

FULL-POTENTIAL PROGRAM PACKAGE" LMTART 6.20" *USER's MANUAL*

S. Y. Savrasov

Max-Planck Institute fuer Festkoerperforschung, D-70569 Stuttgart, Germany.
Department of Physics and Astronomy, Rutgers University, Piscataway, NJ 08854.

October 12, 2000

Contents

1	INTRODUCTION	4
2	WHAT'S NEW	6
3	INSTALLATION	7
4	RUNNING LMTART	9
5	BANDLAB AND VISUALIZATION ISSUES	11
6	MAIN CONTROL FILE: INIFILE	12
6.1	<SECTION=HEAD> ! Project Head	13
6.2	<SECTION=CTRL> ! Control Parameters	13
6.3	<SECTION=EXCH> ! Exchange Correlation	15
6.4	<SECTION=ITER> ! Iterative Procedures	16
6.5	<SECTION=MAIN> ! Main Atomic Data:	17
6.6	<SECTION=SORT> ! Sort Data	19
6.6.1	<Subsection=LMTO> ! Valence states:	22
6.6.2	<Subsection=SEMI> ! Semicore states:	23
6.7	<SECTION=FFTS> ! FFT Grids:	24
7	STRUCTURE CONTROL FILE: STRFILE	26
7.1	<SECTION=HEDS> ! Structure title:	26
7.2	<SECTION=CTRS> ! Control Structure:	26
7.3	<SECTION=TRAN> ! Primitive Translations:	28
7.4	<SECTION=BASS> ! Basis Atoms :	29
7.5	<SECTION=DIST> ! Displacement Field:	29
7.6	<SECTION=STRN> ! Strain Matrix:	29
7.7	<SECTION=SYMM> ! Symmetry Group:	30
7.8	<SECTION=ZONE> ! Brillouin Zone :	31

7.9	<SECTION=DIRS> ! High-Symmetry Lines:	31
7.10	<SECTION=PLOT> ! Settings to Plot Bands:	31
8	INPUT CHARGE DENSITY FILE: SCSFILE	32
9	INPUT FOR CORRELATED ELECTRONS: HUBFILE	33
9.1	<SECTION=CTRL> ! Control Parameters:	36
9.2	<SECTION=CORR> ! Correlated States	37
9.3	<SECTION=DHUB> !Partial occupancies:	38
10	INPUT FOR TIGHT-BINDING: HOPFILE	39
10.1	<SECTION=CTRL> ! Control Parameters:	40
10.2	<SECTION=TBAS> ! Tight-Binding Basis:	40
10.3	<SECTION=HOPE> ! Description of Hoppings:	40
11	OUTPUT MESSAGES: OUTFILE	41
11.1	Reading input data	41
11.2	Finding Crystal Group	46
11.3	Finding Spherical Harmonics Expansions	48
11.4	Determining MT-spheres	49
11.5	Generating FFT Grid	50
11.6	Generating Plane Waves	50
11.7	Building/Reading Input Charge Density	51
11.8	Making PseudoHankel Functions	51
11.9	Preparing Fourier Transform for pseudoLMTOs:	53
11.10	Generating k-grid	53
11.11	Preparing Structure Constants	53
11.12	Finding Full Potential	54
11.13	Calculating Energy Bands	55
11.14	Brillouin Zone Integrals	56
11.15	Constructing Charge Density	58
11.16	Renormalizing Core Levels	58
11.17	Evaluating Total Energy	59
11.18	Evaluating Forces	60
11.19	Mixing Charge Densities	61
12	ADDITIONAL INPUT: KOVFILE	63
13	DRAWING THE BANDS: FATFILE	70
13.1	<SECTION=CTRL> ! Control Parameters:	70
14	COMPUTING DOS: DOSFILE	71
14.1	<SECTION=CTRL> ! Control Parameters:	71
15	FINDING OPTICS: OPTFILE	72
15.1	<SECTION=CTRL> ! Control Parameters:	72

16 CORE MEMORY	73
17 ERROR MESSAGES	74
17.1 Errors connected with input	74
17.2 Errors connected with iterative procedures	74
17.3 Other errors	74
18 VERSIONS DIFFERENCES	75
19 Acknowledgements	76
20 COPYRIGHT	76

1 INTRODUCTION

The full-potential linear-muffin-tin-orbital (Ref. [1]) (FP-LMTO) programs described here are designed to perform band structure, total energy and force calculations within the methods of density functional theory (DFT) (Refs. [2, 3, 4]). Main features include:

- (i) Local (spin) density approximation (LSDA) available in many parametrizations together with the gradient corrected density functionals (GGA91 & GGA96).
- (ii) Multiple- κ LMTO (possibly tight-binding) basis sets and many panel technique.
- (iii) Total energy and force calculations for determining the equilibrium structure and phonons.
- (iv) LDA+U method for strongly correlated systems.
- (v) Spin-orbit coupling for heavy elements.
- (vi) Finite temperatures
- (vii) Full three dimensional treatment of magnetization in relativistic calculations including LDA+U
- (viii) Non-collinear magnetism
- (ix) Tight-binding regime
- (x) Hoppings integrals extraction regime
- (xi) Optical properties

The LmtART is written on FORTRAN90 and uses fully dynamical memory scheme. No additional recompilations is required when changing numbers of atoms, spins, plane waves, l_{\max} 's, etc. The simplest input to LmtART involves only atomic charges of the atoms as well as crystal structure.

There are two basic regimes for working with LmtART program.

- The self-consistent charge density calculation.
- Physical properties (such as electronic structure, optical proeprties, etc.) calculation.

Windows 95/98/NT written commercial software BandLab should be used to do all visualizations such as energy bands, bands characters, densities of states, etc. This software can be also used to run LmtART itself with mouse-click operations, prepare input files and read output files. Check <http://www.quantsims.com> for the latest verisons of the software.

The LmtART works with three different methods:

- ASA: overlapping atomic spheres, potential is non-spherical inside the spheres, no interstitial region. Fast and dirty: provides reasonably good bands, but is not sufficiently accurate for phonons and distortions.

- PLW (plane waves): non-overlapping muffin-tin (MT) spheres, potential is expanded in spherical harmonics inside the spheres and Fourier transformed in the interstitial region. Provides the best accuracy at the price of increasing the computation time. A short description of this method can be found in Ref. [6].
- FTB - tight binding regime. If hopping integrals are explicitly specified, LmtART can run in tight-binding mode.

About the notations in this document:

- all file names like **nio.ini**, **main.exe**, are boldfaced.
- all directory names like */lmtart/run/* are italicized.
- capitalized names like INIFILE, STRFILE are made to shorten references to the MAIN INPUT CONTROL FILE (for INIFILE), STRUCTURE CONTROL FILE (for STRFILE), etc.

2 WHAT'S NEW

This is a third edition of the programs. While the first edition included three independent packages NMTASA, NMTCEL, and NMTRUN, the second and the third edition combines NMTASA and NMTPLW techniques while NMTCEL method has been removed due to rare use. The code combining NMTASA and NMTPLW packages is what is called LmtART program. Unfortunately, if one plans to use linear-response programs like PHNPLW, MAGPLW, MAGASA, one still has to use the old NMT* packages to generate the self-consistent densities. Linear-response programs are not yet rewritten on FORTRAN90 and are not adjusted to the output produced by the LmtART. However if the main goal is to do electronic structure/total energy/force calculations it is strongly recommended to use LmtART due to its simplified input and dynamical memory features.

The following features have been added to version 6.x compared to version 5.x:

- (a) Finite temperatures
- (b) Full three dimensional treatment of magnetization in relativistic calculations including LDA+U
- (c) Non-collinear magnetism
- (d) Tight-binding regime
- (e) Gaussian broadening k-space integration
- (f) Optical properties

Essential modifications:

- All input/output files are FORMATTED. This gives computer/system independent input/output.
- Structure of HUBFILE has been altered.

Visualization software for windows 95/98/NT, BandLab, should be used to do all visualizations such as energy bands, bands characters, densities of states, etc. This software can be also used to run LmtART itself with mouse-click operations.

3 INSTALLATION

In this section the directories used for running the programs and storing input/output data will be described. LMTART is located in directory: */lmtart/*. It contains several subdirectories:

- */lmtart/run/* - directory containing the source code ***.f** (text of the program written on FORTRAN90), object files ***.o** and executable file usually named as **main.exe**.
- */lmtart/dat/* - directory with the input/output data files. Usually, many subdirectories are created here according to the element (compound) name to be calculated. */lmtart/dat/Samples/* directory contains sample input/output data files for many materials which have been calculated using the LMTART.

There also exists another important directory named */atomdat/*. It contains the data for each element of the periodic table. There are several kinds of data files here: **rat.***, input files for making self-consistent free-atom calculation; **den.***, self-consistent free-atom densities calculated using the Libermann program; **lmt.***, LMTO basis sets described for each element of the periodic table. (There also are structure data files **str.fcc**, **str.bcc**, etc describing standard crystal structures stored here for convenience.) By default, the LMTART will use the data stored in */atomdat/*. For example, after setting the atomic charges, the LMTART will find self-consistent free-atom densities in this directory in order to produce the initial guess to the self-consistent crystal density using the Mattheiss procedure. The LMTO basis sets should normally not be described in the input files of the LMTART since, by default, after reading the atomic charges, the LMTART finds corresponding **lmt.*** files in the */atomdat/* and will set LMTO basis automatically. There are, of course, ways to avoid the default settings.

All programs and data files are tared, and gzipped into 2 files named as **lmtart.tar.gz** and **atomdat.tar.gz**. To unpack them use the following commands:

1. **gunzip lmtart.tar.gz**
2. **tar -x -f lmtart.tar**

Repeat these steps for **atomdat.tar.gz**. The directory trees will be created automatically.

To be able to run LMTART it is necessary to compile the source data files. A few comments must be said here.

1. Edit the file **ini_setup.f** and specify the path to the *scratch* and *atomdat* directories. Also check that other items match your computer settings.
2. Edit the file **lib_timel.f** and specify the call to the *system* subroutine to learn CPU time.
3. The maximum size of every array (such as maximum number of atoms, l_{\max} , etc) in the program should never be touched since all the arrays are allocated dynamically. The file PARAM.DAT existing in the old NMT* versions is now removed.
4. Compile all programs, link them to get executable file **main.exe**. Compilation is done in two steps: first compile the file **mod_dimart.f** and other **mod_*.f** files. These files contain modules which will be included in other subroutines. After **mod*** files are compiled, compile all other

*.f files. Under UNIX, using AIX XL Fortran Compiler this looks like: **xlf -cOw *.f**, which will compile only, with optimization, and will suppress all warning messages. The command **xlf -cCg *.f** will compile only, suppress optimization and provide debugging information. To link, use the command **xlf *.o -o main.exe**. To create a load map use the command **xlf *.o -o main.exe -bloadmap:map**. At the end of the map file a total amount of the static memory allocated by the program is printed out. It is less than 2 MByte. The actual core memory requirement depends on the number of atoms and other input data. The LMTART prints out the allocated memory for each particular set-up during the execution.

4 RUNNING LMTART

One runs LmtART program either interactively or using a batch job. There are three control lines read by the LmtART at the beginning of its execution. The first line characterizes the project title. The second line provides input information by establishing input files read by the LmtART. The third line provides run mode information by setting the output files produced by the LmtART. The input and output files used by the LmtART have standard naming convention: the file names are set according to the project title, the files extensions characterize data stored. As an example, if one does the calculation of the electronic structure for NiO, one first creates the directory `/lmtart/dat/nio/`. This directory will contain all input and output files produced by the LmtART. Second, one has to do self-consistent total-energy calculation for NiO. One can entitle this project as "total" or "scf" or just "nio". (Another way to entitle projects is to relate the calculation according to approximations used, as e.g. "asa" or "plw".) Let's pick up **nio**. There are two basic input files which are required to run LmtART: INIFILE which contain atomic data, and STRFILE which contain crystal structure. They have standard extensions **.ini** and **.str**. Assuming that the project is called **nio**, the filenames will correspondingly be **nio.ini** and **nio.str**. The same naming convention is valid for all other files, there is a self-consistent charge density file (SCFFILE) which will be called **nio.scf**, there is a standard output file (OUTFILE) which will be called **nio.out**, etc. Understanding this naming convention, it is easy to understand the input lines read by the LmtART during its run:

- **PROJECT**. Set project title as the first input line while starting LmtART. This will be the word **nio** for our example.
- **INPINFO**. Set input file extensions separated by sign "+" at the second line. This is in simplest case just **ini+scs+str** meaning that only **nio.ini** and **nio.str** are supposed to be read by the LmtART. If starting charge density is known from the previous run of the LmtART (SCSFILE with extension **.scs**), the program also reads **nio.scs**. If **nio.scs** is absent the program will create it. There also are two more input files which can be used by the LmtART: HUBFILE (extension **.hub**) for setting up the data of strongly-correlated electrons, e.g. for LDA+U calculations; HOPFILE (extension **.hop**) is used to withdraw hopping integrals for building tight-binding parameters. Therefore, the INPINFO line may include such combinations as **ini+str+scs+hop**, **ini+scs+str+hub**, **ini+str+hub+hop**, etc. depending on the aim of the calculation. The following lists possible combinations of INPINFO string

ini+str+scs: simplest input which requires performing self-consistent calculation or properties calculation. For this mode INIFILE, and STRFILE should be created (see sections INIFILE and STRFILE).

ini+str+scs+hub: input which requires performing self-consistent calculation or properties calculation using LDA+U method. For this, in addition to INIFILE and STRFILE, HUBFILE should be created. (see section HUBFILE)

ini+str+hop: input which requires performing properties calculation using tight-binding method. The hopping integrals should be listed in HOPFILE (see section HOPFILE).

ini+str+hop+hub: input which requires performing self-consistent calculation or properties calculation using tight-binding method. This corresponds to static mean-field solution of the Hubbard model.

ini+str+scs+hop: input which requires hopping integrals withdrawal. Note that screened, tight-binding LMTO method should be used in this mode.

ini+str+scs+hop+hub: input which requires hopping integrals withdrawal within LDA+U. In fact, the simplest way to work with INPINFO is to use **ini+scs+str+hop+hub** as the input line. For this, INIFILE, STRFILE, HOPFILE, and HUBFILES should be manually prepared while SCSFILE will be created automatically or it can be used if this continuation of self-consistency or properties calculation. Both HOPFILE and HUBFILE in its simplest form contain just several control lines which switch off all possible options.

- **RUNMODE**. The third line contains information about the output files produced by the LmtART during its run. The most important is SCFFILE (extension **.scf**) containing self-consistent charge density. Another file is OUTFILE (extension **.out**) containing the total energy, and other useful information. The following is the list of RUNMODE strings allowed to do self-consistent calculations:

scf: tells the program to make self-consistent calculation for NiO and to store the charge density in **nio.scf**.

scf+out: tells the program to make self-consistent total energy calculation for NiO and to store the charge density in **nio.scf** as well as to produce standard output file **nio.out**.

scf+hbr+out: If LDA+U is running use **scf+hbr** keywords. This will tell the program to create HBRFILE which is the output file containing LDA+U information.

scf+con: if LMTO structure constants take some time to compute at every self-consistent iteration, they can be stored by adding **+con** flag into RUNMODE string.

- To do properties calculations, the following RUNMODEs should be used:

grp: withdraws crystal group information into GRPFILE. The point group can be visualized using BandLab software

fat: withdraws fat bands information into FATFILE. The energy bands and their characters can be visualized using BandLab software

dos: withdraws density of states information into DOSFILE. The density of states and partial contributions can be visualized using BandLab software.

hpp: withdraws hopping matrix elements into HPPFILE. The TB-models can be built using BandLab software. Note that this mode requires HOPFILE in the INPINFO line.

pot: withdraws full potential information into POTFILE. The fullpotential can be visualized using BandLab software.

opt: withdraws optical properties information into OPTFILE. The optical properties can be visualized using BandLab software

frs: (not yet implemented): withdraws Fermi surface information into FRSFILE. The Fermi surface can be visualized using BandLab software.

chi: (not yet implemented): withdraws susceptibility information into CHIFILE. $\chi(\mathbf{q}, \omega)$ can be visualized using BandLab software.

5 BANDLAB AND VISUALIZATION ISSUES

Windows 95/98/NT written software BandLab can be used to set up input files and analyze output files. This software is necessary to visualize all data withdrawn from the LmtART code.

6 MAIN CONTROL FILE: INIFILE

The main control file of the full-potential package has extension `.ini` and is called INIFILE. The INIFILE is separated into several sections, and each of the sections contains certain set of data described by the keywords. An example of this file for NiO is given below

```
<FILE=INIFILE,INPUT=MODERN,TRACE=FALSE>
*****
<SECTION=HEAD>          ! PROJECT HEAD:
title =nio
<SECTION=CTRL>         ! CONTROL PARAMETERS:
FulPot=PLW              ! set: ASA / PLW / FTB
<SECTION=EXCH>        ! EXCHANGE CORRELATION:
LDA  =Vosko             ! set: none/Barth/Gunn/etc.
GGA  =none              ! set: none/91/96
<SECTION=ITER>        ! ITERATIVE PROCEDURES:
Niter1= 50              ! # of iterations in SCF loop
<SECTION=MAIN>        ! MAIN ATOMIC DATA:
Natom = 4               ! # of atoms in the unit cell
Nsort = 3               ! # of sorts in the unit cell
Nspin = 2               ! # of spins
Par0  = 7.92600         ! lattice parameter in a.u.
VVO   = 1.00000         ! volume compression
Is(:) = 1  2  3  3
<SECTION=SORT>        ! SORT DATA:
Name  = Ni1             ! atom label
Znuc  = 28.0000         ! nuclear charge
Smts  = 2.17900         ! non-overlapping MT-sphere
Split = 0.50000         ! initial spin splitting (Ry)
<SECTION=SORT>        ! SORT DATA:
Name  = Ni2             ! atom label
Znuc  = 28.0000         ! nuclear charge
Smts  = 2.17900         ! non-overlapping MT-sphere
Split =-0.50000        ! initial spin splitting (Ry)
<SECTION=SORT>        ! SORT DATA:
Name  = 0               ! atom label
Znuc  = 8.00000         ! nuclear charge
Smts  = 1.78300         ! non-overlapping MT-sphere
Split = 0.00000         ! initial spin splitting (Ry)
<SECTION=FFTS>        ! FFT GRIDS:
Ndiv(:)= 4  4  4       ! Tetrahedron mesh
Ndic(:)= 1  1  1       ! Tetrahedron mesh for semicore
Nfft(:)= ?             ! FFT mesh will be determined below
```

The first control line in the INIFILE is `<FILE=INIFILE,INPUT=MODERN,TRACE=FALSE>`. The LmtART calls the corresponding subroutine to read INIFILE according to the command `FILE=INIFILE`.

Next command INPUT=MODERN indicates that the INIFILE corresponds to the LMTART program, and not to the early NMT* versions. Another option, INPUT=CLASSIC, is reserved for reading old INIFILES used by the NMT* programs. This option is not currently available. The command TRACE=FALSE does nothing, but changing it to TRACE=TRUE will run the LMTART program in step-by-step mode. This first control line should always be present in the INIFILE.

After this control line there are several sections. Each section starts by typing the command line <SECTION=Section Name>. The INIFILE command interpreter is NOT case sensitive! Comment lines can be placed anywhere inside the INIFILE. Comment line starts either with "*" or with "!" sign as it is standard for FORTRAN programming. Sign "!" can also be used for in-line comments, everything beyond "!" sign is ignored. Note that the INIFILE will be overwritten during the execution.

6.1 <SECTION=HEAD> ! Project Head

The first section in the INIFILE is <SECTION=HEAD>. This is an optional section and not necessarily to be presented. It contains a single keyword "title" which specifies the title of the compound.

6.2 <SECTION=CTRL> ! Control Parameters

The second section in the INIFILE is <SECTION=CTRL>. It contains control parameters information. All the parameters in this section are optional, if certain keyword is not present, it will take its default value. In the sample INIFILE above, the only opened keyword is *FulPot* which has the value *PLW*. This sets full potential plane wave based calculation of the electronic structure. The full set of keywords which can be opened in this section is given below:

```
<SECTION=CTRL>           ! CONTROL PARAMETERS:
Lift  =SCF                ! set: Struc / Bands / SCF
Lmto  =Bare               ! set: Bare / Screened / Rspace
FulPot=PLW                ! set: ASA / PLW /FTB
RadPot=Relax              ! set: Relax / RelaxP / Froze
FrcPul=none               ! set: none / fast / full
```

- *Lift*

- *Struc*: Calculate structural data files, then stop. This includes structure constants of the LMTO method. Setting *Lift=Struc* is useful if the structural information is necessary for a number of jobs which will be executed spontaneously. If this is the case, one first runs the job with *Lift=Struc* to prepare structural information, and then runs LMTART with the self-consistent mode.
- *Bands* (default for *RUNMODE=bnd,fat,dos*): Calculate structural data files and make one calculation of the energy bands. This is useful if energy bands are necessary to calculate and store in one of the output files. No BZ integration is performed and the new charge density is not constructed.
- *SCF* (default for *RUNMODE=scf*) : The whole self-consistent looping is switched on. This normally includes preparation of structural data files, calculation of energy bands, integration over BZ, charge density construction, evaluation of total energy, forces and preparation of the new charge density for the next iteration.

- *Lmto*
 - *Bare* (default): Unscreened (original) LMTO. LMTART uses unscreened long-range LMTO representation as originally formulated in Ref.1
 - *Screened*: Screened (tight-binding, short-ranged) LMTO. The screening is done for every k-point by inverting the structure constants matrix. It has no advantages (nor fast neither more accurate) comparing to the *Bare* LMTO key but a short-range tight-binding Hamiltonian can be withdrawn in this regime. The latter is helpful for building the tight-binding parametrization of the energy bands.
 - *RSpace*: Screened (tight-binding, short-range) LMTO where the screening is done by inverting the structure constants matrix in the real space. Less stable procedure comparing to the previous option, especially for multiple-kappa basis sets. However, the real-space Hamiltonian can be most readily produced in this way.
- *FulPot*
 - *ASA*: Atomic sphere approximation will be used in the calculation. Less accurate procedure.
 - *PLW* (default): Full potential plane wave representation is used in the calculation. Most accurate procedure.
 - *FTB*: Tight-binding mode. To run tight-binding mode, hoppings integrals must be specified (see below).
- *RadPot* This parameter is used to control adjustment of the radial wave functions to the spherical part of the potential. Normally, the radial Schrödinger's equation is solved with the spherical potential at the current iteration. In case of spin polarized calculation, the equation is solved for both spin-up and spin-down potential and, therefore, radial wave functions have a spin dependence. There are however special cases when it becomes useful to solve radial Schrödinger's equation not with the spin dependent potential, but with the average potential $(V_{up}+V_{dn})/2$. This eliminates explicit dependence of the radial wave functions from the spin index. It is necessary, for example, when calculating susceptibility functions using linear response theory. Another option is provided to froze radial wave functions for one particular spherical potential and do not recalculate them at every iteration of the self-consistency. If radial wave functions are frozen, then the calculation of forces is exact in the sense that the calculated force is exact derivative of the LMTO expression of the total energy without any further assumptions.

The parameter *RadPot* can take one of the following values:

- *Relax* (default): Adjust radial wave functions to spin dependent potential. This is what is usually done and must be used in most cases.
- *RelaxP*: Adjust radial wave functions to spin average part of the potential. This eliminates the dependence of radial wave functions from the spin index. It is necessary, when calculating dynamical susceptibility functions using linear response theory and the program MAGPLW (currently is only available within the LMTO Electrons I).

- *Froze*: Do not recalculate radial wave functions. The spherical potential (for which the radial wave functions will be constructed) must be stored in the POTFILE (see below). This feature should bring complete correspondence between calculated total energies and forces if there is a trouble that the calculated forces are inaccurate. In fact, it is useful for debugging purposes only, generally, the force formulae programmed are sufficiently accurate.

- *FrcPul*

- *None* (default): no accurate atomic force calculation. The output will only contain the Hellmann-Feuynman forces which are normally not accurate at all when using the LMTO method due to the large incomplete basis set (or Pulay) corrections.
- *Full*: atomic forces including both the Hellmann-Feynmann and Pulay contributions will be evaluated. The accuracy of the forces due to nonself-consistency of the charge density can be controlled. Since evaluation of the Pulay forces is computationally demanding, switching this option is recommended after the self-consistency is reached.
- *Fast*: more fast option for force calculations but with shorter output. The accuracy of forces due to nonself-consistency cannot be controlled. In fact forces are accurate only within *FulPot=PLW* option. They are not accurate within *ASA*. See also file */lmtart/run/frc_forces.f* for more comments.

6.3 <SECTION=EXCH> ! Exchange Correlation

This section is devoted to set up approximations for the exchange–correlation functional of the density functional theory. It contains two keywords controlling LDA and GGA parts.

```
<SECTION=EXCH>          ! EXCHANGE CORRELATION:
LDA   =Vosko             ! set: none/Barth/Gunn/etc.
GGA   =none              ! set: none/91/96
```

- *LDA*

- *None*: not included.
- *Barth*: after von Barth and Hedin
- *Gunn*: after Gunnarsson and Ludqvist
- *Moruzzi*: after Moruzzi, Janak, and Williams
- *Vosko (default)*: after Vosko, Wilk, and Nussair, formally exact LDA parametrization based on the Monte–Carlo data.
- *Perdew*: after Perdew and Wang (local part of GGA-1991, very similar to Vosko, Wilk, and Nussair data)
- *Gaspar*: Gaspar, Kohn, and Sham, with no correlation.

- *GGA*

- *None (default)*: No generalized gradient approximations is used. Plane LDA calculation is done.
- *91*: Switches ON generalized gradient approximation of Perdew and Wang, 1991.
- *96*: Switches ON most recent generalized gradient approximation of Perdew et.al, 1996. Produces very similar results as GGA91.

6.4 <SECTION=ITER> ! Iterative Procedures

<SECTION=ITER> describes iterative procedure limits and mixing parameters. There are many optional keywords in this section. All of them have their default values. One might specify the maximum number of iterations, switch on or off the Broyden mixing when doing the self-consistency of the charge density and set different accuracy parameters. In the sample INIFILE for NiO only one keyword *Niter1* was opened to limit the total number of iterations. Full list of keywords and their meaning is described below:

```

<SECTION=ITER>           ! ITERATIVE PROCEDURES:
Niter1= 50                ! # of iterations in SCF loop
Admix1= 0.10000          ! initial mixing for density
Adspin= 0.30000          ! initial mixing for magnetization
Lbroy = 1                 ! Broyden mixing for low l.le.lbroy
Nbroy =15                 ! Broyden updated after Nbroy iters
Ibroy = 0                 ! Broyden switched after Ibroy iters
AdmixB= 0.30000          ! Broyden mixing parameter
AdmixH= 0.30000          ! Mixing for high l>lbroy
Epstot= .10E-06          ! total energy accuracy
Epsrho= .10E-06          ! charge density accuracy
Epsmag= .10E-06          ! magnetization accuracy

```

- *Niter1 (default is 50)*: max. number of iterations which LMTART makes in the self-consistent cycle.
- *Admix1 (default =0.1)*: starting mixing of the charge density in linear mixing scheme. During the iterations towards self-consistency the mixing will be optimally adjusted according to the Pratt scheme. This parameter is ignored if Broyden mixing (see below) is switched on.
- *Adspin (default =0.3)*: starting mixing for the magnetization in linear mixing scheme. During the iterations towards self-consistency the magnetization mixing will remain constant and will NOT be adjusted. The parameter has no effect for non-spin-polarized calculations or if Broyden mixing (see below) is switched on.
- *Lbroy (default =1)*: switches on the Broyden mixing. If -1 then Broyden is OFF, if 0 then Broyden is ON for $l=0$ component of $\rho(\mathbf{r})$, if +1,+2 then Broyden is on for $l.le.lbr$ component of $\rho(\mathbf{r})$ Recommended value is +1 since it does not take much disk space [because Broyden saves $\rho(\mathbf{r})$ for all previous iterations.]
- *Nbroy (default =15)* Broyden restarts every time after *Nbroy* iterations.

- *Ibroy* (default =0) starts Broyden after $I=Ibroy$ iterations. If $Ibroy=0$ then start immediately. If $Ibroy>0$ then first $I=Ibroy$ iterations will be done with the linear mixing scheme where the mixing parameters *Admix1* and *Adspin* are specified above.
- *AdmixB* (default=0.3): this is initial guess for Jacobian which is closely related to mixing parameter *mix* in the linear mixing scheme. It was found that *AdmixB* cannot be small and it is usually of the order 0.3-0.4
- *AdmixH* (default=0.3): this is linear minxing parameter for higher *l*-components ($l>lbr$) of the charge density. Since it is assumed that these components do not influence much the self-consistence loop, they are mixed within linear mixing scheme and do not stored for all previous iterations.
- *Epstot* (default=1.e-06 Ry): total energy convergency criterion. The program will stop if the total energy difference between two consequent iterations is less then *epstot*.
- *Epsrho* (default=1.e-06): charge density convergency criterion. The program will stop if the integral difference between two charge densities at two consequent iterations is less then *epsrho*.
- *Epsmag* (default=1.e-06): Magnetization convergency criterion. The program will stop if the integral difference between two magnetization densities at two consequent iterations is less then *epsmag*.

6.5 <SECTION=MAIN> ! Main Atomic Data:

<SECTION=MAIN> must be present in any INIFILE since it contains several data characterized the compound.

```

<SECTION=MAIN>                ! MAIN ATOMIC DATA:
Natom = 4                      ! # of atoms in the unit cell
Nsort = 3                      ! # of sorts in the unit cell
Nspin = 2                      ! # of spins
Norbs = 1                      ! 1-without/2 -with spin orbit coupling
Par0 = 7.92600                 ! lattice parameter in a.u.
VVO = 1.00000                 ! volume compression
Ovrl = 1.50000                 ! Maximum allowed overlap for ASA
Rcls = 0.00000                 ! Cluster size in TB calculation
Is(:) = 1 2 3 3
Nkap =1                       ! # of tail energies
Ek(:) =(-0.10000, 0.00000)    ! tail energies (Ry)

```

- *Natom* (necessary keyword, no default): total number of atoms per unit cell
- *Nsort* (necessary keyword, no default): the number of non-equivalent atoms
- *Nspin* :
 - 1 (default) for non-spin-polarized calculations

– 2 for spin-polarized calculations

- *Norbs*:

- 1 (*default*) without spin-orbit coupling

- 2 including effects of spin-orbit coupling. In this case the size of the Hamiltonian is doubled. This works only in combination with the switch *Nspin=2*, you cannot specify *nspin=1*, and *norbs=2*, i.e. spin-orbit coupling and NO spin-polarization. If spin-orbit coupling is ON, then spin-polarization is always assumed. In case non-magnetic calculation is required like Pb, for example specify initial splitting of the potential (see parameter *Split* below) equal to zero (by default it is not zero). If after self-consistency is reached for non-magnetic spin-orbit coupled calculation, the spin-polarization has to be included, specify some splitting, and set *Ispl=kickup* (see also below). Do not forget to set *ispl=none* after one run, since *ispl=kickup* will always split the potential at the beginning of every run.

Notes to orbital magnetism: spin-polarized spin-orbit coupled calculation makes non-zero average orbital moment. The program calculates orbital contribution to the magnetic moment and prints it out. However no contribution to the potential arises from the orbital moment in LSDA! Therefore, the spin densities remain unchanged. In all places where the magnetic moment is calculated and printed out, it is SPIN magnetic moment, WITHOUT orbital contribution. The orbital contribution is printed out separately and must be added to the spin moment in order to obtain the total magnetic moment.

Notes to the group symmetry: since spin-orbit coupling operator lowers the symmetry of crystal group, do not wonder if after switching SO-coupling, the crystal group will contain only 8 operations instead of 48 in the cubic case.

- *Par0* (*necessary keyword, no default*): lattice parameter in atomic units

- *VV0* (*default is 1.00*): if uniform compression is necessary, change V/V_0 ratio here. Do not change lattice parameter and anything else. It will be recalculated automatically.

- *Is(:)* (*necessary keyword, no default*): for each atom from 1 to *natom*, gives the sort of this atom. (The sequence of atoms should be the same as in the STRFILE, see below). This pointer sets the correspondence between atoms and sorts.

- *Nkap*: number of different tail energies $E = \kappa^2$ in the valence band in order to set multiple-kappa basis set. The default value in the ASA calculation is 1 and it is 2 for PLW calculation. To reach best accuracy, *Nkap=3* in the *FulPot=PLW* calculation should be used.

- *Ekap(:)*: energies $E = \kappa^2$ (complex numbers). If $\text{Re}(\kappa^2) > 0$ then $\text{Im}(\kappa^2)$ must be non-zero and close to 0.03 Ry (to avoid singularities in the Ewald summations). By default, these energies are negative, the first energy is chosen to be -0.1 Ry, the second one is -1.0 Ry, and the third one is -2.5 Ry. Overriding the default implementation is not recommended, negatively defined multiple-kappa basis set works well and numerically stable in most cases. Some discussion on choosing tail energies can be given: for the MT-geometry the interstitial region can be large, therefore an additional variational freedom of the basis functions is desired. For the states forming broad energy bands 2 or 3 kappa basis set must be chosen to make sure that the result of the band structure calculation is well convergent. The values for these kappa's are not that important,

the only condition is that they should be separated from each other by the energy of the order 1 Ry to avoid linear dependency of the LMTOs. Most popular choice here is negative-kappa basis set is formed with $\kappa_1^2=-0.1$ Ry, $\kappa_2^2=-1.0$ Ry, and $\kappa_3^2=-2.5$ Ry. The advantage of the negative energies is that they allow to avoid singularities of the structure constants connected with the free-electron poles. Another choice is positive-kappa basis set is formed with the first κ^2 placed in the center of gravity of the occupied band, another two kappa's are placed with the step 1 Ry above, i.e. $\kappa_1^2=0.4$ Ry, $\kappa_2^2=1.4$ Ry, and $\kappa_3^2=2.4$ Ry. Note that for the positive kappa case small imaginary part (0.03 Ry or so) must be added to avoid singularities in the structure constants. Positive kappa basis set reminds an expansion over plane waves while negative kappa basis looks closer to the LCAO (linear combination of atomic orbitals) like representation. For the ASA version usage of the multiple kappa basis is not important, single kappa basis is always OK. The tail energy can be fixed to 0 (or slightly smaller value, -0.1 Ry, to avoid structure constant singularity) as was proposed in the original paper [1].

- *Ovrl* (default = 1.2): maximum allowed overlap for atomic spheres can be controlled by this parameter. If the atomic spheres overlap strongly, the empty spheres should be introduced. This only works for ASA calculations.
- *Rcls* (necessary parameter for TB-LMTO and Tight-Binding calculations, no default): cluster size for the screened LMTOs, usually two-three coordination spheres should be included. If you specify *Lmto=Screened* or *Lmto=RSpace*, this parameter must be open in this section.

6.6 <SECTION=SORT> ! Sort Data

This section brings the atom information to the LMTART. This information is specified for every non-equivalent atom, i.e for every sort of atom. This section must be present *Nsort* times in the INIFILE where the parameter *Nsort* is specified above. There is one most important keyword in this section, *Znuc*, the value of atomic charge of the atom. Using this value, the LMTART can set all other parameters to their default values. It will use the atomic data information files located in */atomdat/* directory. Another parameter which is not necessarily to be presented but highly recommended to be opened, is *Smt*, the value of the MT-sphere. The MT-spheres are not known at the beginning of self-consistency, therefore one can open this parameter with "?" mark. The LMTART will find MT-sphere radii and write them to the INIFILE in this case. A third useful parameter is atom label, *Name*. Just entitle every atom by some label, the output file will use these labels to print out information specific for every atom.

Full set of parameters presented in this section is discussed below. Note that there are two SUB-SECTIONS which can be opened within this section. They will be discussed later on.

```

<SECTION=SORT>          ! SORT DATA:
Name = Ni1              ! atom label
Znuc = 28.0000         ! nuclear charge
Zval = 10.0000         ! # of valence electrons
Zsem = 0.00000        ! # of semicore electrons
Zcor = 18.0000        ! # of deepcore electrons
Amas = 58.7000        ! atomic mass as in periodic table
Smts = 2.17900        ! non-overlapping MT-sphere
Sasa = ?              ! ASA sphere will be determined later

```

```

LmaxV = 6           ! lmax for the potential
LmaxT = 6           ! lmax for the wave functions
LmaxB = 2           ! lmax for the basis set
Lrtv =semi         ! controls relativistic: none/semi
Icor =soft         ! soft core key: frozen/soft
Ispl =none         ! splitting key: none/always/kickup
Split = 0.50000    ! initial spin splitting (Ry)

```

- *Name (default value is just element name)*: title for every atom. Note that this character string (maximum 10 letters) will be read and widely used in the output file. Therefore it is recommended to use this parameter.
- *Znuc (necessary keyword, no default)* atomic number, (if empty sphere, specify *Znuc=0*)
- *Zval, Zsem, Zcor*: (defaults exists for every atom and stored in */atomdat/lmt.* file*) valence charge, semicore charge and deep-core charge, i.e. those atomic states which will be treated as valence bands, semicore bands and atomic levels. The program can treat three kinds of states: (i) Valence states, i.e. those which form the bands. These states form main valence panel and will be found by diagonalizing LMTO Hamiltonian. (ii) Semicore states, i.e. states which have small but negligible band width. These states have small hybridization with the valence states, and therefore, are treated in separate energy panels. They also found by diagonalizing LMTO Hamiltonian corresponding to each of the semicore state. (iii) Deep-core states, which are found by solving Dirac's equation for free atom with the potential taken as the spherical part of the crystalline potential until the MT-sphere and zero outside it. A simple rule to sort out the states over these three sets is the following: if the free-atom state (see corresponding **rat.<element>** or **lmt.<element>** file in the directory */atomdat/*) has an energy approximately above -2 Ry from the vacuum zero (or above -1 Ry from the MT-zero, but note that the latter is not known until the band structure calculation is performed), then this state should be treated as valence state and described in the main valence panel. If the state has an energy lying between -4 and -2 Ry from the vacuum zero, this state should be treated as a semicore state and must be described in one of the semicore panel. All other low-lying states can be treated as levels. This sorting has been performed and stored for each element in the periodic table. See */atomdat/lmt.* files* for the default description.
- *Amas (default is the atomic mass of the element)* atomic mass of the element. This value can be read but it is not used by the LMTART program.
- *AxMag(:)* local magnetization axis to study non-collinear magnetic structures. If spin-polarization and spin-orbit coupling are treated together, non-collinear magnetic structures can be investigated. By default magnetization axis is pointed along 001 direction. Opening this parameter allows to specify a custom magnetization axis for a particular atom. Example to set it: *AxMag(:)=1/2,0.0,sqrt(2)/2*
- *Smts (by default, this value will be calculated automatically)*:
In the *PLW* version, this is the non-touching muffin-tin sphere radius. At the beginning of calculation the MT-spheres are not normally available. It is possible to put the command *Smts = ?*

in the INIFILE, the MT-spheres will be calculated and overwritten instead of the question mark. The algorithm of finding MT-spheres is based on analyzing the crystal Hartree potential built with help of the superposed atomic charge densities. The algorithm works well in most cases. It is recommended to keep MT-spheres in the INIFILE, since if one does volume compression the MT-spheres must be rescaled accordingly. If one does the atomic movements, the MT-spheres must be chosen as non-touching for all atomic configurations, therefore one has to find minimum possible radii.

In the ASA version the MT-spheres will be blown up to atomic spheres. Normally, the blowing is done until space-filling occurs. Only relative values of MT-spheres for different atoms play a role, as they can be properly normalized according to the formula:

$$s_{\tau} = \left(\frac{\Omega_{cell}}{\sum_{\tau} \frac{4\pi}{3} w_{\tau}^3} \right)^{1/3} \times w_{\tau} \quad (1)$$

where s_{τ} is the atomic sphere radius, w_{τ} is the input mt-sphere, and Ω_{cell} is the cell volume. The actual values of the atomic sphere radii are printed out in the output file. It is however possible to control the overlap using the parameter *Ovrl* described above. If overlap between two atomic spheres will exceed the value of *Ovrl*, the blowing will stop. The LMTART not necessarily assumes that the atomic spheres are space filling.

- *Sasa*: the actual value of the atomic sphere can be printed out the INIFILE if one use the command *Sasa = ?*
- *LmaxT* (default value is 4 for ASA and 6 for PLW): maximal angular momentum l (not $l + 1$!) for the basis functions (i.e. for the decomposition of the tails coming from other atoms). Normally it is 4 in the ASA calculation, and 6 for PLW calculation
- *LmaxB* (default value is specified for every element and located in */atomdat/lmt.* file*): maximal l actually included in basis.
- *LmaxV* (default value is 4 for ASA and 6 for PLW): maximum l -value for the expansion of the charge density and the potential in spherical harmonics. Normally it is the same as *LmaxT*.
- *Lrtv*:
 - *none* (default for $Znuc < 21$) for non-relativistic calculations,
 - *semi* (default for $Znuc \geq 21$) for scalar-relativistic valence states
- *Icor*:
 - *frozen*: frozen deep core
 - *soft* (default): soft deep core (always fully relativistic)
- *Ispl, Split*: This controls the splitting of the up and down potential. *Ispl=none* (default): assumes no permanent splitting of the spin-up and spin-down components of the potential (which models in this case applying an external magnetic field). *Ispl=always*: with permanent splitting (on each

iteration); then *Split* is the splitting between spin-up and spin-down components of the potential in Ry. This is done according to: $V_\sigma(\mathbf{r}) + \frac{1}{2}\delta V(-1)^\sigma$ where δV is the splitting and σ runs from 1 to 2 for spin-up and spin-down components respectively. Note then if *Ispl=none* but *Split* is not zero, the latter will be used only at the first iteration. Specify *Ispl=none* and some non-zero splitting when doing spin-polarized calculation and starting from NON-spin-polarized charge density. Then, the input charge density file contains only charge density and no spin densities, the potential for spin-up and spin-down states is equivalent and must be splitted to push the system out of the paramagnetic solution. The magnetization density will be artificially introduced after the first iteration. At the following iterations *split* will be set to zero automatically and if the system tends to be magnetic, the self-consistent procedure should converge to it. If one continues self-consistency starting from the SPIN-POLARIZED charge density, then non-zero splitting in INIFILE will be ignored. This is done in order to perform smooth continuation of the self-consistent procedure from one run to another run. It might happens, however, that it is useful to make the splitting at the first iteration even if the input charge density is spin polarized. For example, the system is too far from the self-consistency or the previous calculation was done non-magnetic but spin-orbit coupled. (In the latter case the charge density file contains both spin-up and spin-down components which are equivalent). For this purpose specify *Ispl=kickup*. This will suppress setting *Split=0* at the first iteration.

6.6.1 <Subsection=LMTO> ! Valence states:

This optional subsection can be opened within each <SECTION=SORT>. It is devoted to describe LMTO basis set used for valence states. Another optional <SUBSECTION=SEMI> is devoted to describe semicore states. Normally, you do not open these subsections in the INIFILE since LMTO basis sets are already chosen for each element in the periodic table and stored in */atomdat/lmt.** files. In rare cases you might need to override the default settings. Below is the information of the parameters describing the LMTO basis. This information can appear once or *Nkap* times, i.e. for each value of kappa used. If the information appears once, it is assumed that it is the same for all kappas.

```
<Subsection=LMTO>                ! Valence states:
!!!!!! s p d f g states for Ekap=(-0.10000, 0.00000)
Mqn(:)= 4 4 3 * *                ! main quantum numbers
Bas(:)= 1 1 1 0 0                ! LMTO basis set
Mnu(:)= 3 3 3 0 0                ! choice of Enu(s)
Enu(:)= .50000 .50000 .50000
Dnu(:)= -1.0000 -2.0000 -3.0000
```

- !!!!! s p d f g states for Ekap=(-0.10000, 0.00000): This is a comment line.

In each of the following lines, only $LmaxB + 1$ first numbers will be read. For instance, if $LmaxB=2$ and you define something for the f-states, this information will be ignored.

- Mqn(:) (see */atomdat/lmt.** files for their default values): 4 4 3 * *: - main quantum numbers for s p d states.

- Bas(:) (see */atomdat/lmt.** files for their default values): *1 1 1 0 0* - basis set; either 1 (if the state is included in basis), or 0. Since LmaxB=2 for Ni, the only first 3 numbers are read, others are ignored
- Mnu(:) (see */atomdat/lmt.** files for their default values): *3 3 3 0 0* - choice of E_ν ; for each state, it may be:
 - 0: E_ν is taken from the *Eny* line and fixed throughout the iterations
 - 1: D_ν is taken from the *Dny* line and fixed throughout the iterations. E_ν are adjusted to these *Dny*.
 - 2: E_ν is found as the energy of the bound state inside the atomic sphere. Advisable for the states which are semicore like but treated as bands in the main valence panel. If the bound state cannot be found (the eigenvalue is lying in the continuous spectrum), the E_ν will be fixed according to $D_\nu = -l - 1$
 - 3: E_ν is adjusted throughout the iterations to the center of gravity of the occupied band from the band structure calculation; at the first iteration, however, the D_ν from the *Dny* line is used, and *Eny* ignored. During the self-consistency, the *Eny* numbers will be stored to the SCFFILE. In case of restart calculation, they will be read from it to obtain a smooth continuation.
- Enu(:) (see */atomdat/lmt.** files for their default values): *0.5,0.5,0.5* - initial set of Enus.
- Dnu(:) (see */atomdat/lmt.** files for their default values): *-1.0,-2.0,-3.0,-4.0* - initial set of Dnus. Normally it is just $-l - 1$.

After the blocks distributing the valence states over different κ 's, follows the description of the semicore panels:

6.6.2 <Subsection=SEMI> ! Semicore states:

This subsection describes semicore panels, i.e. those states which have some small dispersion and should likely be treated as bands. For the NiO there is no such states either in Ni or in O. In general, one may open the following keywords for the description of the semicore panels

```
<Subsection=SEMI>          ! Semicore states:
Nsem = 0                    ! # of semicore states
```

- *Nsem* (see */atomdat/lmt.** files for the default value): number of semicore states which will be treated as band states in separate energy windows (without hybridizing with the main valence panel)
- *Lsem* (see */atomdat/lmt.** files for the default value): principal quantum # n and angular momentum of the semicore state. Use notations like *Lsem=3p*.
- *Esem* (see */atomdat/lmt.** files for the default value): the tail energy for this semicore state. It should be described as a complex number. Its value is closely related to the actual binding energy of the given state in the atomic calculation. Use notations like *Esem=(-3.0,0.0)*.

A possibility is provided that semicore states from different atoms belong to the same panel (and are therefore allowed to hybridize); In order to do this, one should indicate the same tail energy for those states. Semicore states with different tail energies are treated as independent. During the iterations, E_ν 's for the semicore states are chosen to be the binding energies for the spherical part of the potential for a given iteration.

6.7 <SECTION=FFTS> ! FFT Grids:

Optional section FFTS contains different grid information. You can change the number of k-points here. All parameters have their own default values. A full set of parameters and their meaning is explained below:

```

<SECTION=FFTS>                ! FFT GRIDS:
Nfull = 13                     ! # fully filled bands
NatEF = 5                      ! # of bands crossing Ef
Broad = 0.10000                ! Linear response broadening
EFermi= 0.00000               ! Approximate Fermi energy (Ry)
NdivDir = 20                   ! Divisions along high--symmetry lines
Ndiv(:)= 4 4 4                 ! Tetrahedron mesh
Ndic(:)= 1 1 1                 ! Tetrahedron mesh for semicore
Nfft(:)= ?                     ! FFT mesh will be determined below
EpsR = 0.02000                ! PseudoHankel accuracy
EpsG = 0.04000                ! PseudoHankel accuracy
BZM = 5.00000                 ! BZ radius
KeyT =0n                       ! Teilor key: 0n/Off }

```

- *Nfull* (default = -1) : number of filled bands in the main valence panel (above the semicore!). If *Nfull*=-1, this number will be determined automatically from the knowledge of the total valence charge and the guessed number of bands crossing the Fermi level (see below).
- *NatEF* (default =5): number of bands crossing the Fermi level or larger. This parameter is used for calculating the Fermi energy and DOS. If this number is not exactly known, or if empty bands should also be taken into account (for plotting DOS above the Fermi energy), *NatEF* may be set smaller than the true number of filled bands. One can always use combination *Nfull*=0 and *NatEF*=total # of bands. This, however, may lead to unnecessary large storage of the LMTO expansion coefficients which are written on the scratch file for the bands crossing the Fermi level.
- *Broad*: this value is not used by the LMTART program.
- *Temp*: temperature in Ry to be used in the calculation
- *EFermi* (default = 0.0 Ry): initial approximation to the Fermi energy. Usually *EFermi*=0.0 works very well.
- *BZint*: Brillouin zone integration scheme. *BZint*=*ttrs* uses tetrahedron method, *BZint*=*gaus* uses gaussian broadening

- *GBroad*: Gaussian broadening parameter in Ry to be used in combination with Gaussian broadening integration scheme.
- *Ndiv(:)*, *Ndic(:)* (*default = 6 6 6*): divisions of the Brillouin zone along three directions for the tetrahedron integration (*Ndic* sets divisions for semicore states). Every k-point is described by the set of three integers (k1,k2,k3) according to:

$$\mathbf{k} = \frac{k_1}{n_1} \mathbf{G}_1 + \frac{k_2}{n_2} \mathbf{G}_2 + \frac{k_3}{n_3} \mathbf{G}_3 \quad (2)$$

where $\mathbf{G}_1, \mathbf{G}_2$, and \mathbf{G}_3 are the reciprocal lattice vectors.

- *Nfft(:)* (*default is found automatically, use ? mark to write it to the INIFILE*): divisions of the unit cell for the fast Fourier transform. Every r-point of the FFT-grid is described by (i1,i2,i3) according to

$$\mathbf{r} = \frac{i_1}{m_1} \mathbf{R}_1 + \frac{i_2}{m_2} \mathbf{R}_2 + \frac{i_3}{m_3} \mathbf{R}_3 \quad (3)$$

where $\mathbf{R}_1, \mathbf{R}_2$, and \mathbf{R}_3 are the primitive lattice translations.

The rule of thumb is, 16 divisions between nearest neighbors guarantees the sufficient accuracy. Another estimate (useful for complex structures) is, total number of divisions $m_1 \times m_2 \times m_3$ should be not less than $4000 \times$ the number of atoms, then distribute the divisions over three lattice vectors so as to get roughly equidistant mesh.

- *EpsR* (*default = 0.02*), *EpsG* (*default = 0.04*): accuracy of matching the spherical Hankel functions in real and reciprocal space. Do not play with these parameters, always specify *epsR=0.02* and *epsG=0.04*.
- *Keyt* (*default = ON*), *Bzm* (*default = 5*): Used to accelerate Fourier transforms when calculating interstitial potential matrix elements, if *Keyt=ON*. In this case the radius of the cutoff sphere in reciprocal space is set by parameter *Bzm* times the radius of the sphere circumscribing the Brillouin zone. Usually it is 4-6. The smaller *Bzm* value the faster calculation the lower the accuracy.

7 STRUCTURE CONTROL FILE: STRFILE

STRFILE is a second input file for the LMTART code. It describes the crystal structure. An example is given below:

```
<FILE=nio.str,INPUT=MODERN>
*****
<SECTION=HEDS>           ! STRUCTURE TITLE:
Slabl =NiO
<SECTION=CTRS>           ! CONTROL STRUCTURE:
Natom = 4                 ! # of atoms in unit-cell
BtoA = 1.0000             ! orthorombicity along b
CtoA = 1.0000             ! orthorombicity along c
Istrn = 1                 ! distort cutoff sphere
Nvecs = 500              ! vectors in Evald method
Evald = 1.0000           ! splitting factor there
<SECTION=TRAN>           ! PRIMITIVE TRANSLATIONS:
1/2,1/2,1.0               ! R1x,R1y,R1z
1/2,1.0,1/2               ! R2x,R2y,R2z
1.0,1/2,1/2               ! R3x,R3y,R3z
<SECTION=BASS>           ! BASIS ATOMS :
0.0,0.0,0.0               ! for Ni1
1.0,1.0,1.0               ! for Ni2
1/2,1/2,1/2               ! for O
3/2,3/2,3/2               ! for O
```

There is a first control line establishing that the file is STRFILE and MODERN input is used. Another option INPUT=CLASSIC is planned to be added for compatibility with the NMT* programs. There are several sections which can be opened with STRFILE.

7.1 <SECTION=HEDS> ! Structure title:

Optional section to provide the title of the crystal structure. A single keyword *Slabl* is used for this purpose.

```
<SECTION=HEDS>           ! STRUCTURE TITLE:
Slabl =NiO
```

7.2 <SECTION=CTRS> ! Control Structure:

This section describes some control parameters

```
<SECTION=CTRS>           ! CONTROL STRUCTURE:
Natom = 4                 ! # of atoms in unit-cell
BtoA = 1.0000             ! orthorombicity along b
```

```

CtoA = 1.0000      ! orthorombicity along c
Ibas = 1           ! basis coordinates
Ibz = 1           ! BZ choice
Icalc = 1         ! calculator switch
Istrn = 1         ! distort cutoff sphere
Nvecs = 500       ! vectors in Evald method
Evald = 1.0000    ! splitting factor there

```

- *Natom*: (no default, this value must always be present in STRFILE): total number of atoms in the unit cell
- *BtoA*, *CtoA*: (defaults are 1.0,1.0): orthorombicity parameters
- *Ibas*: switch how to interpret atomic coordinates:
 - 1 (default): basis vectors are given in Cartesian system
 - 0: basis vectors are given in units of primitive translations
- *Ibz*: switch for Brillouin zone construction;
 - 1 (default): Brillouin zone translations are set up automatically (as reciprocal lattice translations)
 - 0: BZ-translations are read from <SECTION=ZONE> in the STRFILE. The idea is, it may be useful if the automatic Brillouin zone has too pathological shape for dividing into tetrahedra.
- *Icalc*: switch for using the calculator to translate the expressions.
 - 1 (default) the calculator is “on” for interpreting the expressions in the sections “TRAN”, “BASS...”, “STRN”, “DIST”, ”ZONE”, “DIRS”. Every line can contain any simple expressions /,*,+,-, brackets are allowed without restrictions, special functions like COS, SIN, TAN, EXP, LOG, SQRT, CBRT=X**(1/3) are allowed in the format of FORTRAN, but nesting of the special functions is not allowed. Special constant PI=3.1415... can be specified. Degrees like (...)*(...) or (...)(...) are also allowed, but the expressions in brackets cannot contain special functions listed above. Expressions are separated by commas, and at the end any comment only after ! is allowed! No letter case sensitivity is assumed. Look also at some comments in the */lmtart/run/calc.f* - file. Some simple examples are:


```

<SECTION=TRAN> ! Primitive Translations for the hexagonal lattice:
1.0 , 0.0 , 0.0      ! Ax,Ay,Az !!! <- note for the comment!”
1/2 , sqrt(3)/2 , 0.0 ! Bx,By,Bz !!! <- which is only allowed
0.0 , 0.0 , 1.0     ! Cx,Cy,Cz !!! <- when ICALC=1
<SECTION=BASS>      ! Basis in the hexagonal lattice:
0.0 , 0.0 , 0.0     ! Zn
1/2 , 1/2/sqrt(3) , 1/2 ! Zn
          
```

```

<SECTION=STRN>                ! Example of simple rotation of the coordinate system:
cos(pi/4) , sin(pi/4) , 0.0
-sin(pi/4) , cos(pi/4) , 0.0
0.0 , 0.0 , 1.00
<SECTION=STRN>                ! Example of tetragonal strain c/a=1.10, conserving the volume
:
(1.10)**(-1/3), 0, 0
0, (1.10)**(-1/3), 0
0, 0, (1.10)**(2/3)

```

– 0 - give up the calculator, use standard READ(1,*) - statement

- *Istrn*: switch for taking care of the charge density plane-waves expansion in the presence of strain:
 - 1 (default) - a sphere in reciprocal lattice which selects plane waves to expand the charge density will be distorted to some ellipsoid if the strain matrix (specified below) is not unit matrix. This is useful for the distortions like strains changing b/a, c/a ratio, etc because if one always chooses a sphere to select plane waves for the Fourier transform, during the distortions the number of plane waves is changed. The latter can in principle lead to some errors in the energy difference for two lattice configurations.
 - 0 - always use a sphere to select plane waves for the Fourier transform.
- *Nvecs* (default=500): average number of lattice vectors used in the Ewald summation. The program will take the number of vectors in reciprocal space approximately twice of *Nvecs* and the number of vectors in direct space half of *Nvecs* (see also next paragraph). The program does not check the convergency of lattice sums but prints out the accuracy of the calculation (see below) which is the relative contribution to the sum going from the largest vectors.
- *Evald* (default = 1) It should be near 1. Since the ratio between numbers of generated vectors in reciprocal and direct spaces is fixed by four (see previous paragraph), the possibility is provided to scale this ratio by a factor of α . For example if $\alpha = 1/4$, the number of generated vectors in both spaces will be the same. This scaling does not change the accuracy of the lattice sums but may accelerate the calculation of those. Parameter *Evald*= 1 was chosen from the condition of fastest calculation for system with the number of atoms of order 10 at the work station IBM-RISC 6000.

7.3 <SECTION=TRAN> ! Primitive Translations:

This is required section in the STRFILE. It gives primitive translations in units of lattice parameter. Note that if orthorombicity parameters are different from 1, the y and z coordinates of primitive translations will be automatically scaled to b/a and c/a ratios. Note that in the OUTFILE, “true” Cartesian coordinates are printed out, after all transformations.

```

<SECTION=TRAN>                ! PRIMITIVE TRANSLATIONS:
1/2,1/2,1.0                    ! R1x,R1y,R1z

```

```

1/2,1.0,1/2          ! R2x,R2y,R2z
1.0,1/2,1/2         ! R3x,R3y,R3z

```

7.4 <SECTION=BASS> ! Basis Atoms :

This is required section in the STRFILE. It gives positions of basis atoms in the cell in units of lattice parameter. Note that if orthorombicity parameters are different from 1, the y and z coordinates of these positions will be automatically scaled to b/a and c/a ratios. The positions can either be given in Cartesian coordinates or in the coordinates of primitive translations. Switch *Ibas* controls this (see above). Note that in the OUTFILE, “true” Cartesian coordinates are printed out, after all transformations.

```

<SECTION=BASS>      ! BASIS ATOMS :
0.0,0.0,0.0        ! for Ni1
1.0,1.0,1.0        ! for Ni2
1/2,1/2,1/2        ! for O
3/2,3/2,3/2        ! for O

```

7.5 <SECTION=DIST> ! Displacement Field:

This optional section gives the possibility to set displacements for every atom from their equilibrium position. Displacements are given in units of lattice parameter. Note that if orthorombicity parameters are different from 1, the y and z coordinates of these vectors will be automatically scaled to b/a and c/a ratios. By default the displacements are set to zero.

```

<SECTION=DIST>     ! DISPLACEMENT FIELD:
0.0,0.0,0.0        ! for Ni1
0.0,0.0,0.0        ! for Ni2
0.0,0.0,0.0        ! for O
0.0,0.0,0.0        ! for O

```

7.6 <SECTION=STRN> ! Strain Matrix:

This optional section allows to define an applied strain field matrix. This matrix performs linear transformation of the translation vectors and the basis positions according to: $R(\text{new})=S \cdot R(\text{old})$. For reciprocal lattice the rule is $G(\text{new})=G(\text{old}) \cdot S(-1)$. By default, it is simply the unit matrix.

```

<SECTION=STRN>    ! STRAIN MATRIX:
1.0,0.0,0.0       ! Rxx,Rxy,Rxz
0.0,1.0,0.0       ! Ryx,Ryy,Ryz
0.0,0.0,1.0       ! Rzx,Rzy,Rzz

```

The matrix is read in row by row, as:

```

do I=1,3
READ(1,*)(STRAIN(I,J),J=1,3)
enddo

```

In order to get “true” translation vectors and basis positions, the program *first* applies the orthorombic scaling:

```

DO 10 I=2,3                ! I=2 for b/a, I=3 for c/a
do IVEC=1,3
RBASR(I,IVEC)=RBASR(I,IVEC)*ORTH(I) ! direct lattice
BBASR(I,IVEC)=BBASR(I,IVEC)/ORTH(I) ! recipr. lattice
enddo
do IATOM=1,NATOM
TAU R(I,IATOM)=TAU R(I,IATOM)*ORTH(I) ! basis positions
enddo
10 CONTINUE
C and \QTR{it}{then} the strain matrix:
DO IVEC=1,3
do I=1,3
RBAS(I,IVEC)=0.DO
do J=1,3
RBAS(I,IVEC)=RBAS(I,IVEC)+STRAIN(I,J)*RBASR(J,IVEC)
enddo
enddo
ENDDO
C APPLY STRAIN FOR BASIS VECTORS: T'=S*T
DO IATOM=1,NATOM
do I=1,3
TAU(I,IATOM)=0.DO
do J=1,3
TAU(I,IATOM)=TAU(I,IATOM)+STRAIN(I,J)*TAUR(J,IATOM)
enddo
enddo
ENDDO

```

7.7 <SECTION=SYMM> ! Symmetry Group:

This optional section allows to set up symmetry group for the lattice. If it is absent, the program will determine the group automatically.

```

<SECTION=SYMM>           ! SYMMETRY GROUP:
rSys = C; KovFile = kov   ! Rotational system and KovFile.

```

rSys: either *C* for cubic systems or *H* for hexagonal systems. Cubic system contains 48 operations of a cube, hexagonal system contains 24 operations of a hexagon (it includes rotations about 60 degrees

along z-axis). Applying all symmetry operations for cubic or hexagonal group, the program picks up those which are consistent with the actual crystal structure. Non-symorphic operations are found as well. Therefore your choice is only to decide whether a particular structure belongs to the cubic or hexagonal symmetry. Note that since C or H rotational operations assume a certain coordinate system (as rotation about 60 degrees along z-axis, not x or y axes) the same coordinate system should be used to describe crystalline structure. (For example, HCP structure cannot be described in the coordinate system with the rotations about 60 degrees along x-axis.) If it is necessary to use another rotational system, set *rSys=A* (arbitrary) and give the file name of KOVFILE describing your own rotational system afterwards. See chapter ADDITIONAL INPUT:KOVFILE for details. If this section is absent, the program first considers cubic system, then hexagonal system and choice the group with the largest numbers of operations found.

7.8 <SECTION=ZONE> ! Brillouin Zone :

This optional section allows to reset the Brillouin zone from the reciprocal lattice unit cell to a custom choice. Use switch *Ibz* for activating this.

```
<SECTION=ZONE>          ! BRILLOUIN ZONE :
-1/2,-1/2,3/2           ! G1x,G1y,G1z
-1/2,3/2,-1/2          ! G2x,G2y,G2z
 3/2,-1/2,-1/2         ! G3x,G3y,G3z
```

7.9 <SECTION=DIRS> ! High-Symmetry Lines:

This optional section is used to set high-symmetry directions for calculating energy bands. The program will do this if RUNMODE is set to **Fat Bands** or **fat**. If this section is absent while RUNMODE is set to **fat**, the LMTART will try to find the default settings located in */atomdat/str.** file. If crystal structure is not found in the default list, the program will stop prompting to open this section.

```
<SECTION=DIRS>          ! HIGH-SYMMETRY LINES:
4                        ! # of directions in BZ
g-X
0.0,0.0,0.0, 1/2,0.0,0.0
g-L
0.0,0.0,0.0, 1/2,1/2,0.0
Z-R
0.0,0.0,1/2, 1/2,0.0,1/2
R-A
1/2,0.0,1/2, 1/2,1/2,1/2
```

7.10 <SECTION=PLOT> ! Settings to Plot Bands:

This section is not used by the LMTART code but it is used by the BandLab windows written software

```
<SECTION=PLOT>          ! SETTINGS TO PLOT BANDS:
X-g-L,Z-R-A.
```

8 INPUT CHARGE DENSITY FILE: SCSFILE

A third important input file to the LmtART is initial charge density distribution file shortly named as SCSFILE (extension **.scs**). If one starts to do self-consistency at the very beginning, this file does not exist and will be created automatically by the LmtART. The procedure to build starting charge density is based on the Mattheiss prescription, i.e one superposes atomic charge densities to find initial approximation to the crystal charge density. The program finds with free atom density files located in */atomdat/den.** by using atomic charges of atoms. After one run, the self-consistent charge density file called SCFFILE is created. In order to restart the calculation, one just renames SCFFILE to SCSFILE and runs LmtART once more. Since SCS and SCFFILES contain only charge density one can change numbers of k-points, FFT grids, LMTO basis sets, etc without any troubles. Moreover, if supercell calculation is necessary, one can first do the calculation for original cell, and then use SCSFILE for the supercell. The LmtART will automatically expands the data from original cell to the supercell which significantly saves the number of iterations.

9 INPUT FOR CORRELATED ELECTRONS: HUBFILE

The LDA+U method is described in Ref. [7]. It turns out to drastically improve the results (comparing to LDA) when doing the calculations of the strongly correlated systems. Another option available here is constrained LDA calculations. See Ref. [8] for a complete description. To include LDA+U and/or LDA+C option, a special HUBFILE must be created. Use *INPINFO*: **ini+str+scs+hub** to make the LmtART reading the hubfile. Use *RUNMODE*: **scf+hbr** in order to create output hubbard file which will have an extension **.hbr**. You can rename it to **.hub** file if you think execution was successful.

An example of this file for NiO system is given below.

```
<FILE=HUBFILE, INPUT=MODERN>
*****
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
Scheme = LDA+U1.3                               ! LDA+U1.# / LDA+C / LDA+CU1.#
Units = Ry                                     ! Units eV/Ry available
Yharm = Cubic                                  ! Cubic/Spherical harmonics
Tharm = Cubic                                  ! Cubic/Spherical harmonics
Rspin =One                                     ! One/Both spins to read
Rorbs =One                                     ! One/Both orbits to read
Format = Complex                               ! Real/Complex input/output
<SECTION=CORR>                                  ! CORRELATED STATES
Ncrl =2                                         ! # of correlated states
cState=Ni1@1::3d                               ! Correlated state pointer
InpSys = local ; OutSys = local                ! global/local coordinate sys
InpAxis = 0,0,1 ; OutAxis = -1,1,0            ! rotational axe
InpAngle= 0*pi ; OutAngle= 0*pi               ! rotational angle
InpInv =no ; OutInv =no                       ! apply inversion after rotat
F0= 0.58800 ; F2= 0.60123 ; F4= 0.37877      ! Slater integrals
cState=Ni2@2::3d                               ! Correlated state pointer
InpSys = local ; OutSys = local                ! global/local coordinate sys
InpAxis = 0,0,1 ; OutAxis = -1,1,0            ! rotational axe
InpAngle= 0*pi ; OutAngle= 0*pi               ! rotational angle
InpInv =no ; OutInv =no                       ! apply inversion after rotat
F0= 0.58800 ; F2= 0.60123 ; F4= 0.37877      ! Slater integrals
<SECTION=DHUB>                                  ! PARTIAL OCCUPANCIES:
cState=Ni1@1::3d                               ! spin up/dn-up/dn data are :
      yz      zx      xy      x2-y2      3z2-1      REAL,UP,LDA+U
0.9882607    0.0000203    0.0000203    -0.0001193    0.0000689      yz
0.0000203    0.9882607    0.0000203     0.0001193    0.0000689      zx
0.0000203    0.0000203    0.9882607     0.0000000   -0.0001377      xy
-0.0001193   0.0001193    0.0000000     0.9921790    0.0000000      x2-y2
0.0000689    0.0000689   -0.0001377     0.0000000    0.9921790      3z2-1
      yz      zx      xy      x2-y2      3z2-1      IMAG,UP,LDA+U
0.0000000    0.0000000    0.0000000    -0.0000001    0.0000000      yz
0.0000000    0.0000000    0.0000000     0.0000001    0.0000000      zx
0.0000000    0.0000000    0.0000000     0.0000000   -0.0000002      xy
```

0.0000001	-0.0000001	0.0000000	0.0000000	0.0000000	x2-y2
0.0000000	0.0000000	0.0000002	0.0000000	0.0000000	3z2-1
yz	zx	xy	x2-y2	3z2-1	REAL, DN, LDA+U
0.9833192	-0.0000109	-0.0000109	0.0014657	-0.0008462	yz
-0.0000109	0.9833192	-0.0000109	-0.0014657	-0.0008462	zx
-0.0000109	-0.0000109	0.9833192	0.0000000	0.0016924	xy
0.0014657	-0.0014657	0.0000000	0.2201276	0.0000000	x2-y2
-0.0008462	-0.0008462	0.0016924	0.0000000	0.2201276	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG, DN, LDA+U
0.0000000	0.0000000	0.0000000	-0.0000092	0.0000053	yz
0.0000000	0.0000000	0.0000000	0.0000092	0.0000053	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000107	xy
0.0000092	-0.0000092	0.0000000	0.0000000	0.0000000	x2-y2
-0.0000053	-0.0000053	0.0000107	0.0000000	0.0000000	3z2-1
cState=Ni2@1::3d					! spin up/dn-up/dn data are :
yz	zx	xy	x2-y2	3z2-1	REAL, UP, LDA+U
0.9833192	-0.0000109	-0.0000109	0.0014657	-0.0008462	yz
-0.0000109	0.9833192	-0.0000109	-0.0014657	-0.0008462	zx
-0.0000109	-0.0000109	0.9833192	0.0000000	0.0016924	xy
0.0014657	-0.0014657	0.0000000	0.2201276	0.0000000	x2-y2
-0.0008462	-0.0008462	0.0016924	0.0000000	0.2201276	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG, UP, LDA+U
0.0000000	0.0000000	0.0000000	-0.0000092	0.0000053	yz
0.0000000	0.0000000	0.0000000	0.0000092	0.0000053	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000107	xy
0.0000092	-0.0000092	0.0000000	0.0000000	0.0000000	x2-y2
-0.0000053	-0.0000053	0.0000107	0.0000000	0.0000000	3z2-1
yz	zx	xy	x2-y2	3z2-1	REAL, DN, LDA+U
0.9882607	0.0000203	0.0000203	-0.0001193	0.0000689	yz
0.0000203	0.9882607	0.0000203	0.0001193	0.0000689	zx
0.0000203	0.0000203	0.9882607	0.0000000	-0.0001377	xy
-0.0001193	0.0001193	0.0000000	0.9921790	0.0000000	x2-y2
0.0000689	0.0000689	-0.0001377	0.0000000	0.9921790	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG, DN, LDA+U
0.0000000	0.0000000	0.0000000	-0.0000001	0.0000000	yz
0.0000000	0.0000000	0.0000000	0.0000001	0.0000000	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000002	xy
0.0000001	-0.0000001	0.0000000	0.0000000	0.0000000	x2-y2
0.0000000	0.0000000	0.0000002	0.0000000	0.0000000	3z2-1
<SECTION=DLDA>					! PARTIAL OCCUPANCIES:
cState=Ni1@1::3d					! spin up/dn-up/dn data are :
yz	zx	xy	x2-y2	3z2-1	REAL, UP, LDA
0.9814361	0.0001127	0.0001127	-0.0003406	0.0001967	yz
0.0001127	0.9814361	0.0001127	0.0003406	0.0001967	zx
0.0001127	0.0001127	0.9814361	0.0000000	-0.0003933	xy

-0.0003406	0.0003406	0.0000000	0.9468045	0.0000000	x2-y2
0.0001967	0.0001967	-0.0003933	0.0000000	0.9468045	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG,UP,LDA
0.0000000	0.0000000	0.0000000	-0.0000233	0.0000134	yz
0.0000000	0.0000000	0.0000000	0.0000233	0.0000134	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000269	xy
0.0000233	-0.0000233	0.0000000	0.0000000	0.0000000	x2-y2
-0.0000134	-0.0000134	0.0000269	0.0000000	0.0000000	3z2-1
yz	zx	xy	x2-y2	3z2-1	REAL, DN, LDA
0.9740245	0.0005125	0.0005125	0.0055347	-0.0031954	yz
0.0005125	0.9740245	0.0005125	-0.0055347	-0.0031954	zx
0.0005125	0.0005125	0.9740245	0.0000000	0.0063909	xy
0.0055347	-0.0055347	0.0000000	0.3862361	0.0000000	x2-y2
-0.0031954	-0.0031954	0.0063909	0.0000000	0.3862361	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG, DN, LDA
0.0000000	0.0000000	0.0000000	-0.0000077	0.0000044	yz
0.0000000	0.0000000	0.0000000	0.0000077	0.0000044	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000089	xy
0.0000077	-0.0000077	0.0000000	0.0000000	0.0000000	x2-y2
-0.0000044	-0.0000044	0.0000089	0.0000000	0.0000000	3z2-1
cState=Ni2@1::3d				! spin up/dn-up/dn data are :	
yz	zx	xy	x2-y2	3z2-1	REAL,UP,LDA
0.9740251	0.0005125	0.0005125	0.0055350	-0.0031956	yz
0.0005125	0.9740251	0.0005125	-0.0055350	-0.0031956	zx
0.0005125	0.0005125	0.9740251	0.0000000	0.0063912	xy
0.0055350	-0.0055350	0.0000000	0.3862469	0.0000000	x2-y2
-0.0031956	-0.0031956	0.0063912	0.0000000	0.3862469	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG,UP,LDA
0.0000000	0.0000000	0.0000000	-0.0000077	0.0000044	yz
0.0000000	0.0000000	0.0000000	0.0000077	0.0000044	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000089	xy
0.0000077	-0.0000077	0.0000000	0.0000000	0.0000000	x2-y2
-0.0000044	-0.0000044	0.0000089	0.0000000	0.0000000	3z2-1
yz	zx	xy	x2-y2	3z2-1	REAL, DN, LDA
0.9814365	0.0001126	0.0001126	-0.0003418	0.0001973	yz
0.0001126	0.9814365	0.0001126	0.0003418	0.0001973	zx
0.0001126	0.0001126	0.9814365	0.0000000	-0.0003947	xy
-0.0003418	0.0003418	0.0000000	0.9468114	0.0000000	x2-y2
0.0001973	0.0001973	-0.0003947	0.0000000	0.9468114	3z2-1
yz	zx	xy	x2-y2	3z2-1	IMAG, DN, LDA
0.0000000	0.0000000	0.0000000	-0.0000233	0.0000134	yz
0.0000000	0.0000000	0.0000000	0.0000233	0.0000134	zx
0.0000000	0.0000000	0.0000000	0.0000000	-0.0000269	xy
0.0000233	-0.0000233	0.0000000	0.0000000	0.0000000	x2-y2
-0.0000134	-0.0000134	0.0000269	0.0000000	0.0000000	3z2-1

9.1 <SECTION=CTRL> ! Control Parameters:

No case sensitivity is assumed in the input parameters described below.

```
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
Scheme = LDA+U1.3                             ! LDA+U1.# / LDA+C / LDA+CU1.#
Units  = Ry                                   ! Units eV/Ry available
Yharm  = Cubic                               ! Cubic/Spherical harmonics
Iharm  = Cubic                               ! Cubic/Spherical harmonics
Rspin  =One                                  ! One/Both spins to read
Rorbs  =One                                  ! One/Both orbits to read
Format = Complex                             ! Real/Complex input/output
```

- *Scheme*: A number of different formulae have been programmed for the LDA+U technique. For a complete description see also file `/lmtart/run/hubpot.f`.
 - ‘LDA’: standard LDA. Can be used to withdraw LDA occupation numbers matrix.
 - ‘LDA+U1.1’: standard LDA+U. Starting occupancies must be listed (see below).
 - ‘LDA+U1.2’: LDA+U with another double counting. Together with the starting occupancies LDA occupation numbers must be given (see below).
 - ‘LDA+U1.3’: LDA+U with the average occupancies.
 - ‘LDA+U1.4’: LDA+U in spherically averaged form.
 - ‘LDA+C’: constrained LDA calculations. A constrained part of the potential must be specified (see below).
 - ‘LDA+CU1.1’: combination of constrained LDA with the LDA+U technique (scheme 1.1, 1.2, 1.3 or 1.4)
- *YHarm*: *Spherical* or *cubic* harmonics representation to be used for printout.
- *IHarm*: If input harmonics are different from the setting of *YHarm*, set this key to convert input occupancies to the proper representation.
- *RSpin*: sets how many spins to read, *one*, *both*
- *ROrbs*: sets if of diagonal spin-up,down components are to be read. *one*, *both*
- *Format* can be either *real* or *complex*.

9.2 <SECTION=CORR> ! Correlated States

```

<SECTION=CORR>                                ! CORRELATED STATES
Ncrl =2                                         ! # of correlated states
cState=Ni1@1::3d                               ! Correlated state pointer
InpSys = local ; OutSys = local                ! global/local coordinate sys
InpAxis = 0,0,1 ; OutAxis = -1,1,0            ! rotational axe
InpAngle= 0*pi ; OutAngle= 0*pi              ! rotational angle
InpInv =no ; OutInv =no                       ! apply inversion after rotat
F0= 0.58800 ; F2= 0.60123 ; F4= 0.37877      ! Slater integrals
cState=Ni2@2::3d                               ! Correlated state pointer
InpSys = local ; OutSys = local                ! global/local coordinate sys
InpAxis = 0,0,1 ; OutAxis = -1,1,0            ! rotational axe
InpAngle= 0*pi ; OutAngle= 0*pi              ! rotational angle
InpInv =no ; OutInv =no                       ! apply inversion after rotat
F0= 0.58800 ; F2= 0.60123 ; F4= 0.37877      ! Slater integrals

```

- *Ncrl*: # of states which are considered as correlated must be given. If *ncrl*=0, no correlated orbitals included.
- *cState*: pointers to the atom and orbital for every correlated state. Syntax is *El@N::nl*, where *El* is element title, *N* is atomic position as given in *STRFILE*, *nl* is main and orbit quantum numbers.
- *F0*, *F2*, *F4*, etc: for each of the correlated state (selected by atom number, main quantum number, orbital quantum number) Slater integrals must be given. For d-electrons the knowledge of on-site Coulomb U and exchange integral J defines Slater integrals as follows: $U = F^{(0)}$, $J = (F^{(2)} + F^{(4)})/14$, and $F^{(2)}/F^{(4)} = 0.625$.
- One can change or choice coordinate systems for the correlated orbitals. There are two systems: one is input system, and another one is the output system. If occupancies are not known, these two systems are the same. If occupancies are already calculated and if one wants to rotate them from one system to another one, one can use input/output coordinate system setups. If *InpSys/OutSys* keywords are set to *local*, the global coordinate system will be rotated by applying a rotational operation. The following keywords set this rotational operation:
 - *InpAxis/OutAxis*: Axis along which the rotation of the global coordinate system must be performed.
 - *InpAngle/OutAngle*: Angle of rotation the global coordinate system along the rotational axis.
 - *InpInv/OutInv*: Specifies whether to perform (*yes* or *no*) an inversional operation after rotation.

9.3 <SECTION=DHUB> !Partial occupancies:

At the end of the HUBFILE matrix of the occupation numbers and/or correction to the LDA potential for each of the correlated state must be given.

```
<SECTION=DHUB>                ! LDA+U\ Occupations:
...
<SECTION=DLDA>                ! LDA occupations:
...
<SECTION=VHUB>                ! Hubbard correction to the potential:
...
<SECTION=VCNS>                ! Constrained potential:
...
```

HUBFILES can be used in combination with tight-binding regimes ($FulPot=FTB$). In this regime, hopping integrals have to be set up using HOPFILE (see below). The HUBFILE will set the interactions and the self-consistent procedure just corresponds to solving multiband Hubbard model in the Hartree-Fock approximation.

10 INPUT FOR TIGHT-BINDING: HOPFILE

WARNING: Due to a permanent development of this part of the program, some input data may differ from realization.

This application is designed to calculate hopping matrix elements for tight-binding description of the energy bands by the LMTART program. This is only possible when using *LMTO=Screened* key. The development is not yet finished although some possibility for evaluating the hoppings is provided.

To be able to calculate hopping integrals, a special HOPFILE must be created. Use *INPINFO: ini+str+scs+hopp* to make the LMTART read the hopfile. Use *RUNMODE: hpp* in order to withdraw hoppings into the HPPFILE. Rename HPPFILE to HOPFILE if you think execution is successful.

Below is an example of HOPFILE for NiO:

```
<FILE=HOPFILE, INPUT=MODERN>
*****
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
Scheme = Bands                                ! Bands currently available
Units = Ry                                    ! Units eV/Ry available
YHarm = Cubic                                 ! Cubic/Spherical harmonics
IHarm = Cubic                                 ! Cubic/Spherical harmonics
Format = Complex                              ! Real/Complex input/output
Check = Avoid                                 ! Perform/Avoid checking the symmetry
<SECTION=TBAS>                                ! TIGHT-BINDING BASIS:
Ntbs =2                                       ! # of states to read
hState = Ni1@1::3d                            ! tb-state
InpSys = local ; OutSys = local              ! global/local coordinate system
InpAxis = 0,0,1 ; OutAxis = 0,0,1           ! rotational axe
InpAngle=-0*pi/4 ; OutAngle= 0*pi/4         ! rotational angle
InpInv =no ; OutInv =no                      ! apply inversion after rotation
hState = 0@3::2p                              ! tb-state
InpSys = local ; OutSys = local              ! global/local coordinate system
InpAxis = 0,0,1 ; OutAxis = 0,0,1           ! rotational axe
InpAngle=-0*pi/4 ; OutAngle= 0*pi/4         ! rotational angle
InpInv =no ; OutInv =no                      ! apply inversion after rotation
<SECTION=HOOPP>                               ! DESCRIPTION OF HOPPINGS:
Nhop=6                                         ! # of hopping integrals
! From:                                     To:                                     Via Connection                                     Energy:                                     Overlap:
Ni1@1-up::3d{x2-y2} | Ni1@1-up::3d{x2-y2} | 0.0,0.0,0.0 | (-0.301,0.0) | (1.0,0.0)
0@3-up::2p{z} | 0@3-up::2p{z} | 0.0,0.0,0.0 | (-0.140,0.0) | (1.0,0.0)
Ni1-up@1::3d{x2-y2} | 0@3-up::2p{z} | 1/2,1/2,1/2 | (+0.162,0.0) | (1.0,0.0)
Ni1@1-dn::3d{x2-y2} | Ni1@1-dn::3d{x2-y2} | 0.0,0.0,0.0 | (-0.301,0.0) | (1.0,0.0)
0@3-dn::2p{z} | 0@3-dn::2p{z} | 0.0,0.0,0.0 | (-0.140,0.0) | (1.0,0.0)
Ni1-dn@1::3d{x2-y2} | 0@3-dn::2p{z} | 1/2,1/2,1/2 | (+0.162,0.0) | (1.0,0.0)
```

10.1 <SECTION=CTRL> ! Control Parameters:

No case sensitivity is assumed in the input parameters described below.

- *Scheme*: Only option *Bands* is available at the moment.
- *YHarm*: The input can be given either in *spherical* or in *cubic* harmonics representation.
- *IHarm*: If input harmonics are different from the setting of *YHarm*, set this key to convert hoppings to the proper representaiton.
- *Format* can be either *real* or *complex*.
- *Check*: Symmetry of the hopping integrals can be checked (option *perform*) or avoided (option *avoid*). Performing symmetry checking can be extremely slow if many hopping integrals are calculated.

10.2 <SECTION=TBAS> ! Tight-Binding Basis:

This section describes tight-binding orbitals and the local coordinate systems in which the tight-binding orbitals are given. Suppose the goal is to find the hoppings between Ni1@1::3d{x2-y2} state and O@3::2p{z} states for the antiferromagnetic NiO. There are *Ntbs* =2 states to work with. One first describes the pointers to to the orbitals (keyword *hState*). Second, one choices the coordinate system for this orbital. There are two systems: one is input system, another one is the output system. If hoppings are not known, these two systems are the same. If hoppings are already calculated and if one wants to rotate them from one system to another one, one can use input/output coordinate system setups. If *InpSys/OutSys* keywords are set to local, the global coordinate system will be rotated by applying a rotational operation. The following keywords set this rotational operation:

- *InpAxis/OutAxis*: Axis along which the rotation of the global coordinate system is performed.
- *InpAngle/OutAngle*: Angle of rotation the global coordinate system along the rotational axis.
- *InpInv/OutInv*: Specifies whether to perform (*yes* or *no*) an inversional operation after rotation.

10.3 <SECTION=HOPP> ! Description of Hoppings:

In this part of the input you must select which of the hopping matrix elements to calculate. Hopping element is defined between two (or the same) orbitals. In the first line, total number of hopping elements (keyword *Nhop*) which is supposed to calculate must be given. Only irreducible hopping integrals have to be given, i.e. those which cannot be obtained by applying group operation.

- To select orbital "From" and To specify a sort title, then atom number as listed in STRFILE (optionally spin up/dn), then main quantum number, orbital quantum number (s for l=0, p for l=1, etc), then magnetic quantum number in brackets {}. Values of $m = -3, -2, -1, 0, +1, +2, +3$ are readable for spherical harmonics. For cubic harmonics use x, y, z when $l=1$, or $yz, zx, xy, x^2-y^2, 3z^2-1$ when $l=2$, or $x(5x^2-3), y(5y^2-3), z(5z^2-3), y(x^2-z^2), z(x^2-y^2), x(y^2-z^2), xyz$ when $l=3$. See also file `/nmt*/run/cubharm.f` for definition of cubic harmonics. Examples are: Ni1@1-up::3d{x2-y2} or Ni1@1-up::3d{x2-y2} for spin-unrestricted case; Ni1@1::3d{x2-y2} for spin restricted case.

- To select *Via Connection* set the vector connecting these to sites. The format to set this vector is given in the example above. Any of expressions understandable by *calculator* can be used to set coordinates. For the description of *calculator* see section STRUCTURE CONTROL FILE.
- *Energy* is the hopping matrix element. It is not input parameter to the main program, and it will be overwritten during the calculation.
- *Overlap* is the overlap matrix element between these two orbitals, for orthogonal representation it is either 1 for the same orbital and 0 otherwise. It is not input parameter.

In order to find irreducible hoppings remove <SECTION=HOPP> completely, and run the program in HOP mode. The LMTART will print out irreducible hoppings allowed by symmetry between the orbitals chosen.

Note that if some hopping integrals are not described, they will be zeroized automatically.

HOPFILES can be used in combination with HUBFILES within tight-binding regime (*FulPot=FTB*). In this regime, hopping integrals have to be set up using the HOPFILE and the HUBFILES will set the interactions. The self-consistent procedure just corresponds to solving multiband Hubbard model in the Hartree-Fock approximation. Use *INPINFO: ini+str+hop+hob* in this case. Use *RUNMODE: hbr* which will make a new hubbard file during the self-consistent run in tight-binding regime.

11 OUTPUT MESSAGES: OUTFILE

Here is the description of the output messages. Also the structure of the program is described

Consider one particular iteration for NiO system by the LMTART.

11.1 Reading input data

The execution of the LMTART package (source file *ini_main.f*) starts from reading the input data controlled by INIT subroutine (see file *ini_init.f*). Beginning of the OUTFILE contains information read from the INIFILE. All default settings are also printed out.

```
<FILE=nio.ini,INPUT=MODERN,TRACE=FALSE>
*****
<SECTION=HEAD>                ! PROJECT HEAD:
title =nio
<SECTION=CTRL>                ! CONTROL PARAMETERS:
Lift =SCF                      ! set: Struc / Bands / SCF
Lmt0 =Bare                     ! set: Bare / Screened / Rspace
FulPot=PLW                     ! set: ASA / PLW
RadPot=Relax                   ! set: Relax / RelaxP / Froze
FrcPul=none                    ! set: none / fast / full
<SECTION=EXCH>                ! EXCHANGE CORRELATION:
LDA =Vosko                     ! set: none/Barth/Gunn/etc.
GGA =none                      ! set: none/91/96
<SECTION=ITER>                ! ITERATIVE PROCEDURES:
Niter1= 50                     ! # of iterations in SCF loop
Admix1= 0.10000                ! initial mixing for density
```

```

Adspin= 0.30000      ! initial mixing for magnetization
Epstot= .10E-06     ! total energy accuracy
Epsrho= .10E-06     ! charge density accuracy
Epsmag= .10E-06     ! magnetization accuracy
Lbroy = 1           ! Broyden mixing for low l.le.lbroy
Nbroy =15           ! Broyden updated after Nbroy iters
Ibroy = 0           ! Broyden switched after Ibroy iters
AdmixB= 0.30000     ! Broyden mixing parameter
AdmixH= 0.30000     ! Mixing for high l>lbroy
<SECTION=MAIN>      ! MAIN ATOMIC DATA:
Natom = 4           ! # of atoms in the unit cell
Nsort = 3           ! # of sorts in the unit cell
Nspin = 2           ! # of spins
Norbs = 1           ! 1-without/2 -with spin orbit coupling
Par0 = 7.92600      ! lattice parameter in a.u.
VVO = 1.00000       ! volume compression
Ovrl = 1.50000      ! Maximum allowed overlap for ASA
Rcls = 0.00000      ! Cluster size in TB calculation
Is(:) = 1 2 3 3
Nkap =2             ! # of tail energies
Ek(:) =(-0.10000, 0.00000) ! tail energies (Ry)
      (-1.00000, 0.00000)
<SECTION=SORT>      ! SORT DATA:
Name = Ni1          ! atom label
Znuc = 28.0000      ! nuclear charge
Zval = 10.0000      ! # of valence electrons
Zsem = 0.00000      ! # of semicore electrons
Zcor = 18.0000      ! # of deepcore electrons
Amas = 58.7000      ! atomic mass as in periodic table
Smts = 2.17900      ! non-overlapping MT-sphere
Srou = 0.00000      ! circumscribed sphere
Sasa = ?            ! ASA sphere will be determined later
Rloc = 0.00000      ! nearest neighbors sphere
LmaxV = 6           ! lmax for the potential
LmaxT = 6           ! lmax for the wave functions
LmaxB = 2           ! lmax for the basis set
Lrtv =semi          ! controls relativistic: none/semi
Icor =soft          ! soft core key: frozen/soft
Ispl =none          ! splitting key: none/always/kickup
Split = 0.50000     ! initial spin splitting (Ry)
<Subsection=LMTD>   ! Valence states:
!!!!!! s p d f g states for Ekap=(-0.10000, 0.00000)
Mqn(:)= 4 4 3 * *   ! main quantum numbers
Bas(:)= 1 1 1 0 0   ! LMTD basis set
Mnu(:)= 3 3 3 0 0   ! choice of Enu(s)

```

```

Enu(:)= .50000 .50000 .50000
Dnu(:)= -1.0000 -2.0000 -3.0000
!!!!!! s p d f g states for Ekap=(-1.00000, 0.00000)
Mqn(:)= 4 4 3 * * ! main quantum numbers
Bas(:)= 1 1 1 0 0 ! LMTO basis set
Mnu(:)= 3 3 3 0 0 ! choice of Enu(s)
Enu(:)= .50000 .50000 .50000
Dnu(:)= -1.0000 -2.0000 -3.0000
<Subsection=SEMI> ! Semicore states:
Nsem = 0 ! # of semicore states
<SECTION=SORT> ! SORT DATA:
Name = Ni2 ! atom label
Znuc = 28.0000 ! nuclear charge
Zval = 10.0000 ! # of valence electrons
Zsem = 0.00000 ! # of semicore electrons
Zcor = 18.0000 ! # of deepcore electrons
Amas = 58.7000 ! atomic mass as in periodic table
Smts = 2.17900 ! non-overlapping MT-sphere
Srou = 0.00000 ! circumscribed sphere
Sasa = ? ! ASA sphere will be determined later
Rloc = 0.00000 ! nearest neighbors sphere
LmaxV = 6 ! lmax for the potential
LmaxT = 6 ! lmax for the wave functions
LmaxB = 2 ! lmax for the basis set
Lrtv =semi ! controls relativistic: none/semi
Icor =soft ! soft core key: frozen/soft
Ispl =none ! splitting key: none/always/kickup
Split =-0.50000 ! initial spin splitting (Ry)
<Subsection=LMTO> ! Valence states:
!!!!!! s p d f g states for Ekap=(-0.10000, 0.00000)
Mqn(:)= 4 4 3 * * ! main quantum numbers
Bas(:)= 1 1 1 0 0 ! LMTO basis set
Mnu(:)= 3 3 3 0 0 ! choice of Enu(s)
Enu(:)= .50000 .50000 .50000
Dnu(:)= -1.0000 -2.0000 -3.0000
!!!!!! s p d f g states for Ekap=(-1.00000, 0.00000)
Mqn(:)= 4 4 3 * * ! main quantum numbers
Bas(:)= 1 1 1 0 0 ! LMTO basis set
Mnu(:)= 3 3 3 0 0 ! choice of Enu(s)
Enu(:)= .50000 .50000 .50000
Dnu(:)= -1.0000 -2.0000 -3.0000
<Subsection=SEMI> ! Semicore states:
Nsem = 0 ! # of semicore states
<SECTION=SORT> ! SORT DATA:
Name = 0 ! atom label

```

```

Znuc = 8.00000      ! nuclear charge
Zval = 6.00000      ! # of valence electrons
Zsem = 0.00000      ! # of semicore electrons
Zcor = 2.00000      ! # of deepcore electrons
Amas = 15.9994      ! atomic mass as in periodic table
Smts = 1.78300      ! non-overlapping MT-sphere
Srou = 0.00000      ! circumscribed sphere
Sasa = ?            ! ASA sphere will be determined later
Rloc = 0.00000      ! nearest neighbors sphere
LmaxV = 6           ! lmax for the potential
LmaxT = 6           ! lmax for the wave functions
LmaxB = 1           ! lmax for the basis set
Lrtv =none          ! controls relativistic: none/semi
Icor =soft          ! soft core key: frozen/soft
Ispl =none          ! splitting key: none/always/kickup
Split = 0.00000     ! initial spin splitting (Ry)
<Subsection=LMT0>   ! Valence states:
!!!!!! s p d f g   states for Ekap=(-0.10000, 0.00000)
Mqn(:)= 2 2 * * *   ! main quantum numbers
Bas(:)= 1 1 0 0 0   ! LMT0 basis set
Mnu(:)= 3 3 0 0 0   ! choice of Enu(s)
Enu(:)= .50000      .50000
Dnu(:)= -1.0000     -2.0000
!!!!!! s p d f g   states for Ekap=(-1.00000, 0.00000)
Mqn(:)= 2 2 * * *   ! main quantum numbers
Bas(:)= 1 1 0 0 0   ! LMT0 basis set
Mnu(:)= 3 3 0 0 0   ! choice of Enu(s)
Enu(:)= .50000      .50000
Dnu(:)= -1.0000     -2.0000
<Subsection=SEMI>   ! Semicore states:
Nsem = 0            ! # of semicore states
<SECTION=OUTS>      ! OUTPUT CONTROLS:
Icon= 1 ; confile=nio.con ! structure constants
Iftr= 0 ; ftrfile=ftr    ! screened constants
Ipsi= 0 ; psifile=psi     ! wave functions
Iscr= 0 ; scrfile=/scratch/nio.scr01 ! scratch storage
Ibnd= 0 ; bndfile=nio.bnd ! band structure
Ipot= 0 ; potfile=pot     ! full potential
Ifat= 0 ; fatfile=fat     ! fat bands
Idos= 0 ; dosfile=dos     ! density of states
Iscf= 2 ; scffile=nio.scf ! charge density
Iout= 2 ; outfile=nio.out ! output file
<SECTION=FFTS>      ! FFT GRIDS:
Nfull = 13          ! # fully filled bands
NatEF = 5           ! # of bands crossing Ef

```

```

Broad = 0.10000          ! Linear response broadending
EFermi= 0.00000         ! Approximate Fermi energy (Ry)
Ndiv(:)= 4  4  4        ! Tetrahedron mesh
Ndic(:)= 1  1  1        ! Tetrahedron mesh for semicore
Nfft(:)= ?             ! FFT mesh will be determined below
EpsR  = 0.02000        ! PseudoHankel accuracy
EpsG  = 0.04000        ! PseudoHankel accuracy
BZM   = 5.00000        ! BZ radius
KeyT  =On              ! Teilor key: On/Off
<SECTION=ADDS>         ! ADDITIONAL INPUTS:
Ihub= 0 ; hubfile=nio.hub    ! Hubbard corrections
Ihop= 0 ; hopfile=hop      ! Hoppings file
Iopt= 0 ; optfile=opt      ! Optical properties
Ienr= 0 ; enrfile=enr      ! Energy bands for weights
Ipnt= 0 ; pntfile=pnt      ! List of q-points

```

This information is printed only to test the correctness of the input data. The next output lines contain the information read from the STRFILE.

```
<<<--- STRFILE READ --->>>
```

```

<FILE=nio.str,INPUT=MODERN>
*****
<SECTION=HEDS>          ! STRUCTURE TITLE:
Slabl =NiO
<SECTION=CTRS>          ! CONTROL STRUCTURE:
Natom = 4               ! # of atoms in unit-cell
BtoA  = 1.0000          ! orthorombicity along b
CtoA  = 1.0000          ! orthorombicity along c
Istrn = 1               ! distort cutoff sphere
Nvecs = 500            ! vectors in Evald method
Evald = 1.0000         ! splitting factor there
<SECTION=TRAN>         ! PRIMITIVE TRANSLATIONS:
.50000 , .50000 , 1.0000 ! R1x,R1y,R1z
.50000 , 1.0000 , .50000 ! R2x,R2y,R2z
1.0000 , .50000 , .50000 ! R3x,R3y,R3z
<SECTION=BASS>         ! BASIS ATOMS :
.00000E+00, .00000E+00, .00000E+00 ! for Ni1
1.0000 , 1.0000 , 1.0000 ! for Ni2
.50000 , .50000 , .50000 ! for 0
1.5000 , 1.5000 , 1.5000 ! for 0
<SECTION=DIST>         ! DISPLACEMENT FIELD:
.00000E+00, .00000E+00, .00000E+00 ! for Ni1
.00000E+00, .00000E+00, .00000E+00 ! for Ni2

```

```

      .00000E+00, .00000E+00, .00000E+00 ! for 0
      .00000E+00, .00000E+00, .00000E+00 ! for 0
<STRAIN MATRIX>      :
      1.0000      , .00000E+00, .00000E+00 ! Sxx,Sxy,Sxz
      .00000E+00, 1.0000      , .00000E+00 ! Syx,Syy,Syz
      .00000E+00, .00000E+00, 1.0000      ! Szx,Szy,Szz
<SECTION=STRN>      ! STRAIN MATRIX:
      1.0000      , .00000E+00, .00000E+00 ! Rxx,Rxy,Rxz
      .00000E+00, 1.0000      , .00000E+00 ! Ryx,Ryy,Ryz
      .00000E+00, .00000E+00, 1.0000      ! Rzx,Rzy,Rzz
<SECTION=SYMM>      ! SYMMETRY GROUP:
! will be determined below
<SECTION=RLAT>      ! RECIPROCAL LATTICE:
      -.50000      , -.50000      , 1.5000      ! G1x,G1y,G1z
      -.50000      , 1.5000      , -.50000      ! G2x,G2y,G2z
      1.5000      , -.50000      , -.50000      ! G3x,G3y,G3z
<SECTION=ZONE>      ! BRILLOUIN ZONE :
      -.50000      , -.50000      , 1.5000      ! K1x,K1y,K1z
      -.50000      , 1.5000      , -.50000      ! K2x,K2y,K2z
      1.5000      , -.50000      , -.50000      ! K3x,K3y,K3z
<SECTION=DIRS>      ! HIGH-SYMMETRY LINES:
ndir = 0      ! # of directions in BZ
Cell Volume = 248.9615

```

Primitive translations, positions of atoms in the basis, reciprocal lattice vectors and the Brillouin zone data are given after taking into account b/a and c/a ratios, and after applying the strain matrix. The data in direct space are given in the units of lattice parameter, the data in reciprocal space are given in units $\frac{2\pi}{a}$. The cell volume Ω_{cell} is printed in $(a.u.)^3$.

11.2 Finding Crystal Group

After INIFILE and STRFILE are read LMTART determines crystal group. The following printout is produced:

```

***** MakeGRP started ; CPU : .43000      ; CUR/MAX mem.(Mb): .000E+00/ .000E+00
Start analyzing crystal group from input data:

```

```

Rotational systems ---> Cubic   Hexagonal
Pure lattice group : 96       16
Operations with atoms : 12     2

```

Classifying atoms over different sorts :

```

PRINTOUT # 1
# of sorts according to input nuclear charges: 2
SORT 1; the following atoms are equivalent:

```

```
Atom 1; ( .00000E+00, .00000E+00, .00000E+00) :Ni1
Atom 2; ( 1.0000 , 1.0000 , 1.0000 ) :Ni2
SORT 2; the following atoms are equivalent:
Atom 3; ( .50000 , .50000 , .50000 ) :0
Atom 4; ( 1.5000 , 1.5000 , 1.5000 ) :0
```

PRINTOUT # 2

```
# of sorts according to lattice group found: 2
SORT 1; the following atoms are equivalent:
Atom 1; ( .00000E+00, .00000E+00, .00000E+00) :Ni1
Atom 2; ( 1.0000 , 1.0000 , 1.0000 ) :Ni2
SORT 2; the following atoms are equivalent:
Atom 3; ( .50000 , .50000 , .50000 ) :0
Atom 4; ( 1.5000 , 1.5000 , 1.5000 ) :0
```

PRINTOUT # 3

```
# of sorts read from input data file INIFILE: 3
SORT 1; the following atoms are equivalent:
Atom 1; ( .00000E+00, .00000E+00, .00000E+00) :Ni1
SORT 2; the following atoms are equivalent:
Atom 2; ( 1.0000 , 1.0000 , 1.0000 ) :Ni2
SORT 3; the following atoms are equivalent:
Atom 3; ( .50000 , .50000 , .50000 ) :0
Atom 4; ( 1.5000 , 1.5000 , 1.5000 ) :0
```

Warning message from <MAKEGRP>:

It is not necessarily true that sorting due to lattice group (printout # 2 above) gives exact classification of atoms over different sorts since sorting was done according to Znuc and symmetry operations. This sorting does not take into account the difference caused by, e.g. magnetic structure. However, since sorting read from INIFILE (printout # 3) is different from printout # 2, it is recommended to check input IS(iatom) array.

```
Group elements discovered for lattice: 12
Cubic rotational system is selected: C
```

```
***** MakeGRP finished; CPU : .81000 ; CUR/MAX mem.(Mb): .587 / .587
```

The determination of the crystal group is done automatically in most cases. The program checks cubic and hexagonal rotational systems and chooses the one with the largest numbers of operations found. There are three checks made. First, the program assumes that atoms with the equivalent charges are equivalent, second, the program sorts out those atoms with crystallographically non-

equivalent positions, and, third, the program uses input IS(:) array establishing non-equivalent sorts. If crystal group found in case 2 and 3 are different, a warning message is given. In principle the non-equivalence of cases 2 and 3 is possible when, e.g., doing spin-polarized calculations since crystallographically equivalent atoms may be non-equivalent due to different magnetization (example: antiferromagnetic NiO).

11.3 Finding Spherical Harmonics Expansions

The following messages prints non-zero expansion coefficients of the spherical harmonics for the charge density which are allowed by symmetry.

```
***** MakeSYM started ; CPU : .81000 ; CUR/MAX mem.(Mb): .587 / .587
Start finding LM expansions for rho(r):
```

```
Position : ( .00000E+00, .00000E+00, .00000E+00) for Ni1
LMT0-basis set is expanded in spherical harmonics up to Lmax= 6
Charge density is expanded in spherical harmonics up to Lmax= 6
Non-zero elements allowed by symmetry are the following:
l= 0 ; m=                0
l= 2 ; m=                -2 -1  1  2
l= 4 ; m=                -4 -3 -2 -1  0  1  2  3  4
l= 6 ; m=               -6 -5 -4 -3 -2 -1  0  1  2  3  4  5  6
Total # of non-zero components found 27
```

```
Position : ( 1.0000 , 1.0000 , 1.0000 ) for Ni2
LMT0-basis set is expanded in spherical harmonics up to Lmax= 6
Charge density is expanded in spherical harmonics up to Lmax= 6
Non-zero elements allowed by symmetry are the following:
l= 0 ; m=                0
l= 2 ; m=                -2 -1  1  2
l= 4 ; m=                -4 -3 -2 -1  0  1  2  3  4
l= 6 ; m=               -6 -5 -4 -3 -2 -1  0  1  2  3  4  5  6
Total # of non-zero components found 27
```

```
Position : ( .50000 , .50000 , .50000 ) for O
LMT0-basis set is expanded in spherical harmonics up to Lmax= 6
Charge density is expanded in spherical harmonics up to Lmax= 6
Non-zero elements allowed by symmetry are the following:
l= 0 ; m=                0
l= 1 ; m=                -1  0  1
l= 2 ; m=                -2 -1  1  2
l= 3 ; m=                -3 -2 -1  0  1  2  3
l= 4 ; m=                -4 -3 -2 -1  0  1  2  3  4
l= 5 ; m=                -5 -4 -3 -1  0  1  3  4  5
l= 6 ; m=               -6 -5 -4 -3 -2 -1  0  1  2  3  4  5  6
Total # of non-zero components found 46
```

```

Position : ( 1.5000 , 1.5000 , 1.5000 ) for 0
LMT0-basis set is expanded in spherical harmonics up to Lmax= 6
Charge density is expanded in spherical harmonics up to Lmax= 6
Non-zero elements allowed by symmetry are the following:
l= 0 ; m= 0
l= 1 ; m= -1 0 1
l= 2 ; m= -2 -1 1 2
l= 3 ; m= -3 -2 -1 0 1 2 3
l= 4 ; m= -4 -3 -2 -1 0 1 2 3 4
l= 5 ; m= -5 -4 -3 -1 0 1 3 4 5
l= 6 ; m= -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6
Total # of non-zero components found 46
***** MakeSYM finished; CPU : .81000 ; CUR/MAX mem.(Mb): .587 / .587

```

11.4 Determining MT-spheres

After finding non-zero expansion coefficients, the program determines MT-sphere radii:

```

***** MakeSMT started ; CPU : .81000 ; CUR/MAX mem.(Mb): .587 / .587
Start finding optimal MT spheres:

```

```

Charge checking: 28.000 27.991 for Ni1 ;/u/a4/savrasov/atomdat/den.ni
Charge checking: 28.000 27.991 for Ni2 ;/u/a4/savrasov/atomdat/den.ni
Charge checking: 8.0000 7.9975 for 0 ;/u/a4/savrasov/atomdat/den.o
Position : ( .00000E+00, .00000E+00, .00000E+00) for Ni1
MT-sphere read from input INIFILE = 2.179000
MT-sphere from Hartree potential = 2.179650
MT-sphere blowed until touching = 2.179650
MT-sphere is now set to the value = 2.179000
Position : ( 1.0000 , 1.0000 , 1.0000 ) for Ni2
MT-sphere read from input INIFILE = 2.179000
MT-sphere from Hartree potential = 2.179650
MT-sphere blowed until touching = 2.179650
MT-sphere is now set to the value = 2.179000
Position : ( .50000 , .50000 , .50000 ) for 0
MT-sphere read from input INIFILE = 1.783000
MT-sphere from Hartree potential = 1.783350
MT-sphere blowed until touching = 1.783350
MT-sphere is now set to the value = 1.783000
Position : ( 1.5000 , 1.5000 , 1.5000 ) for 0
MT-sphere read from input INIFILE = 1.783000
MT-sphere from Hartree potential = 1.783350
MT-sphere blowed until touching = 1.783350
MT-sphere is now set to the value = 1.783000

```

```

Maximum overlap allowed from setup = 1.500000
Overlap required to fill volume = 1.228859
Overlap finally set to the value = 1.228859
Nrad = 320 ; Smt = 2.179000 ; Sasa = 2.677683 for Ni1
Nrad = 320 ; Smt = 2.179000 ; Sasa = 2.677683 for Ni2
Nrad = 228 ; Smt = 1.783000 ; Sasa = 2.191055 for 0
***** MakeSMT finished; CPU : .98000 ; CUR/MAX mem.(Mb): .587 / .587

```

11.5 Generating FFT Grid

The following output contains the information about the FFT grid:

```

***** MakeFFT started ; CPU : .98000 ; CUR/MAX mem.(Mb): .587 / .587
Start building fast Fourier transform grid:

Optimal FFT divisions nfftg(:) for LDA = 28 28 28
# of points of the FFT grid within MTS= 11920
# of points of the FFT grid in the INT= 10032
Total # of FFT points generated = 21952
Sum (MTS+INT)= 21952

***** MakeFFT finished; CPU : 19.420 ; CUR/MAX mem.(Mb): .680 / .587

```

11.6 Generating Plane Waves

Next, numbers of plane waves for representing pseudocharge density and pseudopotential are generated:

```

***** MakePLW started ; CPU : 19.420 ; CUR/MAX mem.(Mb): .680 / .587
Start building plane waves:

Number of plane waves to be used 2502
Plane-wave energy cutoff will be 70.85462 Ry
Reciprocal lattice sphere radius 10.61838 2pi/a
Fast Fourier transform divisions 28 28 28
Orthorombicity optimizator gives 28 28 28

***** MakePLW finished; CPU : 23.080 ; CUR/MAX mem.(Mb): 2.00 / 2.00

```

11.7 Building/Reading Input Charge Density

MakeSCF makes input charge density file if SCSFILE is not found. If SCSFILE exists, it will be read by this subroutine.

```
}***** MakeSCF started ; CPU : 23.080 ; CUR/MAX mem.(Mb): 2.00 / 2.00
Start building initial charge density:

Charge checking: 28.000 27.991 for Ni1 ;/u/a4/savrasov/atomdat/den.ni
Charge checking: 28.000 27.991 for Ni2 ;/u/a4/savrasov/atomdat/den.ni
Charge checking: 8.0000 7.9975 for 0 ;/u/a4/savrasov/atomdat/den.o
Charge in elementary cell must be 72.00000
Charge found in elementary cell 71.98322
Renormalization of the density is 1.000233
Charge in the interstitial region 4.102908

Start reading SCFFILE:

Plane waves: old= 2502; new= 2502; coinciding 2502
***** MakeSCF finished; CPU : 89.560 ; CUR/MAX mem.(Mb): 2.00 / 5.37
```

MakeSCF makes input charge density file if SCSFILE is not found. If SCSFILE exists, it will be read by this subroutine.

11.8 Making PseudoHankel Functions

```
***** MakeHAN started ; CPU : 89.560 ; CUR/MAX mem.(Mb): 2.00 / 5.37
Start making pseudoHankel functions:

Position : ( .00000E+00, .00000E+00, .00000E+00) for Ni1
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -.100 ; p#1
''s'' 274 15.6 .99847 1.0001
''p'' 486 22.6 .99587 .99796
''d'' 802 32.7 .99327 .99031
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -1.00 ; p#1
''s'' 274 15.6 .99425 1.0002
''p'' 486 22.6 .99152 .99597
''d'' 802 32.7 .98969 .98567
l= Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S)
'' 0'' 2502 70.9 1.00000 1.0000
'' 1'' 2502 70.9 .99993 .99988
'' 2'' 2502 70.9 .99951 .99955
'' 3'' 2502 70.9 .99786 1.0028
'' 4'' 2502 70.9 .99324 1.0096
'' 5'' 2502 70.9 .98314 .99208
'' 6'' 2502 70.9 .96501 .94102
```

```

Position : ( 1.0000 , 1.0000 , 1.0000 ) for Ni2
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -.100 ; p#1
''s'' 274 15.6 .99847 1.0001
''p'' 486 22.6 .99587 .99796
''d'' 802 32.7 .99327 .99031
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -1.00 ; p#1
''s'' 274 15.6 .99425 1.0002
''p'' 486 22.6 .99152 .99597
''d'' 802 32.7 .98969 .98567
l= Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S)
'' 0'' 2502 70.9 1.00000 1.0000
'' 1'' 2502 70.9 .99993 .99988
'' 2'' 2502 70.9 .99951 .99955
'' 3'' 2502 70.9 .99786 1.0028
'' 4'' 2502 70.9 .99324 1.0096
'' 5'' 2502 70.9 .98314 .99208
'' 6'' 2502 70.9 .96501 .94102
Position : ( .50000 , .50000 , .50000 ) for 0
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -.100 ; p#1
''s'' 500 23.1 .99864 1.0001
''p'' 844 34.4 .99604 .99770
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -1.00 ; p#1
''s'' 500 23.1 .99589 1.0003
''p'' 844 34.4 .99314 .99612
l= Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S)
'' 0'' 2502 70.9 .99998 1.00000
'' 1'' 2502 70.9 .99965 1.0006
'' 2'' 2502 70.9 .99787 1.0014
'' 3'' 2502 70.9 .99222 .98928
'' 4'' 2502 70.9 .97931 .97149
'' 5'' 2502 70.9 .95642 1.0177
'' 6'' 2502 70.9 .92340 1.1365
Position : ( 1.5000 , 1.5000 , 1.5000 ) for 0
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -.100 ; p#1
''s'' 500 23.1 .99864 1.0001
''p'' 844 34.4 .99604 .99770
Basis Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S); Ekap= -1.00 ; p#1
''s'' 500 23.1 .99589 1.0003
''p'' 844 34.4 .99314 .99612
l= Nplw Ecut (Ry) RH(S)/H(S) GH(S)/RH(S)
'' 0'' 2502 70.9 .99998 1.00000
'' 1'' 2502 70.9 .99965 1.0006
'' 2'' 2502 70.9 .99787 1.0014
'' 3'' 2502 70.9 .99222 .98928
'' 4'' 2502 70.9 .97931 .97149

```

```

'' 5'' 2502 70.9 .95642 1.0177
'' 6'' 2502 70.9 .92340 1.1365
***** MakeHAN finished; CPU : 90.470 ; CUR/MAX mem.(Mb): 2.00 / 5.37

```

11.9 Preparing Fourier Transform for pseudoLMTOs:

```

***** MakeTEI started ; CPU : 90.470 ; CUR/MAX mem.(Mb): 2.00 / 5.37
Start preparing Fourier transform for pseudoLMTOs:

```

```

# of (k+G)-terms in Fourier sums 154
Brillouin zone radius calculated .8660254 2pi/a
Teilor''s sphere radius estimated 4.330127 2pi/a
Teilor''s cutoff energy estimated 11.78292 Ry

```

```

***** MakeTEI finished; CPU : 90.470 ; CUR/MAX mem.(Mb): 2.00 / 5.37

```

11.10 Generating k-grid

```

***** MakeTTR started ; CPU : 90.470 ; CUR/MAX mem.(Mb): 2.00 / 5.37
Start preparing mesh of k-points:

```

```

K-points generated for main valence panel: 13

```

```

***** MakeTTR finished; CPU : 90.570 ; CUR/MAX mem.(Mb): 2.00 / 5.37

```

11.11 Preparing Structure Constants

Structure constants for the unscreened LMTOs are calculated in **strmsh.f**

```

***** Strmsh started ; CPU : 90.610 ; CUR/MAX mem.(Mb): 2.00 / 5.37

```

```

Result from VECGEN for direct/reciprocal spaces -->
Rmax= 3.101852 ; Accuracy= .1921585E-19; # of vectors= 259
Gmax= 7.816026 ; Accuracy= .7021878E-26; # of vectors= 1033
Smax= 5.304022 ; Accuracy= .2152549E-28; # of vectors= 1257
Min.energy for using Evald''s method= -3.282133 Ry
Total # of connecting vectors found 7

```

```

Minimum difference between |k+G|^2 and kappa1^2 is .1591287

```

```

Minimum difference between |k+G|^2 and kappa2^2 is 1.591287

```

```

***** Strmsh finished; CPU : 95.980 ; CUR/MAX mem.(Mb): 2.06 / 5.37

```

The information below contains the set-up for using the Ewald method to sum up the structure constants. Minim differences between tail energies and the poles of free-electron Green function show how far is the singularity on the real axe. Note that when using the positive κ^2 a small imaginary part (approximately 0.03 Ry) must be placed to avoid this singularity.

For the screening LMTOs in the real space (option *LMTO=rSpace*), another program **scrcon.f** is used and another output is produced.

11.12 Finding Full Potential

The self-consistency is controlled by the program SCF1 (see source file **scf1.f**). At the beginning of each iteration, first, the full potential is calculated. As a result, the table below is produced in the OUTFILE. It should be noted that the boundary values of potential, $V(S)$, are given with respect to the “vacuum zero”, *i.e.*, to the energy zero of the atomic program. Once the average $V(S)$ is found, energy zero is put there, and since that the items ”Average potential in the interstitial region”, Kappa’s and band energies are given with respect to it. It is recommended to adjust the MT-radii in such a way as to make, if it is possible, the boundary potential values $V(S)$ above not very different for different atoms. The V and P values stand for the potential and pseudopotential while RO and PD values denote density and pseudodensity. M(S) is the magnetic moment within the MT-sphere, PM(S) is the pseudomagnetic moment (has no physical meaning. Notation S is for the sphere while 0 is for the atom origin.

```

***** FullPOT started ; CPU : 98.360 ; CUR/MAX mem.(Mb): 6.01 / 6.01
Input data for Ni1 in the position 1 ----->
V-up(S)= -.5894001 | RO-up(S)= .2238512E-01
P-up(S)= -.3404617 | PD-up(S)= .2244590E-01
P-up(0)= -10.00800 | PD-up(0)= .9727026E-01
V-dn(S)= -.8940006E-01 | RO-dn(S)= .2238512E-01
P-dn(S)= -.3404617 | PD-dn(S)= .2244590E-01
P-dn(0)= -10.00800 | PD-dn(0)= .9727026E-01
M(S)= .0000000E+00 | PM(S)= .0000000E+00
Input data for Ni2 in the position 2 ----->
V-up(S)= -.8940006E-01 | RO-up(S)= .2238512E-01
P-up(S)= -.3404617 | PD-up(S)= .2244590E-01
P-up(0)= -10.00800 | PD-up(0)= .9727026E-01
V-dn(S)= -.5894001 | RO-dn(S)= .2238512E-01
P-dn(S)= -.3404617 | PD-dn(S)= .2244590E-01
P-dn(0)= -10.00800 | PD-dn(0)= .9727026E-01
M(S)= .0000000E+00 | PM(S)= .0000000E+00
Input data for O in the position 3 ----->
V-up(S)= -.4396388 | RO-up(S)= .3139612E-01
P-up(S)= -.4345041 | PD-up(S)= .3147958E-01
P-up(0)= -8.940579 | PD-up(0)= .2100194
V-dn(S)= -.4396388 | RO-dn(S)= .3139612E-01
P-dn(S)= -.4345041 | PD-dn(S)= .3147958E-01
P-dn(0)= -8.940579 | PD-dn(0)= .2100194
M(S)= .0000000E+00 | PM(S)= .0000000E+00

```

```

Input data for 0      in the position  4 ----->
V-up(S)= -.4396388   | R0-up(S)= .3139612E-01
P-up(S)= -.4345041   | PD-up(S)= .3147958E-01
P-up(0)= -8.940579   | PD-up(0)= .2100194
V-dn(S)= -.4396388   | R0-dn(S)= .3139612E-01
P-dn(S)= -.4345041   | PD-dn(S)= .3147958E-01
P-dn(0)= -8.940579   | PD-dn(0)= .2100194
      M(S)= .0000000E+00 |      PM(S)= .0000000E+00
Average potential over the sphere boundaries is  -.3895194
Average potential in the interstitial region is  -.1135063
Total charge in the interstitial region must be  4.103864
Total charge found via fourier transform is  4.103864
Auxiliary density renormalization coefficient is  .9997709
Magnetization in the interstitial region is  .0000000E+00
Total magnetization found in elementary cell is  .0000000E+00
***** FullPOT finished; CPU :  125.38      ; CUR/MAX mem.(Mb):  6.01      /  9.36

```

11.13 Calculating Energy Bands

After constructing the full potential the execution of the program goes to the package of program for solving the eigenvalue problem of the LMTO method. It is controlled by the program BANDS (see source file **bands.f**). Information about choice of Eny is printed below. Eny, Dny stand for the E_ν , D_ν values used for this particular l-channel, Cny is the estimated center of the l-band, Wny is its width. All the estimates are done using potential parameter relations.

```

***** Bands      started ; CPU :  125.38      ; CUR/MAX mem.(Mb):  4.03      /  9.36
2kappa spin-up panel # 1 : Band Structure Calculation of E(k) with :
      Eny:      Dny:      Cny:      Wny:      Et= -.100      for Ni1
      .65829    -1.00000    .65829    6.0679    for 4s-state, center
      2.1218    -2.0000    2.1218    2.8435    for 4p-state, center
      .47200    -3.0000    .47200    .42234    for 3d-state, center
      Eny:      Dny:      Cny:      Wny:      Et= -1.00      for Ni1
      .65829    -1.00000    .65829    6.0679    for 4s-state, center
      2.1218    -2.0000    2.1218    2.8435    for 4p-state, center
      .47200    -3.0000    .47200    .42234    for 3d-state, center
      Eny:      Dny:      Cny:      Wny:      Et= -.100      for Ni2
      1.1583    -1.0000    1.1583    6.0679    for 4s-state, center
      2.6218    -2.0000    2.6218    2.8435    for 4p-state, center
      .97200    -3.0000    .97200    .42234    for 3d-state, center
      Eny:      Dny:      Cny:      Wny:      Et= -1.00      for Ni2
      1.1583    -1.0000    1.1583    6.0679    for 4s-state, center
      2.6218    -2.0000    2.6218    2.8435    for 4p-state, center
      .97200    -3.0000    .97200    .42234    for 3d-state, center
      Eny:      Dny:      Cny:      Wny:      Et= -.100      for 0
      -1.0733   -1.0000    -1.0733    2.0253    for 2s-state, center

```

```

.39842      -2.0000      .39842      .71742      for 2p-state, center
  Eny:        Dny:        Cny:        Wny:        Et= -1.00      for 0
-1.0733     -1.0000     -1.0733     2.0253     for 2s-state, center
.39842      -2.0000      .39842      .71742     for 2p-state, center
2kappa spin-dn panel # 1 : Band Structure Calculation of E(k) with :
  Eny:        Dny:        Cny:        Wny:        Et= -.100      for Ni1
1.1583     -1.00000    1.1583     6.0679     for 4s-state, center
2.6218     -2.0000     2.6218     2.8435     for 4p-state, center
.97200     -3.0000     .97200     .42234     for 3d-state, center
  Eny:        Dny:        Cny:        Wny:        Et= -1.00      for Ni1
1.1583     -1.00000    1.1583     6.0679     for 4s-state, center
2.6218     -2.0000     2.6218     2.8435     for 4p-state, center
.97200     -3.0000     .97200     .42234     for 3d-state, center
  Eny:        Dny:        Cny:        Wny:        Et= -.100      for Ni2
.65829     -1.00000    .65829     6.0679     for 4s-state, center
2.1218     -2.0000     2.1218     2.8435     for 4p-state, center
.47200     -3.0000     .47200     .42234     for 3d-state, center
  Eny:        Dny:        Cny:        Wny:        Et= -1.00      for Ni2
.65829     -1.00000    .65829     6.0679     for 4s-state, center
2.1218     -2.0000     2.1218     2.8435     for 4p-state, center
.47200     -3.0000     .47200     .42234     for 3d-state, center
  Eny:        Dny:        Cny:        Wny:        Et= -.100      for 0
-1.0733     -1.0000     -1.0733     2.0253     for 2s-state, center
.39842     -2.0000     .39842     .71742     for 2p-state, center
  Eny:        Dny:        Cny:        Wny:        Et= -1.00      for 0
-1.0733     -1.0000     -1.0733     2.0253     for 2s-state, center
.39842     -2.0000     .39842     .71742     for 2p-state, center
***** Bands finished; CPU : 181.13 ; CUR/MAX mem.(Mb): 4.37 / 9.36

```

11.14 Brillouin Zone Integrals

Finding the Fermi level and weights for integrating over the Brillouin zone is done by **bzint.f**

```

***** BZint started ; CPU : 181.14 ; CUR/MAX mem.(Mb): 6.36 / 9.36
EF,TOS,DOS== .0000000E+00 4.000000 .0000000E+00
EF,TOS,DOS== .8000000 26.00000 .0000000E+00
EF,TOS,DOS== 1.600000 42.37689 66.42424
EF,TOS,DOS== 1.284228 38.11820 2.898284
EF,TOS,DOS== 1.140727 36.36411 5.198598
EF,TOS,DOS== .9304977 27.41325 64.96339
EF,TOS,DOS== .9979959 32.70743 48.18848
EF,TOS,DOS== .9849082 31.46928 92.54797
EF,TOS,DOS== .9905879 32.36942 43.94401
EF,TOS,DOS== .9872609 31.98831 370.8193
EF,TOS,DOS== .9872927 31.99996 363.5218

```

```

EF,TOS,DOS== .9872928      32.00000      363.4976
EF,TOS,DOS== .9872928      32.00000      363.4976
  TUP,DUP==    16.00000      181.7488
  TDN,DDN==    16.00000      181.7488
MM=TUP-TDN==          -0.1936229E-11
Calculated average square of electron velocities:
<Vx^2>= .3803598      ; <Vy^2>= .3803598      ; <Vz^2>= .3803598
<Ux^2>= .1901799      ; <Uy^2>= .1901799      ; <Uz^2>= .1901799
<Dx^2>= .1901799      ; <Dy^2>= .1901799      ; <Dz^2>= .1901799
Calculated bare plasma frequencies (in eV):
om_p^x= 2.666093      ; om_p^y= 2.666093      ; om_p^z= 2.666093
om_u^x= 1.885212      ; om_u^y= 1.885212      ; om_u^z= 1.885212
om_d^x= 1.885212      ; om_d^y= 1.885212      ; om_d^z= 1.885212
# of fully filled bands = 15; input # = 13
# of bands crossing Ef = 2; input # = 5
Energy bands at the Gamma-point for spin-up states are
-.74305      -.65657      .16735      .30967      .30967
.47174      .47174      .47905      .47905      .47907
.49391      .59754      .59754      .93269      .94406
.94438      .94438      1.0128      1.0128      1.2738
1.5068      1.8952      1.8952      2.3293      2.3376
2.3376      2.4769      2.5123      2.5123      2.5159
2.8389      2.8389      2.9061      2.9061      3.3645
3.6257      3.6257      3.6334      3.7516      4.4229
4.4229      4.9384      5.1360      5.1360      5.4295
5.4295      5.4799      6.9461      7.0788      7.7043
7.7048      7.7048
Energy bands at the Gamma-point for spin-dn states are
-.74305      -.65657      .16735      .30967      .30967
.47174      .47174      .47905      .47905      .47907
.49391      .59754      .59754      .93269      .94406
.94438      .94438      1.0128      1.0128      1.2738
1.5068      1.8952      1.8952      2.3293      2.3376
2.3376      2.4769      2.5123      2.5123      2.5159
2.8389      2.8389      2.9061      2.9061      3.3645
3.6257      3.6257      3.6334      3.7516      4.4229
4.4229      4.9384      5.1360      5.1360      5.4295
5.4295      5.4799      6.9461      7.0788      7.7043
7.7048      7.7048
LR information: # of fully filled bands (nff) = 8
LR information: # of bands crossing EF (nef) = 13
***** BZint finished; CPU : 202.76 ; CUR/MAX mem.(Mb): 6.28 / 9.36

```

11.15 Constructing Charge Density

When the charge density is calculated (program RHOFUL, see source file **rhoful.f**), the following output lines allow to check for the correct normalization. If it is more the a few per cent (more in the ASA), something is going wrong. If overlap matrix is not positive define or the ghos bands occur, the renormalization coefficient can strongly deviate from unity. Watch out then for the mistakes in the INIFILE.

```
***** FullRHO started ; CPU : 202.76 ; CUR/MAX mem.(Mb): 6.28 / 9.36
Valence charge in whole elementary cell must be 32.00000
Valence charge found via fourier transform is 32.07829
Renormalization coefficient of the val.density is .9975595
***** FullRHO finished; CPU : 213.15 ; CUR/MAX mem.(Mb): 6.08 / 9.36
```

11.16 Renormalizing Core Levels

The renormalization of the deep core levels (program RENCOR, see source file **rencor.f**) for each atom results in the following output table:

```
***** RenCOR started ; CPU : 213.15 ; CUR/MAX mem.(Mb): 6.08 / 9.36
Orbital n l j # el. Levels(Ry) for Ni1 ; Zcor= 18.000
1s1/2 1 0 1/2 2 -600.7811 ; deepcore
2s1/2 2 0 1/2 2 -70.57977 ; deepcore
2p1/2 2 1 1/2 2 -61.46961 ; deepcore
2p3/2 2 1 3/2 4 -60.18755 ; deepcore
3s1/2 3 0 1/2 2 -6.713606 ; deepcore
3p1/2 3 1 1/2 2 -3.942613 ; deepcore
3p3/2 3 1 3/2 4 -3.779989 ; deepcore
3d3/2 3 2 3/2 4 ; valence
3d5/2 3 2 5/2 4 ; valence
4s1/2 4 0 1/2 2 ; valence
Orbital n l j # el. Levels(Ry) for Ni2 ; Zcor= 18.000
1s1/2 1 0 1/2 2 -600.7811 ; deepcore
2s1/2 2 0 1/2 2 -70.57977 ; deepcore
2p1/2 2 1 1/2 2 -61.46961 ; deepcore
2p3/2 2 1 3/2 4 -60.18755 ; deepcore
3s1/2 3 0 1/2 2 -6.713606 ; deepcore
3p1/2 3 1 1/2 2 -3.942613 ; deepcore
3p3/2 3 1 3/2 4 -3.779989 ; deepcore
3d3/2 3 2 3/2 4 ; valence
3d5/2 3 2 5/2 4 ; valence
4s1/2 4 0 1/2 2 ; valence
Orbital n l j # el. Levels(Ry) for O ; Zcor= 2.0000
1s1/2 1 0 1/2 2 -36.42931 ; deepcore
2s1/2 2 0 1/2 2 ; valence
```

```

2p1/2  2  1  1/2  2          ; valence
2p3/2  2  1  3/2  2          ; valence
**** RenCOR finished; CPU : 213.36 ; CUR/MAX mem.(Mb): 6.08 / 9.36

```

11.17 Evaluating Total Energy

The work of the program evaluating the total energy (program ENERGY, see source file **energy.f**) starts by printing the partial numbers and densities of states for every atom and for every energy panel. Different contributions to the total energy are printed afterwards.

```

**** Energy started ; CPU : 213.40 ; CUR/MAX mem.(Mb): 4.11 / 9.36
Occupation numbers for the 1th panel; <E>= .3838274 ---->
Summed over tail energies partial states for Ni1
spdf... TOS = .2128 .2243 7.705 .3347E-01
spdf... MAG = .5338E-01 .3487E-01 1.973 .1598E-03
spdf up-TOS = .1331 .1296 4.839 .1681E-01
spdf dn-TOS = .7973E-01 .9474E-01 2.866 .1665E-01
spdf up-DOS = 1.537 .6765E-01 .6517E-01 .1563E-01
spdf dn-DOS = .2755 .4877E-01 168.3 .3735E-03
Summed over tail energies partial states for Ni2
spdf... TOS = .2128 .2243 7.705 .3347E-01
spdf... MAG = -.5338E-01 -.3487E-01 -1.973 -.1598E-03
spdf up-TOS = .7973E-01 .9474E-01 2.866 .1665E-01
spdf dn-TOS = .1331 .1296 4.839 .1681E-01
spdf up-DOS = .2755 .4877E-01 168.3 .3735E-03
spdf dn-DOS = 1.537 .6765E-01 .6517E-01 .1563E-01
Summed over tail energies partial states for O
spdf... TOS = 1.739 4.442 .2247E-01 .5349E-02
spdf... MAG = -.3553E-14 -.3109E-14 -.4406E-15 -.6388E-15
spdf up-TOS = .8696 2.221 .1124E-01 .2675E-02
spdf dn-TOS = .8696 2.221 .1124E-01 .2675E-02
spdf up-DOS = .5647 1.467 .4290E-01 .6397E-01
spdf dn-DOS = .5647 1.467 .4290E-01 .6397E-01
Different contributions for the orbitals with many kappas --->
Diagonal/cross spin-up occupations for Ni1
State Eny Value(Ry) DTOS CTOS :
4s center .65829 .1331104 .0000000E+00
4p center 2.1218 .1296099 .0000000E+00
3d center .47200 4.839174 .0000000E+00
Diagonal/cross spin-dn occupations for Ni1
State Eny Value(Ry) DTOS CTOS :
4s center 1.1583 .7973114E-01 .0000000E+00
4p center 2.6218 .9473707E-01 .0000000E+00
3d center .97200 2.865867 .0000000E+00
Diagonal/cross spin-up occupations for Ni2

```

```

State Eny      Value(Ry)      DTOS      CTOS :
4s center    1.1583      .7973114E-01 .0000000E+00
4p center    2.6218      .9473707E-01 .0000000E+00
3d center    .97200      2.865867      .0000000E+00
Diagonal/cross spin-dn occupations for Ni2
State Eny      Value(Ry)      DTOS      CTOS :
4s center    .65829      .1331104      .0000000E+00
4p center    2.1218      .1296099      .0000000E+00
3d center    .47200      4.839174      .0000000E+00
Diagonal/cross spin-up occupations for O
State Eny      Value(Ry)      DTOS      CTOS :
2s center   -1.0733      .8696054      .0000000E+00
2p center    .39842      2.221096      .0000000E+00
Diagonal/cross spin-dn occupations for O
State Eny      Value(Ry)      DTOS      CTOS :
2s center   -1.0733      .8696054      .0000000E+00
2p center    .39842      2.221096      .0000000E+00
E{ptn} : MT,INT-parts are -10086.92      .6850260
E{cou} : M,I,Z0-parts are -12562.84      .9838957      -.4285961E-02
E{xc } : MT,INT-parts are -275.3543      -1.843034
*****
*      BAND ENERGY      = 12.282476055293      *
*      CORE ENERGY      =-3631.4045016318      *
*      POTENTIAL ENERGY=-10086.234978347      *
*      KINETIC ENERGY   = 6467.1129527701      *
*      COULOMB ENERGY   =-12561.859875033      *
*      EXCH-CORR ENERGY=-277.19736022872      *
*      HUBBARD ENERGY   = .000000000000000E+00      *
*      TOTAL ENERGY     =-6371.9442824916      *
*****

```

11.18 Evaluating Forces

The evaluation of forces (program FORCES, see source file **forces.f**) is the next step at the iteration. Note that the Hellmann-Feynmann forces are not accurate and large incomplete basis set corrections must be taken into account. (See the description of the input parameter *npfr* in the INIFILE).

```

* CALCULATED FORCES AT THE CENTERS OF ATOMS ----->
* Position --> ( .00000E+00, .00000E+00, .00000E+00) for Ni1
* HF FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* HF+CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* Position --> ( 1.0000 , 1.0000 , 1.0000 ) for Ni2
* HF FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00

```

```

* HF+CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* Position --> ( .50000 , .50000 , .50000 ) for 0
* HF FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* HF+CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* Position --> ( 1.5000 , 1.5000 , 1.5000 ) for 0
* HF FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* HF+CR FORCE: Fx= .0000000E+00; Fy= .0000000E+00; Fz= .0000000E+00
* CALCULATED SUM RULES FOR THESE FORCES ----->
* HF+CR SUMRL: Sx= .0000000E+00; Sy= .0000000E+00; Sz= .0000000E+00
***** Energy finished; CPU : 246.59 ; CUR/MAX mem.(Mb): 4.11 / 9.43

```

11.19 Mixing Charge Densities

The last step at the iteration is mixing of the input charge density and the output charge density to prepare the input density to the next iteration. The self-consistency of the charge density can be simply watched by comparing the input/output charges of the spheres at the iteration (Values S(inp) and S(out)). The same is done for magnetization, M(inp) and M(out). Values I(inp) and I(out) stand for the interstitial charges. After Broyden mixing procedure the charge density and magnetization are constructed, the corresponding charges and magnetic moments within spheres are also printed out. A useful parameter for watching whether the Broyden mixing is properly working is the iteration weight. At the beginning this number is set to one. If the charge density with every iteration approaches the self consistency, the iteration weight is bigger than 1 and increased (can be up to infinity). If the iteration gives a bad guess for the charge density, the iteration weight becomes smaller than 1. If a few (say 10) consequent iterations gives the weight much smaller than 1 (of the order 0.1), than it is advised to switch off the Broyden mixing and use linear mixing instead.

```

Input/output charge transfer at the iteration --->
  S(inp)          S(out)
-1.315963        -1.832755    for Ni1
-1.315963        -1.832755    for Ni2
-.7359690        .1959939     for 0
-.7359690        .1959939     for 0
Input/output magnetic moment at the iteration --->
  M(inp)          M(out)
.0000000E+00     2.056759     for Ni1
.0000000E+00     -2.056759     for Ni2
.0000000E+00     -.2220446E-13  for 0
.0000000E+00     -.1865175E-13  for 0
  I(inp)          I(out)
4.103864         3.273523
4.103864         3.273523
*****
Broyden Mixing for Rho(r), Iter-Weight= 1.000000

```

Charge: Magnetization:
-1.471001 .6170278 for Ni1
-1.471001 -.6170278 for Ni2
-.4563801 -.5773160E-14 for O
-.4563801 -.4884981E-14 for O
Interst. Charge after Broyden = -3.854762
Magnetization over MT-spheres = -.5595524E-12

Charge Density Self-Consistency= 1.577652 .7890674 1
Magnetization Self-Consistency= 1.914647 .1692967E-12

Maximum memory allocated (Mbyte) == 9.425797
Peak memory reached in <MULTFTR> Subroutine

12 ADDITIONAL INPUT: KOVFILE

If it is necessary to build your own rotational system, the file describing rotational operations (KOVFILE) must be created. Two examples of this file for cubic and hexagonal groups are given below. Note that both cubic and hexagonal symmetries are understandable automatically if the symmetry code is set to either *C* or *H* (see the description of the symmetry code in the chapter describing STRFILE) If symmetry code is absent in STRFILE, the LMTART checks both C and H systems, and chooses the one with the largest number of operations present. If the symmetry code is set to *A* (arbitrary), a specific group must be described in the KOVFILE which will be read by the LMTART. Two examples for C and H systems are given below. You can make your own KOVFILE using these examples.

KOVFILE for describing the *cubic* rotational system is listed below. All the expressions must be recognizable by the CALC program (see the description of the calculator in the chapter describing STRFILE) Cubic rotations are given after Kovalev. Note the difference between the present and Kovalev numbering which is given by the second column. Notations are the following:

- U - equivalent operation
- A - arbitrary rotation along (X,Y,Z) (except Z-axis)
- Z - rotation along Z
- I - inversion
- C - combination

```
48      ! total # of operations in the Cubic group
U       ! operation: equivalent
1
A       ! operation: rotations along arbitrary axe
20      ! element #
pi/2
+1, 0, 0
A       ! operation
2
pi
+1, 0, 0
A       ! operation
19
3*pi/2
+1, 0, 0
A       ! operation
22
pi/2
0,+1, 0
A       ! operation
```

```

3
pi
0,+1, 0
A      ! operation
24
3*pi/2
0,+1, 0
Z      ! operation, rotations along Z-axe
15
pi/2
Z      ! operation
4
pi
Z      ! operation
14
3*pi/2
A      ! operation
5
2*pi/3
+1,+1,+1
A      ! operation
9
4*pi/3
+1,+1,+1
A      ! operation
10
2*pi/3
+1,-1,+1
A      ! operation
8
4*pi/3
+1,-1,+1
A      ! operation
6
2*pi/3
-1,-1,+1
A      ! operation
11
4*pi/3
-1,-1,+1
A      ! operation
12
2*pi/3
-1,+1,+1
A      ! operation

```

```

7
4*pi/3
-1,+1,+1
A      ! operation
16
pi
+1,+1, 0
A      ! operation
13
pi
+1,-1, 0
A      ! operation
18
pi
 0,+1,+1
A      ! operation
17
pi
 0,-1,+1
A      ! operation
23
pi
+1, 0,+1
A      ! operation
21
pi
-1, 0,+1
I      ! operation: inversion
25
C      ! operation: combinations
26
25 2
C      ! operation
27
25 3
C      ! operation
28
25 4
C      ! operation
29
25 5
C      ! operation
30
25 6
C      ! operation

```

31
25 7
C ! operation
32
25 8
C ! operation
33
25 9
C ! operation
34
25 10
C ! operation
35
25 11
C ! operation
36
25 12
C ! operation
37
25 13
C ! operation
38
25 14
C ! operation
39
25 15
C ! operation
40
25 16
C ! operation
41
25 17
C ! operation
42
25 18
C ! operation
43
25 19
C ! operation
44
25 20
C ! operation
45
25 21
C ! operation

```

46
25 22
C      ! operation
47
25 23
C      ! operation
48
25 24

```

KOVFILE for describing the *hexagonal* rotational system is listed below. All the expressions must be recognizable by the CALC facility (see the description of the calculator in the chapter describing STRFILE). Hexagonal rotations are given after Kovalev. Note the difference between the present and Kovalev numbering which is given by a second column. Notations are the following:

- U - equivalent operation
- A - arbitrary rotation along (X,Y,Z) (except Z-axe)
- Z - rotation along Z
- I - inversion
- C - combination

```

24      ! total # of elements in the Hexagonal group
U      ! operation: equivalent
1
Z      ! operation: rotations along Z
2      ! element #
pi/3
Z      ! operation
3
2*pi/3
Z      ! operation
4
pi
Z      ! operation
5
4*pi/3
Z      ! operation
6
5*pi/3
A      ! operation
10
pi
+1,0,0

```

```

A      ! operation: rotations along arbitrary axe
9
pi
+sqrt(3)/2,+1/2,0
A      ! operation
8
pi
+1/2,+sqrt(3)/2,0
A      ! operation
7
pi
0,+1,0
A      ! operation
12
pi
-1/2,+sqrt(3)/2,0
A      ! operation
11
pi
-sqrt(3)/2,+1/2,0
I      ! operation: inversion
13
C      ! operation: combinations
14
13 2
C      ! operation
15
13 3
C      ! operation
16
13 4
C      ! operation
17
13 5
C      ! operation
18
13 6
C      ! operation
19
13 7
C      ! operation
20
13 8
C      ! operation
21

```

13 9
C ! operation
22
13 10
C ! operation
23
13 11
C ! operation
24
13 12

13 DRAWING THE BANDS: FATFILE

When RUNMODE is set to **fat**, the energy bands and partial characters should be computed along the directions in the Brillouin zone. Before such calculation is performed FATFILE containing (optionally) number of k-points and (optionally) list of directions should be created.

An example of this file for NiO system is given below.

```
<FILE=FATFILE, INPUT=MODERN>
*****
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
nDivDir = 20                                   !
```

13.1 <SECTION=CTRL> ! Control Parameters:

No case sensitivity is assumed in the input parameters described below.

```
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
nDivDir = 20                                   !
```

- *nDivDir* (*default = 20*): # of divisions to set-up the k-grid along high-symmetry directions. List of high-symmetry lines can be specified in the STRFILE. For standard structures like BCC or FCC there is a default list of high-symmetry lines located in the files */atomdat/str.**.

14 COMPUTING DOS: DOSFILE

When RUNMODE is set to **dos**, the density of states and partial contributions should be computed. Before such calculation is performed DOSFILE containing (optionally) energy interval should be created

An example of this file for NiO system is given below.

```
<FILE=DOSFILE, INPUT=MODERN>
*****
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
EminDos = 0.0                                  !
EmaxDos = 1.0                                  !
nEnrDos = 100                                  !
nDiv(:) = 12 12 12                             !
```

14.1 <SECTION=CTRL> ! Control Parameters:

No case sensitivity is assumed in the input parameters described below.

```
<SECTION=CTRL>                                ! CONTROL PARAMETERS:
EminDos = 0.0                                  !
EmaxDos = 1.0                                  !
nEnrDos = 100                                  !
nDiv(:) = 12 12 12                             !
```

- *EminDos* - lower limit (in Ry) for DOS computation
- *EmaxDos* - upper limit (in Ry) for DOS computation
- *nEnrDos* - # of energy points to divide interval between *EminDos* and *EmaxDos*.
- *nDiv(:)* - # of divisions in k-space to be used for DOS calculations. This set will suppress the corresponding setting in INIFILE.

15 FINDING OPTICS: OPTFILE

When RUNMODE is set to **opt**, the optical properties should be computed. Before such calculation is performed OPTFILE containing (optionally) frequency interval should be created

An example of this file for NiO system is given below.

```
<FILE=OPTFILE, INPUT=MODERN>
*****
<SECTION=CTRL> ! CONTROL PARAMETERS:
WminOpt = 0.0 !
WmaxOpt = 1.0 !
nOmgOpt = 100 !
nDiv(:) = 12 12 12 !
```

15.1 <SECTION=CTRL> ! Control Parameters:

No case sensitivity is assumed in the input parameters described below.

```
<SECTION=CTRL> ! CONTROL PARAMETERS:
WminOpt = 0.0 !
WmaxOpt = 1.0 !
nOmgOpt = 100 !
nDiv(:) = 12 12 12 !
```

- *WminDos* - lower limit (in Ry) for OPT computation
- *WmaxDos* - upper limit (in Ry) for OPT computation
- *nOmgOpt* - # of energy points to divide interval between *WminDos* and *WmaxDos*.
- *nDiv(:)* - # of divisions in k-space to be used for OPT calculations. This set will suppress the corresponding setting in INIFILE.

16 CORE MEMORY

Memory in LMTART is dynamically allocated. It depends on the current set-up. In general, the storage is taken by several arrays. The structure constants (complex*16) are allocated as $(LmaxT+2)**2 * Natom * (LmaxB+1)**2 * Natom * Nkap$. The charge density and the potential (complex*16) are both allocated as $(Nrad+1) * Nsym * Natom * Nspin$. Here, $Nrad$ is the number of radial points (of the order 300-400), $Nsym$ is $\leq (LmaxT+2)**2$. The energy bands (real*8) are stored in the array of the size $Ndim * Nspin * Kmax * Npan$, where $Ndim$ is the dimension of the LMTO Hamiltonian, and $Kmax$ is the number of k-points. The size of the Hamiltonian and the overlap matrices (both complex*16) is $(Ndim * Nspin)**2$

To estimate the needed memory in bytes, multiply size of the complex array by 16, and real array by 8.

17 ERROR MESSAGES

Generally, two kind of errors exist in the program: warning messages (when the program does not terminate) and the error messages (when the program terminates). Normally warning messages mean that the program can either correct the problem itself or the problem is not important for the execution. The error messages always mean that the program can give a wrong result if the input files will not be corrected.

17.1 Errors connected with input

Some input data can be easily checked (like the number of atoms which is read from different input files). If there is a mismatch in the input, a corresponding message is printed and execution is terminated.

17.2 Errors connected with iterative procedures

A number of iterational procedures is programmed inside the LMTART package in order to find the Fermi energy or the E_ν values from a fixed set of D_ν . Limiting number of iterations and the accuracy is set in the file `/lmtart/run/setup.f`. If the number of iterations is exceeded here, the message is printed and execution is terminated.

17.3 Other errors

Some warnings and errors are connected with the lost of accuracy in solving differential equations or in iterational procedures. Another type of errors can be due to not positively defined overlap matrix which is most likely due to an error in the input. The overlap matrix defined with the non-overlapping MT-spheres is *always* positively defined. When MT-spheres overlap, there is a warning message.

18 VERSIONS DIFFERENCES

This section traces differences between current and previous versions of the LMTO programs.

- *ASA1.10,CEL0.30,PLW2.01 or earlier*: not important.
- *ASA1.20,CEL0.41,PLW2.10*: Broyden mixing is added.
- *ASA1.30,CEL3.50,PLW2.20*: LDA+U is added.
- *ASA1.40,CEL3.62,PLW2.30*: Spin-orbit coupling is added.
- *ASA1.42,CEL3.62,PLW2.32*: Relativistic solution of the semicore problem is rewritten, no references to the Libermann atomic program exist anymore.
- *ASA1.50,CEL3.70,PLW2.40*: Includes the possibility to calculate hopping integrals for tight-binding calculations. Does not actually work well.
- *ASA1.51,CEL3.71,PLW2.41*: Marginal internal changes.
- *ASA1.52,CEL3.72,PLW2.42*: Contains LRWF key connected with the adjustment of the radial wave functions. This substitutes previously used key NOVR in the INIFILE.

A bug in calculating exchange-correlation energy for IXC=4 (Vosko et.al parametrization of LSDA functional) has been found. It is not expected to influence much on the calculated results. The bug is corrected in following versions.

ASA1.60,CEL3.80,PLW2.50: Generalized gradient corrections after Perdew et.al added and tested. Two forms are available here: GGA91 and GGA96.

- LMTART 5.x: combines ASA and PLW packages, CEL package has been removed. Fully dynamical memory scheme based on FORTRAN90 allocatable arrays, input files are all essentially reelaborated. Default input data simplify the number of parameters to be described in the input. In the simplest case, only atomic charges and crystal structure must be given.
- LMTART 6.x: the following features added: (a) Finite temperatures (b) Full three dimensional treatment of magnetization in the relativistic calculations including LDA+U (c) Non-collinear magnetism (d) Tight-binding regime (e) Hoppings integrals extraction regime. Also, gaussian broadening k-space integration scheme is inserted (f) Optical properties added.

19 Acknowledgements

I greatly acknowledge Dr. Andrej Postnikov who has initiated writing of this manual. Part of the developments has been done in collaboration with my brother Dr. Dmitrij Savrasov. Special thanks to Prof. Ole Andersen and Dr. Ove Jepsen who are my LMTO teachers.

The development of the LmtART has been started in P.N. Lebedev Physical Institute, Moscow based on computer programs for electronic structure calculations written by E. G. Maksimov, I. I. Mazin and Yu. A. Uspenski. The LmtART has been further extensively developed in Max-Planck Institute, Stuttgart in the Department of Prof. O. K. Andersen. The most recent add-ons are from the developments done in Department of Physics, Rutgers University, Piscataway, NJ 08854.

20 COPYRIGHT

These programs are a free software for scientific and/or educational purposes.

It is not allowed to redistribute them without prior written consent of the Copyright owners.

It is illegal to commercially distribute these programs as a whole or incorporate any part of it into a commercial product.

References

- [1] O. K. Andersen, Phys. Rev. B **13**, 3050 (1975).
- [2] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).
- [3] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).
- [4] For a review, see, also , *Theory of the Inhomogeneous Electron Gas*, edited by S. Lundqvist and S. H. March (Plenum, New York, 1983).
- [5] S. Y. Savrasov and D. Y. Savrasov, Phys. Rev. B. **46**, 12181 (1992).
- [6] S. Y. Savrasov, Phys. Rev. B **54**, 16470 (1996).
- [7] For a review and complete set of references see, e.g., V. Anisimov, F. Aryasetiawan, and A. I. Lichtenstein, J. Phys.: Condens. Matter **9**, 767 (1997).
- [8] M. S. Hybertsen, M. Schlüter, and N. E. Christensen, Phys. Rev. B **39**, 9028 (1989).