option selection ... except if it was never done before. If $Makefile$s are not yet installed, *./configure* will be called once).

- `-n, --no-cache`: Do not use cached configuration.
- `-t, --dist`: To run `make dist` (skips configuration).
- `Configure options`: Any option can be transferred to *configure*, for more information use `-h` or `--help`[102], but `--prefix`, `--bindir` and `--libdir` are already defined in *./coconut*. You should not change this options. Values defined in `$CC`, `$CCLAGS`, `$CXX`, `$CXXFLAGS`, `$CPP`, `$CPPFLAGS` and `$LDFLAGS` environment variable are always used if defined. To use the values defined in `$F77` and `$FFLAGS` environment variable in *./coconut*, the following environment variable has to be defined:

  ```
  setenv CORSIKA_USER_COMP yes
  ```

  If `$CORSIKA_USER_COMP` is not empty, then *./coconut* does not set any FORTRAN compiler name or flags. If they are not defined by you (see A.1 to know the recommended flags), *./configure* will give some default value (compilation not guaranteed). So don't use this option if you don't know exactly what you are doing (no support for this).

Run it a first time to select the options you want to use (saved in `$libdir/include-/config.h`) and then work directly on the `src`/*corsika*.F file (FORTRAN+preprocessor commands). To compile this modified source without calling *configure* again, use *./coconut* `[dev] make` for debugging. It's equivalent just to go into the proper subdirectory (depending on `dev`) and to type

```
rm -f compilefile.f
make
```

This will update the $compilefile$.f with your modifications and then compile it. When the compilation is successful, you can link objects and libraries into `run`/$corsika\langle VER\rangle\langle OS\rangle$-$\_\langle HIGH\rangle\_\langle low\rangle$[103] by typing

```
./coconut --install
```

in your `corsika-74xxx/` directory[104].
When the development phase is over, you can use the standard procedure

```
./coconut [dev]
```

[102] Call *configure* help.

[103] Where $\langle VER\rangle$ is the version number, $\langle OS\rangle$ is the operating system used for compilation, $\langle HIGH\rangle$ is the chosen high energy hadronic interaction model and $\langle low\rangle$ is the chosen low energy hadronic interaction model.

[104] The *./coconut* compiles and installs all the libraries needed by CORSIKA to be linked with and then creates a binary file in `run/`. Then, if you change $corsika$.F, doing simply *./coconut* `--make` will produce a *corsika* executable file in `src/`. You can then copy this file into `run/` to use it with all the data files.

to have binary files with different options (If `dev` is not used any more, you will have to type *./coconut* `--distclean` or `make distclean` in `corsika-74xxx/` first.).
We strongly recommend to use this scheme (instead of the traditional $compilefile$.f) because it is the easiest way for you and for us, if later this modification has to be implemented in the official release of CORSIKA (with proper preprocessor commands) or if you want to use your modifications with different CORSIKA options.
If you prefer to work on $compilefile$.f anyway, you can use the produced $Makefile$s to compile it. To produce the $Makefile$s, use *./coconut* `dev` (note the `dev` option[105]), select your options and, at the end, save the source code and do not compile. Then, you only have to rename your source file $corsika\langle VER\rangle\langle OS\rangle\_\langle HIGH\rangle\_\langle low\rangle\_compilefile$.f to $compilefile$.f and then you have to type *make* in `src/` to compile or *make* `install` in your `corsika-74xxx/` directory to (compile and) link to get a binary file. In this case, you should not use *./coconut* any more, because this will not take into account your modifications (and even overwrite the $compilefile$.f). Of course, if you already used some "home-made" $Makefile$s with a former version of CORSIKA, you can use it with this $compilefile$.f.

12 In Case of Problems . . .

In spite of our care to avoid faults in the physics model and errors in the programming of CORSIKA and in writing this CORSIKA GUIDE, you may have problems of technical or physical nature with the code or the results you obtain from it. Please report all problems to the address mentioned on page 2. When applying for help in cases of crash, **it is recommended to transmit the control printout file (.lst file)**, as it contains the selected preprocessor options as well as it echos the employed input keywords with their parameters.
Special interest exists in comparisons of CORSIKA simulations with experimental cosmic ray data.
Suggestions to improve CORSIKA in any respect are welcome.

[105] If you don't use the `dev` option when running *./coconut*, you can follow the same following scheme but using $Makefile$s located in `corsika-74xxx/lib/'uname'/` and `corsika-74xxx/lib/'uname'/src/` (for instance `corsika-74xxx/lib/Linux/src/` if you work on a LINUX system) instead of just `corsika-74xxx/`.

13 Acknowledgments

The authors thank to all CORSIKA users who have helped to eliminate errors and to improve the program. Special thanks go to the authors of the various hadronic interaction models for their help to get their programs running and for their advice in coupling the programs with CORSIKA. We are indebted to K. Bernlöhr (Heidelberg) for making available and updating his package of **C**-routines for Cherenkov telescopes and external atmospheres and for supplying the VIEWCONE option, and to D. Chirkin (Berkeley) for contributing the VOLUMECORR option. We thank F. Schröder (Wuppertal) for his engagement in developing and testing the CURVED version. The authors are obliged to J. Wentz (Karlsruhe) for his pioneering work in coupling the UrQMD model with CORSIKA and for contributing the COMPACT option. We thank the Auger group Naples (C. Aramo, G. Miele, O. Pisanti, L. Rosa) for developing the NUPRIM option. Thanks go to P. Homola (Krakow) who has written the **C**-routines to treat the pre-showering of ultra-high energetic photons in the Earth's magnetic field. We thank A. Chou (Fermilab) to bring our attention to GHEISHA correction patches, and R. E. Cassell and G. Bower (SLAC) to make them available for CORSIKA. The programming of the AUGERINFO 'info file', the support by developing the AUGERHIST extensions, and the detection of several severe bugs by M. Risse (Karlsruhe, now at Siegen) is acknowledged. We are indebted to F. Schmidt (Leeds, now at Chicago) for the development of the PLOTSH2 option including the visualization program *map2png*, R. Ulrich (KIT, Karlsruhe) for programming the ROOTOUT and all COAST options and the initial development together with S. Müller of the coconut program, and T. Hüge (KIT, Karlsruhe) for programming the CoREAS code. We thank R.P. Kokoulin and A. Bogdanov (MEPHI, Moscow) for the more precise treatment of muonic interactions and energy loss. The introduction of the TAULEP option by P. Abreu (LIP Lisboa) is acknowledged. The support by H. Singla, S. Kalra, and S. Sharma under the guidance of G. Poghosyan (Steinbuch Computing Center, KIT) in developing the parallel versions is acknowledged. We are obliged to M. Doert (TU Dortmund) for the TRAJECT option. Thanks go to A. Gascón (Granada, Spain) for programming of the interactions of charmed and bottom particles using the PYTHIA package and R. Conceição and Eva Santos (LIP, Portugal) for the MUONHIST option.

A Compilation and Linking

Using the *./coconut* script, it will call $make$ automatically and nothing has to be done. But for expert people who want to do their own $Makefile$, a description of what has to be done is given in this appendix.

A.1 Compilation

After running `./coconut` asking not to compile at the end of the selection process, you get a source file $corsika\langle VER\rangle\langle OS\rangle\_\langle HIGH\rangle\_\langle low\rangle\_compilefile$.f in the `src/` subdirectory. This subdirectory is the working directory for the following commands. This *corsika_compilefile*.f FORTRAN file and the $dpmjet25$i.f, ***epos***-xxx.f, +***nexus***-xxx.f, +$qgsjet$-II-03.f (rsp. $qgsjet01d$.f), $sibyll2.1$.f, $venus$.f, and $gheisha\_2002d$.f sources and/or the ***urqmd*** FORTRAN files have to be compiled before linking them together to obtain an executable module. To prevent overwriting of local variables by optimization of some compilers SAVE statements (which replace the '- static' option to be used for the previous release) have been implemented throughout in the FORTRAN files of ***corsika_compilefile***.f, $gheisha\_2002d$.f, $dpmjet25$i.f, $qgsjet$-II-04.f (rsp. $qgsjet01d$.f), $sibyll2.1$.f, and $venus$.f. It is recommended to use the 'bounds check' [106] option for first trials to inhibit uncontrolled array operations outside the allowed index range.

- For DEC-UNIX machines the procedure looks like
  `f77 -c -check_bounds $1.f 2>$1.err`
  (This means: Suppress the loading phase; generate code to perform runtime checks on subscript.)

- Procedures for LINUX computers with GNU g77 compilers should be used without optimization[107] and are
  `g77 -c -O0 $1.f 2>$1.err`
  to ensure correct simulations.

- Procedures for LINUX hosts (with 64bit AMD CPU's) with GNU g77 compilers should be used without optimization[107] and are
  `g77 -c -O0 -m32 $1.f 2>$1.err`
  to ensure correct simulations with the correct data format of the binary output files.

- Hosts with Portland pgf77 compiler available might use
  `pgf77 -c -O2 $1.f 2>$1.err`
  which gives a fast and reliable executable on LINUX hosts.

[106] But not using GFORTRAN compiler, it will result in unexpected stop.

[107] Do not use the optimization without carefully checking the results. There is bad experience with GNU g77 (v0.5.24 and egcs-2.91.66) which frequently brings NaN in the particle output file for the x and y coordinates of particles or results in unidentified hang-ups within the QGSJET routines. Also the g77 optimization causes the DADMUL integration routine to end with an error stop (message: DBRSGM: IFAIL= 1 ....) despite the correct programming respecting all FORTRAN standards.

- Compile procedures for IBM RS6000 are
  `xlf -c -C -O -qextname -qsave -qmaxmem=-1 $1.f 2>$1.err`
- For HP-UX processors the compilation procedures look like
  `fort77 -c -K +E1 +ppu +Dportable -O2 +Onolimit $1.f 2>$1.err`
  and the `+E1` option should also be used in the link step.

These compiler procedures should be used for all needed FORTRAN programs. Our experience with machines other than DEC or LINUX are sparse.

DPMJET option: To compile the $dpmjet25$i.f files (i=3bc, 3c, 4, 5c, 6c) in `../dpmjet/`, a procedure equivalent to the one for f77 compilers on DEC-UNIX machines[108] is recommended:

```
f77 -c -C -fpe4 -check underflow -check overflow -g3 -O1
```

(This means: Suppress the loading phase; generate code to perform runtime checks on subscript; continue program after overflow, zero divide, invalid data, or underflow; check underflow and overflow at run time; produce trace back and debugging information in object file; enable local optimizations.)
As the file $dpmjet253$c.f (PYTHIA package) is extremely large you presumably have to give in advance[109]

```
limit datasize unlimited
limit stacksize unlimited
```

to overcome the small default values of many compilers which lead to an error stop during compilation. For machines other than DEC-UNIX and LINUX there is no experience with the compilation of DPMJET routines.

EPOS option: The compiler procedures of the standard case (see begin of this subsection A.1) should be used.
To compile the *epos*`*.f` source files in `../epos/` you should use the $Makefile$.ka available with EPOS (omitting the link step).
For compiling and linking EPOS you presumably have to give[110]

```
limit datasize unlimited
limit stacksize unlimited
```

to overcome the small default values of many compilers which lead to an error stop during linking. Linking is performed e.g. by

[108] For GNU g77 compilers the options
`-c -C -fno-automatic -finit-local-zero -Wunused -m32 -Wuninitialized -malign-double -O`
should be used.
[109] Usage of csh shell is assumed. In other shells (sh or ksh) you should use `ulimit -d unlimited` and `ulimit -s unlimited` .
[110] See footnote at DPMJET compilation page 129.

```
f77 corsika_compilefile.o gheisha_2002d.o ../epos/*.o -o corsika\vers
```

where the compiled EPOS program parts are expected to be available within the subdirectory `../epos/` .

NEXUS option: The compiler procedures of the standard case (see begin of this subsection A.1) should be used.
To compile the *nexus*-`xxx.f` source files in `../nexus/` you should use the $Makefile$.ka available with NEXUS (omitting the link step).
For compiling and linking NEXUS you presumably have to give[111]

```
limit datasize unlimited
limit stacksize unlimited
```

to overcome the small default values of many compilers which lead to an error stop during linking. Linking is performed e.g. by

```
f77 corsika_compilefile.o gheisha_2002d.o ../nexus/*.o -o corsika\vers
```

where the compiled NEXUS program parts are expected to be available within the subdirectory `../nexus/` .

FLUKA option: For compiling the CORSIKA-FLUKA version of the *corsika_compilefile*.f, the FLUKA include files should be available in the subdirectory from which you are calling the compiler to include them into the CORSIKA-FLUKA linking routines at the appropriate places. Preferentially you use the f77 rsp. g77 compiler with the option

`-I`*flukadirectory*`/flukapro`

to indicate the compiler where to find the include files. For all steps using the FLUKA package you presumably have to give[111]

```
limit datasize unlimited
limit stacksize unlimited
```

to overcome the small default values of many compilers which lead to an error stop during linking.

URQMD option: For compiling the CORSIKA-URQMD version of the *corsika_compilefile*.f, the UrQMD include files $boxinc$.f, $colltab$.f, $comres$.f, $coms$.f, $inputs$.f, $newpart$.f, and $options$.f should be available in the directory from which you are calling the compiler to include them into the CORSIKA-URQMD linking routines at the appropriate places. Alternatively you may use the f77 rsp. g77 compiler with the option

`-I../urqmd`

to indicate the compiler where to find the include files. The compiler procedures of the standard case (see begin of this subsection A.1) should be used to compile the *corsika_compilefile*.f .
The *urqmd1.3_cors*.tar.gz file contains the UrQMD1.3_cors source routines with slight modifications to adapt them for the use with CORSIKA. To compile these UrQMD source files one

[111] See footnote at DPMJET compilation.

uses the `(g)make` command with the option *-f GNUmakefile_corsika* (omitting the link step) in the `../urqmd/` subdirectory.

NUPRIM option: When compiling the CORSIKA-NUPRIM version of the *corsika_compilefile*.f, the HERWIG_C.INC include file should be available in the subdirectory from which you are calling the compiler to include it into the CORSIKA-HERWIG linking routines at the appropriate places. Alternatively you may use the f77 rsp. g77 compiler with the option

```
-I../herwig
```

to indicate the compiler where to find the include file.
This include file is a simple copy of the HERWIGnn.INC file provided with the source code of the HERWIG model. In the `herwig/` subdirectory you give the command:

```
cp HERWIGnn.INC HERWIG_C.INC
```

Before compiling the HERWIG routines you should

- remove (erase or comment off) the function *HWRGEN* with entries *HWRSET* and *HWRGET* thus using the CORSIKA random generator (sequence 5) within the HERWIG routines;
- when using the NUPRIM option together with NEXUS or EPOS: Rename the `COMMON /JET/` within Herwig.

The compiler procedures of the standard case (see begin of this subsection A.1) should be used for the CORSIKA-NUPRIM version as well as for HERWIG.

C-file: If you have specified the TIMERC or the PRESHOWER option you need the compilation of the **C**-file $timerc$.c or $preshw$.c with $utils$.c and $veto$.c using a command like

`cc -c timerc.c` (rsp. `cc -c preshw.c`).

For the compilation of the **C**-routines of the bernlohr package see the instructions delivered with the bernlohr package.

A.2 Linking

For linking of most CORSIKA versions typically a procedure is used (assuming QGSJET is employed) like:

```
g77 corsika_compilefile.o gheisha_2002d.o qgsjet-II-04.o trapfpe.o  \
   -m32 -o corsika
```

For linking the large packages of FLUKA, UrQMD, DPMJET, EPOS, HERWIG, NEXUS, and VENUS you presumably have to give[112]

```
limit datasize unlimited
```

[112] See footnote at DPMJET compilation page 129.

`limit stacksize unlimited`
to overcome the small default values of many compilers which lead to an error stop during linking.
If the `pgf77` compiler has been used for compilation, you should also use pgf77 for linking:

```
pgf77 corsika_compilefile.o gheisha_2002d.o qgsjet-II-04.o -o corsika
```

For linking **FLUKA** versions the FLUKA library has to be included in the link step:

```
g77 corsika_compilefile.o qgsjet-II-04.o trapfpe.o \
   -Lflukadirectory -lflukahp -m32 -o corsika
```

For linking FLUKA with `pgf77` you should use the option `-g77libs` in the link step to ensure that the runtime library[113] of `g77` is available as it is needed for running the FLUKA routines.
In the run step of FLUKA versions the environment variable FLUPRO has to be set for csh shell

`setenv FLUPRO` *flukadirectory*

or for bash/sh shells

`export FLUPRO=`*flukadirectory*

which is necessary to link the data files which will be read in by the FLUKA package. Additionally the stacksize and datasize limits have to be overcome also in the run step.

Linking of **URQMD** versions is performed easiest (e.g. for DEC-UNIX) by

```
f77 corsika_compilefile.o qgsjet-II-04.o ../urqmd/obj_urqcors/*.o \
   trapfpe.o -m32 -o corsika74xxx_QGS-II_urq13
```

to include the `.o` files of UrQMD which are stored into the subdirectory `../urqmd/obj_urqcors` by the GNU-make compilation procedure.

No other libraries are normally required.

If you have specified the TIMERC or the PRESHOWER option, you need the linking of the compiled **C**-file $timerc$.o (rsp. $preshw$.o with $utils$.o and $veto$.o).
For linking the compiled **C**-routines of the bernlohr package see the instructions delivered with the bernlohr package.

[113] Be aware that the FLUKA LINUX version and the runtime library are fitting together.

B Flow Diagram

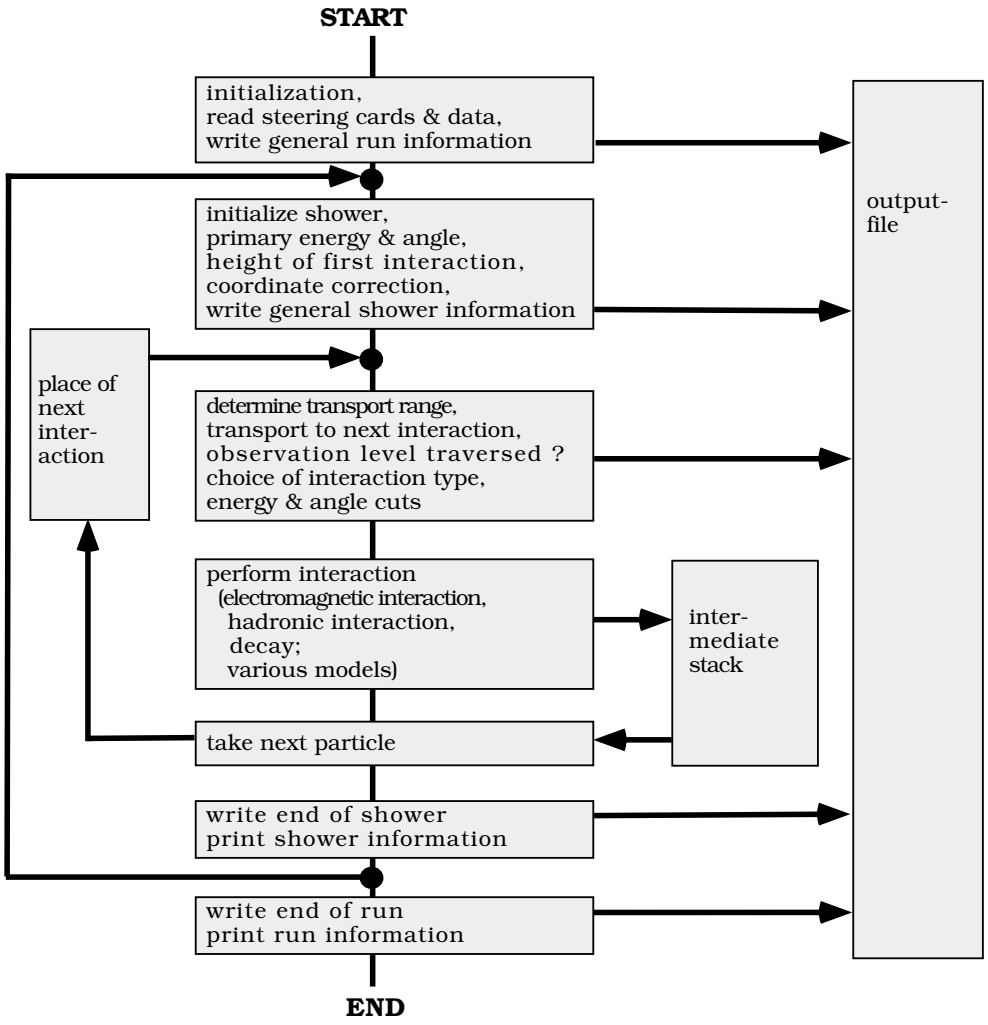

Figure 2: Simplified flow diagram of CORSIKA.

C Sequence of Initializations

The sequence of initializations is shown for the QGSJET option (INTTEST option in brackets) as example, other interaction model options may differ slightly. The indentation gives the hierarchy of subroutines. Subroutine names are written in CAPITALS.

```
AAMAIN (CORSIKA main program)
     START
          write CORSIKA version and options
          PAMAF
              set particle masses and decay times
          DATAC
              read keywords from input
          set OBSLEV, atmospheres and layers
          initialize random generator
          read NUCNUCCS cross-section tables
          FILOPN
              open output files and external stack
          QGSINI
              initialize QGSJET-II
              QGSET (QGSJET-II)
              QGAINI (QGSJET-II)
                  read qgsdat-II-04 and sectnu-II-04
              QGSSIGINI
          CGHINI
              initialize GHEISHA
          EGSIN1
              initialize EGS4
          EGSIN2
              read EGSDAT6_x.x
          calculate physical constants
          (set projectile and target for HSINI)
          (HSINI)
              (histogram initialization for INTTEST)
     INPRM
          check input parameters
          set various parameters:  magn. field, Cherenkov...
          MUPINI
              set μ-parameters and μ-cross-section tables
          write RUNHEADER
          write .dbase (rsp.  .info) file
     ININKG
          initialize NKG parameters
     clear statistics arrays:  multiplicity, elasticity, weight...
     shower loop
     .
     .
```

D Atmospheres

The atmosphere adopted consists of N_2, O_2, and Ar with the volume fractions of 78.1%, 21.0%, and 0.9% [73]. The density variation of the atmosphere with altitude is modeled by 5 layers. In the lower four of them the density follows an exponential dependence on the altitude leading to a relation between the mass overburden $T(h)$ of the atmosphere and the height h of the form

$$T(h) = a_i + b_i \cdot e^{-h/c_i} \quad i = 1, \ldots, 4 \quad . \tag{1}$$

In the fifth layer the mass overburden decreases linearly with height

$$T(h) = a_5 - b_5 \cdot h/c_5 \quad .$$

The boundary of the atmosphere in this model is defined at the height where the mass overburden $T(h)$ vanishes (which is at $h = 112.8$ km for the U.S. standard atmosphere).
Various atmospheres are foreseen: U.S. standard atmosphere parameterized according to J. Linsley [74], 7 typical atmospheres as measured above Stuttgart (about 60 km away from Karlsruhe) at various days of 1993 and transmitted by Deutscher Wetterdienst Offenbach (parameterized according to Ref. [75]), 4 South pole atmospheres (parameterized by D. Chirkin according to the MSIS-90-E model [76]), two South pole atmospheres by P. Lipari [77], and 12 monthly dependent atmospheres for the Pierre Auger Observatory experiment at Malargüe (Argentina) parameterized with GDAS-data by M. Will and B. Keilhauer [78, 79]. B. Keilhauer provided also a new parameterization of the U.S. standard atmosphere. The parameters a_i, b_i, and c_i are selected in a manner that the function $T(h)$ is continuous at the layer boundaries and can be differentiated continuously. In Tables 17 - 43 the parameters for the various models are listed. Additional atmospheres [40] are available by the keyword ATMOSPHERE (page 68) in the ATMEXT option (page 41). User specific atmosphere parameters may be read in using the keywords ATMOD, ATMA, ATMB, ATMC, and ATMLAY.

U.S. Standard Atmosphere

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \ldots 4$	-186.555305	1222.6562	994186.38
2	$4 \ldots 10$	-94.919	1144.9069	878153.55
3	$10 \ldots 40$	0.61289	1305.5948	636143.04
4	$40 \ldots 100$	0.0	540.1778	772170.16
5	> 100	0.01128292	1	10^9

Table 17: Parameters of the U.S. standard atmosphere (after Linsley).

Middle Europe

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-118.1277	1173.9861	$919546.$
2	$4 \dots 10$	-154.258	1205.7625	963267.92
3	$10 \dots 40$	0.4191499	1386.7807	$614315.$
4	$40 \dots 100$	$5.4094056 \cdot 10^{-4}$	555.8935	739059.6
5	> 100	0.01128292	1	10^9

Table 18: Parameters of the AT115 atmosphere (January 15, 1993).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-195.837264	1240.48	$933697.$
2	$4 \dots 10$	-50.4128778	1117.85	$765229.$
3	$10 \dots 40$	0.345594007	1210.9	$636790.$
4	$40 \dots 100$	$5.46207 \cdot 10^{-4}$	608.2128	733793.8
5	> 100	0.01128292	1	10^9

Table 19: Parameters of the AT223 atmosphere (February 23, 1993).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-253.95047	1285.2782	$1088310.$
2	$4 \dots 10$	-128.97714	1173.1616	$935485.$
3	$10 \dots 40$	0.353207	1320.4561	$635137.$
4	$40 \dots 100$	$5.526876 \cdot 10^{-4}$	680.6803	727312.6
5	> 100	0.01128292	1	10^9

Table 20: Parameters of the AT511 atmosphere (May 11, 1993).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-208.12899	1251.474	$1032310.$
2	$4 \dots 10$	-120.26179	1173.321	$925528.$
3	$10 \dots 40$	0.31167036	1307.826	$645330.$
4	$40 \dots 100$	$5.591489 \cdot 10^{-4}$	763.1139	720851.4
5	> 100	0.01128292	1	10^9

Table 21: Parameters of the AT616 atmosphere (June 16, 1993).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-77.875723	1103.3362	$932077.$
2	$4 \dots 10$	-214.96818	1226.5761	$1109960.$
3	$10 \dots 40$	0.3721868	1382.6933	$630217.$
4	$40 \dots 100$	$5.5309816 \cdot 10^{-4}$	685.6073	726901.3
5	> 100	0.01128292	1	10^9

Table 22: Parameters of the AT822 atmosphere (August 22, 1993).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-242.56651	1262.7013	$1059360.$
2	$4 \dots 10$	-103.21398	1139.0249	$888814.$
3	$10 \dots 40$	0.3349752	1270.2886	$639902.$
4	$40 \dots 100$	$5.527485 \cdot 10^{-4}$	681.4061	727251.8
5	> 100	0.01128292	1	10^9

Table 23: Parameters of the AT1014 atmosphere (October 14, 1993).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \dots 4$	-195.34842	1210.4	$970276.$
2	$4 \dots 10$	-71.997323	1103.8629	$820946.$
3	$10 \dots 40$	0.3378142	1215.3545	$639074.$
4	$40 \dots 100$	$5.48224 \cdot 10^{-4}$	629.7611	731776.5
5	> 100	0.01128292	1	10^9

Table 24: Parameters of the AT1224 atmosphere (December 24, 1993).

South pole

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \ldots 4$	-137.656	1130.74	867358.
2	$4 \ldots 10$	-37.9610	1052.05	741208.
3	$10 \ldots 40$	0.222659	1137.21	633846.
4	$40 \ldots 100$	-0.000616201	442.512	759850.
5	> 100	0.00207722	1	$5.4303203 \cdot 10^9$

Table 25: Parameters of South pole atmosphere for March 31, 1997 (MSIS-90-E).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \ldots 4$	-163.331	1183.70	875221.
2	$4 \ldots 10$	-65.3713	1108.06	753213.
3	$10 \ldots 40$	0.402903	1424.02	545846.
4	$40 \ldots 100$	-0.000479198	207.595	793043.
5	> 100	0.00188667	1	$5.9787908 \cdot 10^9$

Table 26: Parameters of South pole atmosphere for Jul. 01, 1997 (MSIS-90-E).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \ldots 4$	-142.801	1177.19	861745.
2	$4 \ldots 10$	-70.1538	1125.11	765925.
3	$10 \ldots 40$	1.14855	1304.77	581351.
4	$40 \ldots 100$	-0.000910269	433.823	775155.
5	> 100	0.00152236	1	$7.4095699 \cdot 10^9$

Table 27: Parameters of South pole atmosphere for Oct. 01, 1997 (MSIS-90-E).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0 \ldots 4$	-128.601	1139.99	861913.
2	$4 \ldots 10$	-39.5548	1073.82	744955.
3	$10 \ldots 40$	1.13088	1052.96	675928.
4	$40 \ldots 100$	-0.00264960	492.503	829627.
5	> 100	0.00192534	1	$5.8587010 \cdot 10^9$

Table 28: Parameters of South pole atmosphere for Dec. 31, 1997 (MSIS-90-E).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 2.67$	-113.139	1133.10	861730.
2	$2.67 \ldots 5.33$	-79.0635	1101.20	826340.
3	$5.33 \ldots 8.0$	-54.3888	1085.00	790950.
4	$8.0 \ldots 100.0$	0.0000	1098.00	682800.
5	> 100.0	0.00421033	1	$2.6798156 \cdot 10^9$

Table 29: Parameters of South pole atmosphere for January (Lipari).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 6.67$	-59.0293	1079.00	764170.
2	$6.67 \ldots 13.33$	-21.5794	1071.90	699910.
3	$13.33 \ldots 20.0$	-7.14839	1182.00	635650.
4	$20.0 \ldots 100.0$	0.0000	1647.10	551010.
5	> 100.0	0.000190175	1	$59.329575 \cdot 10^9$

Table 30: Parameters of South pole atmosphere for August (Lipari).

Keilhauer's U.S. standard atmosphere

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 7.0$	-149.801663	1183.6071	954248.34
2	$7.0 \ldots 11.4$	-57.932486	1143.0425	800005.34
3	$11.4 \ldots 37.0$	0.63631894	1322.9748	629568.93
4	$37.0 \ldots 100.0$	$4.35453690 \cdot 10^{-4}$	655.67307	737521.77
5	> 100.0	0.01128292	$1.$	10^9

Table 31: Parameters of the U.S. standard atmosphere (after Keilhauer).

Malargüe (Argentina)

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.4$	-136.72575606	1174.8298334	982815.95248
2	$9.4 \ldots 15.3$	-31.636643044	1204.8233453	754029.87759
3	$15.3 \ldots 31.6$	1.8890234035	1637.7703583	594416.83822
4	$31.6 \ldots 100.0$	$3.9201867984 \cdot 10^{-4}$	735.96095023	733974.36972
5	> 100.0	0.01128292	$1.$	10^9

Table 32: Parameters of January (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.2$	-137.25655862	1176.0907565	981369.6125
2	$9.2 \ldots 15.4$	-31.793978896	1197.8951104	756657.65383
3	$15.4 \ldots 31.0$	2.0616227547	1646.4616955	592969.89671
4	$31.0 \ldots 100.0$	$4.1243062289 \cdot 10^{-4}$	755.18728657	731345.88332
5	> 100.0	0.01128292	$1.$	10^9

Table 33: Parameters of February (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.6$	-132.36885162	1172.6227784	972654.0563
2	$9.6 \ldots 15.2$	-29.077046629	1215.3964677	742769.2171
3	$15.2 \ldots 30.7$	2.090501509	1617.0099282	595342.19851
4	$30.7 \ldots 100.0$	$4.3534337925 \cdot 10^{-4}$	769.51991638	728921.61954
5	> 100.0	0.01128292	1.	10^9

Table 34: Parameters of March (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 10.0$	-129.9930412	1172.3291878	962396.5521
2	$10.0 \ldots 14.9$	-21.847248438	1250.2922774	711452.06673
3	$14.9 \ldots 32.6$	1.5211136484	1542.6248413	603480.61835
4	$32.6 \ldots 100.0$	$3.9559055121 \cdot 10^{-4}$	713.1008285	735460.83741
5	> 100.0	0.01128292	1.	10^9

Table 35: Parameters of April (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 10.2$	-125.11468467	1169.9511302	947742.88769
2	$10.2 \ldots 15.1$	-14.591235621	1277.6768488	685089.57509
3	$15.1 \ldots 35.9$	0.93641128677	1493.5303781	609640.01932
4	$35.9 \ldots 100.0$	$3.2475590985 \cdot 10^{-4}$	617.9660747	747555.95526
5	> 100.0	0.01128292	1.	10^9

Table 36: Parameters of May (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 10.1$	-126.17178851	1171.0916276	940102.98842
2	$10.1 \ldots 16.0$	-7.7289852811	1295.3516434	661697.57543
3	$16.0 \ldots 36.7$	0.81676828638	1455.3009344	612702.0632
4	$36.7 \ldots 100.0$	$3.1947676891 \cdot 10^{-4}$	595.11713507	749976.26832
5	> 100.0	0.01128292	1.	10^9

Table 37: Parameters of June (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.6$	-126.17216789	1172.7340688	934649.58886
2	$9.6 \ldots 16.5$	-8.6182537514	1258.9180079	672975.82513
3	$16.5 \ldots 37.4$	0.74177836911	1450.0537141	614888.52458
4	$37.4 \ldots 100.0$	$2.9350702097 \cdot 10^{-4}$	583.07727715	752631.28536
5	> 100.0	0.01128292	1.	10^9

Table 38: Parameters of July (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.6$	-123.27936204	1169.763036	931569.97625
2	$9.6 \ldots 15.9$	-10.051493041	1251.0219808	678861.75136
3	$15.9 \ldots 36.3$	0.84187346153	1436.6499372	617363.34491
4	$36.3 \ldots 100.0$	$3.2422546759 \cdot 10^{-4}$	627.42169844	746739.16141
5	> 100.0	0.01128292	1.	10^9

Table 39: Parameters of August (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.5$	-126.94494665	1174.8676453	936953.91919
2	$9.5 \ldots 16.2$	-9.5556536981	1251.5588529	678906.60516
3	$16.2 \ldots 37.2$	0.74939405052	1440.8257549	618132.60561
4	$37.2 \ldots 100.0$	$2.9823116961 \cdot 10^{-4}$	606.31473165	750154.67709
5	> 100.0	0.01128292	1.	10^9

Table 40: Parameters of September (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.5$	-133.13151125	1176.9833473	954151.404
2	$9.5 \ldots 15.5$	-13.973209265	1244.234531	692708.89816
3	$15.5 \ldots 36.5$	0.8378263431	1464.0120855	615439.43936
4	$36.5 \ldots 100.0$	$3.111742176 \cdot 10^{-4}$	622.11207419	747969.08133
5	> 100.0	0.01128292	1.	10^9

Table 41: Parameters of October (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.6$	-134.72208165	1175.7737972	964877.07766
2	$9.6 \ldots 15.3$	-18.172382908	1238.9538504	706199.57502
3	$15.3 \ldots 34.6$	1.1159806845	1505.1614366	610242.24564
4	$34.6 \ldots 100.0$	$3.5217025515 \cdot 10^{-4}$	670.64752105	741412.74548
5	> 100.0	0.01128292	1.	10^9

Table 42: Parameters of November (GDAS).

Layer i	Altitude h (km)	a_i (g/cm^2)	b_i (g/cm^2)	c_i (cm)
1	$0. \ldots 9.6$	-135.40825209	1174.644971	973884.44361
2	$9.6 \ldots 15.6$	-22.830409026	1227.2753683	723759.74682
3	$15.6 \ldots 33.3$	1.4223453493	1585.7130562	600308.13983
4	$33.3 \ldots 100.0$	$3.7512921774 \cdot 10^{-4}$	691.23389637	738390.20525
5	> 100.0	0.01128292	1.	10^9

Table 43: Parameters of December (GDAS).

References

[1] T. Antoni et al. (KASCADE Collaboration), *Nucl. Instr. Meth.* **A 513** (2003) 490

[2] K.-H. Kampert et al. (KASCADE-Grande Collaboration), *Nucl. Phys. B (Proc. Suppl.)* **122** (2003) 422; G. Navarra et al. (KASCADE-Grande Collaboration), *Nucl. Instr. Meth.* **A 518** (2004) 207

[3] D. Heck et al., Report **FZKA 6019** (1998), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/corsika/physics_description/corsika_phys.html

[4] J. Ranft, *Phys. Rev.* **D51** (1995) 64; preprint arXiv: *hep-ph/9911213* and *hep-ph/9911232* (1999)

[5] J.N. Capdevielle et al., Report **KfK 4998** (1992), Kernforschungszentrum Karlsruhe

[6] N.N. Kalmykov and S.S. Ostapchenko, *Yad. Fiz.* **56** (1993) 105; *Phys. At. Nucl.* **56** N3 (1993) 346; N.N. Kalmykov, S.S. Ostapchenko, and A.I. Pavlov, *Izv. RAN Ser. Fiz.* **58** (1994) N12 p.21; N.N. Kalmykov, S.S. Ostapchenko, and A.I. Pavlov, *Bull. Russ. Acad. Science (Physics)* **58** (1994) 1966; N.N. Kalmykov, S.S. Ostapchenko, and A.I. Pavlov, *Nucl. Phys. B (Proc. Suppl.)* **52B** (1997) 17; S.S. Ostapchenko, private communications (2001)

[7] R.S. Fletcher, T.K. Gaisser, P. Lipari, and T. Stanev, *Phys. Rev.* **D50** (1994) 5710; J. Engel, T.K. Gaisser, P. Lipari, and T. Stanev, *Phys. Rev.* **D46** (1992) 5013

[8] R. Engel, T.K. Gaisser, P. Lipari, and T. Stanev, *Proc.* 26^{th} *Int. Cosmic Ray Conf.*, Salt Lake City (USA), **1** (1999) 415; E.-J. Ahn, R. Engel, T.K. Gaisser, P. Lipari, and T. Stanev, *Phys. Rev.* **D80** (2009) 094003; R. Engel, private communications

[9] K. Werner, *Phys. Rep.* **232** (1993) 87

[10] H.J. Drescher et al., *Phys. Rep.* **350** (2001) 93

[11] T. Pierog et al., arXiv:1306.0121[hep-ph] (2013)

[12] S.S. Ostapchenko, *Phys. Rev.* **D83** (2011) 014018

[13] A. Ferrari, P.R. Sala, A. Fassò, and J. Ranft, Report **CERN-2005-10** (2005); **INFN/TC_05/11**; **SLAC-R-773** (2005); A. Fassò, A. Ferrari, S. Roesler, P.R. Sala, G. Battistoni, F. Cerutti, E. Gadioli, M.V. Garzelli, F. Ballarini, O. Ottolenghi, A. Empl, and J. Ranft, Computing in High Energy and Nuclear Physics 2003 Conference (CHEP2003), La Jolla, CA (USA), March 24-28, 2003 (paper MOMT005); eConf C0303241 (2003); arXiv: *hep-ph/0306267* (2003); http://www.fluka.org/

[14] H. Fesefeldt, Report **PITHA-85/02** (1985), RWTH Aachen, available from: http://cds.cern.ch/record/162911/files/CM-P00055931.pdf

[15] S.A. Bass et al., *Prog. Part. Nucl. Phys.* **41** (1998) 225; M. Bleicher et al., *J. Phys. G: Nucl. Part. Phys.* **25** (1999) 1859; http://www.th.physik.uni-frankfurt.de/˜urqmd/

[16] W.R. Nelson, H. Hirayama, and D.W.O. Rogers, Report **SLAC 265** (1985), Stanford Linear Accelerator Center; http://www.slac.stanford.edu/pubs/slacreports/slac-r-265.html; http://www.irs.inms.nrc.ca/inms/irs/EGS4/get_egs4.html

[17] A.A. Lagutin, A.V. Plyasheshnikov, and V.V. Uchaikin, *Proc.* 16^{th} *Int. Cosmic Ray Conf.*, Kyoto (Japan), **7** (1979) 18; J.N. Capdevielle for KASCADE Collaboration, *Proc.* 22^{nd} *Int. Cosmic Ray Conf.*, Dublin (Ireland), **4** (1991) 405

[18] G. Corcella et al., *JHEP* **0101** (2001) 010; G. Marchesini et al., *Comp. Phys. Comm.* **67** (1992) 465; http://www.hep.phy.cam.ac.uk/theory/webber/Herwig/

[19] M. Ambrosio et al., Auger technical note **GAP 2003-013** (2003); *Proc.* 28^{th} *Int. Cosmic Ray Conf.*, Tsukuba (Japan), (2003) 595; http://www1.na.infn.it/wsubnucl /cosm/auger/activities.html

[20] J. Abraham et al., (Auger Coll.), *Nucl. Instr. Meth.* **A 523** (2004) 50; *Nucl. Instr. Meth¿* **A 620** (2010) 227

[21] T. Pierog et al., *Nucl. Phys. B (Proc. Suppl.)* **151** (2006) 159

[22] T. Bergmann, R. Engel, D. Heck, N.N. Kalmykov, S. Ostapchenko, T. Pierog, T. Thouw, and K. Werner, *Astropart. Phys.* **26** (2007) 420

[23] D. Heck et al., Proc. of Workshop *Computational Methods in Science and Engineering* (SimLabs@KIT 2010), Karlsruhe (Germany), (2010) 87; http://digbib.ubka.uni-karlsruhe.de/volltexte/documents/1794913

[24] K. Werner, F. M. Liu, and T. Pierog, *Phys. Rev.* **C74** (2006) 044902

[25] J. Knapp, D. Heck, and G. Schatz, Report **FZKA 5828** (1996), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/˜heck/publications/; J. Knapp, *Nucl. Phys. B (Proc. Suppl.)* **75A** (1999) 89; D. Heck et al. (KASCADE Collaboration), *Proc.* 27^{th} *Int. Cosmic Ray Conf.*, Hamburg (Germany), (2001) 233; J. Knapp et al., *Astropart. Phys.* **19** (2003) 77; D. Heck, M. Risse, and J. Knapp, *Nucl. Phys. B (Proc. Suppl.)* **122** (2003) 40

[26] D. Heck, *Nucl. Phys. B (Proc. Suppl.)* **151** (2006) 127

[27] R. Engel, D. Heck, and T. Pierog, *Ann. Rev. Nucl. Sci.* **61** (2011) 467

[28] J. Knapp and D. Heck, Report **KfK 5196B** (1993), Kernforschungszentrum Karlsruhe; for an up-to-date version see http://www-ik.fzk.de/corsika/usersguide/corsika_tech.html

[29] R. Brun and D. Lienhart, HBOOK User's Guide, CERN Program Library **Y250** (1987)

[30] T. Sjöstrand, S. Mrenna, and P. Skands, *JHEP* **0605** (2006) 026; http://www.thep.lu.se/~torbjorn/pythia.html

[31] N.N. Kalmykov, S.S. Ostapchenko, and M.K. Alekseeva, *Proc.* 26^{th} *Int. Cosmic Ray Conf.*, Salt Lake City (USA), **1** (1999) 419

[32] Application Software Group, GEANT, CERN Program Library (1994)

[33] R.E. Cassell and G. Bower (SLAC), private communication (2002)

[34] R.L. Ford and W.R. Nelson, Report **SLAC 210** (1978), Stanford Linear Accelerator Center

[35] J.R. Cudell et al., *Phys. Rev.* **D61** (2000) 034019

[36] H. Burkhardt, S.R. Kelner, and R.P. Kokoulin, Report **CERN-SL-2002-016** (AP) (2002), CERN Geneva

[37] S. Martinez et al., *Nucl. Instr. Meth.* **A357** (1995) 567

[38] K. Bernlöhr, internal report MPI-Heidelberg (1998) (unpublished); *Astropart. Phys.* **12** (2000) 255; *Astropart. Phys.* **30** (2008) 149

[39] see e.g. Numerical Recipes, W.H. Press et al., Cambridge University Press, Cambridge (UK), (1992)

[40] F.X. Kneizys et al., The MODTRAN 2/3 Report and LOWTRAN 7 Model, Phillips Laboratory, Hanscom AFB, MA 01731-3010 (USA), (1996) available from: http://www.gps.caltech.edu/~vijay/pdf/modrept.pdf

[41] M. Risse and D. Heck, *Astropart. Phys.* **20** (2004) 661

[42] A. Gascón and A. Bueno, Auger technical note **GAP 2012-125** (2012); http://www.auger.org/admin-cgi-bin/woda/gap_notes.pl#; A. Bueno and A. Gascón, *arXiv: 1301.2672* (2013); R. Barceló, J.I. Illana, M.D. Jenkins, and M. Masip, *Phys. Rev.* **D83** (2012) 034027

[43] D. Heck, Report **FZKA 7366** (2008), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/~heck/publications/

[44] T. Pierog, R. Engel, and D. Heck, *Proc.* 31^{st} *Int. Cosmic Ray Conf.*, Lodz (Poland), (2009) contr. 0425

[45] T. Pierog, R. Engel, D. Heck, and R. Ulrich, *Proc.* 32^{nd} *Int. Cosmic Ray Conf.*, Beijing (China), (2011) contr. 1170

[46] H.-J. Drescher and G.R. Farrar, *Phys. Rev.* **D67** (2003) 116001

[47] T. Huege, M. Ludwig, and C.W. James, *AIP Conf. Proc.* **1535** (2013) 128

[48] C.W. James, H. Falcke, T. Huege, and M. Ludwig, *Phys. Rev.* **E84** (2011), 056602

[49] M. Ludwig and T. Huege, *Astropart. Phys.* **34** (2011) 438

[50] D. Heck, KIT-Scientific Working Papers **KIT-SWP-2** (2012), Karlsruhe Institute of Technology; http://digbib.ubka.uni-karlsruhe.de/volltexte/1000028191; http://www-ik.fzk.de/~heck/publications/

[51] D. Heck and R. Engel, KIT Scientific Working Papers **KIT-SWP-5** (2013), Karlsruhe Institute of Technology; http://digbib.ubka.uni-karlsruhe.de/volltexte/1000034601; http://www-ik.fzk.de/~heck/publications/

[52] D. Heck and J. Knapp, Report **FZKA 6097** (1998), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/~heck/publications/

[53] L. Cazon et al., *Astropart. Phys.* **36** (2012) 211

[54] S. Sharma et al., *MPI-Runner for CORSIKA Simulations (Algorithms)*, available with the CORSIKA files in subdirectory `doc/MPI-Runner_PHYSICS.pdf`; S. Sharma and G. Poghosyan, *User Guide of MPI-Runner for parallel CORSIKA simulations on computing systems supporting Message Passing Interface*, available with the CORSIKA files in subdirectory `doc/MPI-Runner_GUIDE.pdf`

[55] P. Homola et al., *Comp. Phys. Comm.* **173** (2005) 71; *Comp. Phys. Comm.* **184** (2013) 1468

[56] D. Heck, Report **FZKA 7082** (2004), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/~heck/publications/

[57] D. Heck, Report **FZKA 7254** (2006), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/~heck/publications/

[58] M. Hillas, *Nucl. Phys. B (Proc. Suppl.)* **52B** (1997) 29

[59] M. Kobal (P. Auger Collaboration), *Astropart. Phys.* **15** (2001) 259

[60] M. Doert and D. Heck, KIT Scientific Working Papers **KIT-SWP-1** (2012), Karlsruhe Institute of Technology; http://digbib.ubka.uni-karlsruhe.de/volltexte/1000027220; http://www-ik.fzk.de/~heck/publications/

[61] E. Andres et al. (AMANDA Collaboration), *Astropart. Phys.* **13** (2000) 1

[62] J. Ahrens et al. (IceCube Collaboration), *Astropart. Phys.* **20** (2004) 507

[63] D. Heck, Report **FZKA 6954** (2004), Forschungszentrum Karlsruhe; http://www-ik.fzk.de/~heck/publications/; F. Schröder, Diss. University Wuppertal (2001); Report **WUB-DIS 2001-17** (2001), Universität Wuppertal; http://elpub.bib.uni-wuppertal.de/edocs/dokumente/fb08/diss2001/schroeder/f080105.pdf

[64] M. Haffke (Univers. Dortmund, Germany), private communication (2007)

[65] http://www.ngdc.noaa.gov/geomag/

[66] T. H. Burnett et. al., *Phys. Rev.* **D35** (1987) 824

[67] A. S. Goldhaber, *Phys. Lett.* **B 53** (1974) 306

[68] T.K. Gaisser and A.M. Hillas, *Proc.* 15^{th} *Int. Cosmic Ray Conf.*, Plovdiv (Bulgaria), **8** (1977) 353

[69] G.J. Alner et al. (UA5 Collaboration), *Phys. Rep.* **154** (1987) 247

[70] F. Abe et al. (CDF Collaboration), *Phys. Rev.* **D41** (1990) 2330; *Phys. Rev. Lett.* **61** (1988) 1819

[71] R. Harr et al., *Phys. Lett.* **B401** (1997) 176

[72] S.J. Sciutto, arXiv: *astro-ph/9911331* (1999)

[73] Handbook of Chemistry and Physics, 67^{th} Edition, ed. R.C. Weast, The Chemical Rubber Co., Cleveland (USA), (1986) F141

[74] J. Linsley, private communication by M. Hillas (1988)

[75] H. Ulrich, K.H. Kampert, and H.O. Klages, Auger technical note **GAP 1998-043** (1998); http://www.auger.org/admin-cgi-bin/woda/gap_notes.pl#

[76] http://nssdc.gsfc.nasa.gov/space/model/atmos/msise.html

[77] P. Lipari, private communication by P. Steffen (1998)

[78] M. Will and B. Keilhauer, Auger technical note **GAP 2011-133** (2011); http://www.auger.org/admin-cgi-bin/woda/gap_notes.pl#

[79] B. Keilhauer et al., *Astropart. Phys.* **22** (2004) 249

Index

Preprocessor Options

Index

Steering Keywords