

```

*****
*
* Section 2 - Input Description *
*
*****

```

This section of the manual describes the input to GAMESS. The section is written in a reference, rather than tutorial fashion. However, there are frequent reminders that more information can be found on a particular input group, or type of calculation, in the 'Further Information' section of this manual. There are also a number of examples shown in the 'Input Examples' section.

The order of this section is chosen to approximate the order in which most people prepare their input (\$CONTRL, \$BASIS/\$DATA, \$GUESS, and so on). After that comes run type related input, then properties input, input for two different solvation models, integral related input, and finally CI/MCSCF input. The next page contains a list of all possible input groups, in the order in which they can be found in this section.

name	function	module:routine
----	-----	-----
Molecule, basis, wavefunction specification:		
\$CONTRL	chemical control data	INPUTA:START
\$SYSTEM	computer related control data	INPUTA:START
\$BASIS	basis set	INPUTB:BASISS
\$DATA	molecule, basis set	INPUTB:MOLE
\$ZMAT	coded z-matrix	ZMATRX:ZMATIN
\$LIBE	linear bend data	ZMATRX:LIBE
\$SCF	HF-SCF wavefunction control	SCFLIB:SCFIN
\$SCFMI	SCF-MI input control data	SCFMI :MIINP
\$MP2	2nd order Moller-Plesset	MP2 :MP2INP
\$GUESS	initial orbital selection	GUESS :GUESMO
\$VEC	orbitals (formatted)	GUESS :READMO
Potential energy surface options:		
\$STATPT	geometry search control	STATPT:SETSIG
\$TRUDGE	nongradient optimization	TRUDGE:TRUINP
\$TRURST	restart data for TRUDGE	TRUDGE:TRUDGX
\$FORCE	hessian, normal coordinates	HESS :HESSX
\$CPHF	coupled-Hartree-Fock options	CPHF :CPINP
\$HESS	force constant matrix (formatted)	HESS :FCMIN
\$GRAD	gradient vector (formatted)	HESS :EGIN
\$DIPDR	dipole deriv. matrix (formatted)	HESS :DDMIN
\$VIB	HESSIAN restart data (formatted)	HESS :HSSNUM
\$MASS	isotope selection	VIBANL:RAMS
\$IRC	intrinsic reaction path	RXNCRD:IRCX

\$DRC	dynamic reaction path	DRC	:DRCDRV
\$SURF	potential surface scan	SURF	:SRFINP

Interpretation, properties:

\$LOCAL	orbital localization control	LOCAL	:LMOINP
\$TWOEI	J,K integrals (formatted)	LOCCD	:TWEIIN
\$ELMOM	electrostatic moments	PRPLIB:	INPELM
\$ELPOT	electrostatic potential	PRPLIB:	INPELP
\$ELDENS	electron density	PRPLIB:	INPELD
\$ELFLDG	electric field/gradient	PRPLIB:	INPELF
\$POINTS	property calculation points	PRPLIB:	INPPGS
\$GRID	property calculation mesh	PRPLIB:	INPPGS
\$PDC	MEP fitting mesh	PRPLIB:	INPPDC
\$MOLGRF	orbital plots	PARLEY:	PLTMEM
\$STONE	distributed multipole analysis	PRRPOP:	STNRD
\$MOROKM	Morokuma energy decomposition	MOROKM:	MOROIN
\$FFCALC	finite field polarizabilities	FFIELD:	FFLDX
\$TDHF	time dependent HF NLO properties	TDHF	:TDHFX

Solvation models:

\$EFRAG	effective fragment potentials	EFINP	:EFINP
\$FRAGNAME	specific named fragment pot.	EFINP	:RDSTFR
\$FRGRPL	inter-fragment repulsion	EFINP	:RDDFRL
\$PCM	polarizable continuum model	PCM	:PCMINP
\$PCMCAV	PCM cavity generation	PCM	:MAKCAV
\$NEWCAV	PCM escaped charge cavity	PCM	:DISREP
\$DISBS	PCM dispersion basis set	PCMDIS:	ENLBS
\$DISREP	PCM dispersion/repulsion	PCMVCH:	MORETS
\$SCRF	self consistent reaction field	SCRF	:ZRFINP

Integral and integral modification options:

\$ECP	effective core potentials	ECPLIB:	ECPPAR
\$EFIELD	external electric field	PRPLIB:	INPEF
\$INTGRL	format for 2e- integrals	INT2A	:INTIN
\$TRANS	integral transformation	TRANS	:TRFIN

MCSCF and CI wavefunctions, and their properties:

\$CIINP	control of CI process	GAMESS:	WFNCI
\$DET	determinantal full CI for MCSCF	ALDECI:	DETINP
\$CIDET	determinantal full CI	ALDECI:	DETINP
\$DRT	distinct row table for MCSCF	GUGDRT:	ORDORB
\$CIDRT	distinct row table for CI	GUGDRT:	ORDORB
\$MCSCF	parameters for MCSCF	MCSCF	:MCSCF
\$MCQDPT	multireference pert. theory	MCQDPT:	MQREAD
\$CISORT	integral sorting	GUGSRT:	GUGSRT
\$GUGEM	Hamiltonian matrix formation	GUGEM	:GUGAEM
\$GUGDIA	Hamiltonian eigenvalues/vectors	GUGDGA:	GUGADG
\$GUGDM	1e- density matrix	GUGDM	:GUGADM
\$GUGDM2	2e- density matrix	GUGDM2:	GUG2DM
\$LAGRAN	CI lagrangian matrix	LAGRAN:	CILGRN
\$TRFDM2	2e- density backtransformation	TRFDM2:	TRF2DM
\$TRANST	transition moments, spin-orbit	TRNSTN:	TRNSTX

\* this column is more useful to programmers than to users.

\$CONTRL group.....	4
\$SYSTEM group.....	13
\$D5 group.....	17
\$DFT group.....	17
\$BASIS group.....	20
\$DATA group.....	22
\$ZMAT group.....	30
\$LIBE group.....	33
\$SCF group.....	33
\$SCFMI group.....	36
\$MP2 group.....	37
\$MP2GRD group.....	41
\$MP3 group.....	43
\$MP4 group.....	43
\$GUESS group.....	44
\$VEC group.....	46
\$STATPT group.....	47
\$TRUDGE group.....	51
\$TRURST group.....	53
\$FORCE group.....	53
\$CPHF group.....	56
\$HESS group.....	56
\$GRAD group.....	57
\$DIPDR group.....	57
\$VIB group.....	57
\$MASS group.....	57
\$IRC group.....	58
\$GRADEX group.....	61
\$DRC group.....	63
\$SURF group.....	66
\$LOCAL group.....	68
\$TWOEI group.....	70
\$ELMOM group.....	71
\$ELPOT group.....	71
\$ELDENS group.....	72
\$ELFLDG group.....	73
\$POINTS group.....	73
\$GRID group.....	73
\$PDC group.....	74
\$MOLGRF group.....	76
\$STONE group.....	77
\$MOROKM group.....	79
\$FFCALC group.....	81
\$TDHF group.....	82

\$TDDFT group.....	86
\$TDVEC group.....	88
\$EFRAG group.....	89
\$FRAGNAME group.....	91
\$FRGRPL group.....	94
\$PCM group.....	95
\$PCMCAV group.....	98
\$NEWCAV group.....	99
\$DISBS group.....	100
\$DISREP group.....	100
\$SCRF group.....	101
\$ECP group.....	101
\$EFIELD group.....	104
\$INTGRL group.....	104
\$FMM group.....	106
\$TRANS group.....	108
\$nbo.....	109
\$CIINP group.....	110
\$CIS group.....	111
\$CISVEC group.....	113
\$DET group.....	114
\$CIDET group.....	114
\$DRT group.....	116
\$CIDRT group.....	116
\$MCSCF group.....	117
\$MCQDPT group.....	121
\$MCQFIT group.....	123
\$CISORT group.....	124
\$GUGDRT group.....	125
\$GUGEM group.....	125
\$GUGDIA group.....	126
\$GUGDM group.....	127
\$GUGDM2 group.....	128
\$LAGRAN group.....	129
\$TRFDM2 group.....	129
\$TRANST group.....	130

=====  
**\$CONTRL group** (optional)

This is a free format group specifying global switches.

SCFTYP	together with MPLEVL or CITYP specifies the wavefunction. You may choose from
= RHF	Restricted Hartree Fock calculation (default)

= UHF            Unrestricted Hartree Fock calculation  
 = ROHF           Restricted open shell Hartree-Fock.  
                   (high spin, see GVB for low spin)  
 = GVB            Generalized valence bond wavefunction  
                   or OCBSE type ROHF. (needs \$SCF input)  
 = MCSCF          Multiconfigurational SCF wavefunction  
                   (this requires \$DET or \$DRT input)  
 = NONE           indicates a single point computation,  
                   rereading a converged SCF function.  
                   This option requires that you select  
                   CITYP=GUGA or ALDET, RUNTYP=ENERGY,  
                   TRANSITN, or SPINORBT, and GUESS=MOREAD.

d5 = 0            Requests cartesian basis sets (default)  
      = 1            Allows the use of pure spherical basis sets. In this  
                   instance, \$d5 input group is read in and parsed.  
                   Note that the current implementation of d5 option is  
                   not compatible with non-standard molecular input  
                   frames (custom orientations of axes).

MPLEVL =           chooses Moller-Plesset perturbation  
                   theory level, after the SCF.  
                   See \$MP2 and \$MCQDPT input groups.  
      = 0            skips the MP computation (default)  
      = 2            performs a second order energy  
                   correction. MP2 is implemented only  
                   for RHF, UHF, ROHF, and MCSCF wave  
                   functions. Gradients are available  
                   only for RHF, so for the others you  
                   may pick from RUNTYP=ENERGY, TRUDGE,  
                   SURFACE, or FFIELD only.  
      = 3            performs a third order energy  
                   correction.  
      = 4            performs a fourth order energy  
                   correction.

CITYP =            chooses CI computation after the SCF,  
                   for any SCFTYP except UHF.  
      = NONE        skips the CI. (default)  
      = CIS           single excitations from a SCFTYP=RHF  
                   reference, only. This is for excited  
                   states, with analytic nuclear gradients  
                   available. See the \$CIS input group.

     = GUGA        runs the Unitary Group CI package,  
                   which requires \$CIDRT input.  
                   Gradients are available only for RHF,  
                   so for other SCFTYPs, you may choose  
                   only RUNTYP=ENERGY, TRUDGE, SURFACE,  
                   FFIELD, TRANSITN, or SPINORBT.

= ALDET runs the Ames Laboratory determinant full CI package, requiring \$CIDET input. RUNTYP=ENERGY only.

Obviously, at most one of MPLEVL or CITYP may be chosen.

RUNTYP specifies the type of computation, for example at a single geometry point:

= ENERGY Molecular energy. (default)  
= GRADIENT Molecular energy plus gradient.  
= HESSIAN Molecular energy plus gradient plus second derivatives, including harmonic

harmonic vibrational analysis. See the \$FORCE and \$CPHF input groups.

multiple geometry options:

= OPTIMIZE Optimize the molecular geometry using analytic energy gradients. See \$STATPT.  
= TRUDGE Non-gradient total energy minimization. See groups \$TRUDGE and \$TRURST.  
= SADPOINT Locate saddle point (transition state). See the \$STATPT group.  
= IRC Follow intrinsic reaction coordinate. See the \$IRC group.  
= GRADEXTR Trace gradient extremal. See the \$GRADEX group.  
= DRC Follow dynamic reaction coordinate. See the \$DRC group.  
= SURFACE Scan linear cross sections of the potential energy surface. See \$SURF.  
= RSURFACE Relaxed potential energy scans. See \$SURF

single geometry property options:

= PROP Properties will be calculated. A \$DATA deck and converged \$VEC group should be input. Optionally, orbital localization can be done. See \$ELPOT, etc.  
= MOROKUMA Performs monomer energy decomposition. See the \$MOROKM group.  
= TRANSITN Compute radiative transition moment. See the \$TRANST group.  
= SPINORBT Compute spin-orbit coupling. See the \$TRANST group.  
= FFIELD applies finite electric fields, most commonly to extract polarizabilities. See the \$FFCALC group.  
= TDHF analytic computation of time dependent polarizabilities. See the \$TDHF group.

\* \* \* \* \*

Note that RUNTYPs involving the energy gradient, which are GRADIENT, HESSIAN, OPTIMIZE, SADPOINT, IRC, GRADEXTR, and DRC, cannot be used for any CI or MP2 computation, except when SCFTYP=RHF.  
 \* \* \* \* \*

EXETYP = RUN            Actually do the run. (default)  
          = CHECK        Wavefunction and energy will not be evaluated. This lets you speedily check input and memory requirements. See the overview section for details.  
          = DEBUG        Massive amounts of output are printed, useful only if you hate trees.  
          = QFMM         obtain timing statistics of the various QFMM stages during SCF.  
          = routine      Maximum output is generated by the routine named. Check the source for the routines this applies to.

MAXIT =                Maximum number of SCF iteration cycles. Pertains only to RHF, UHF, ROHF, or GVB runs. See also MAXIT in \$MCSCF. (default = 30)

\* \* \* \* \*

ICHARG =                Molecular charge. (default=0, neutral)  
 MULT =                Multiplicity of the electronic state  
          = 1             singlet (default)  
          = 2,3,...       doublet, triplet, and so on.

ICHARG and MULT are used directly for RHF, UHF, ROHF. For GVB, these are implicit in the \$SCF input, while for MCSCF or CI, these are implicit in \$DRT/\$CIDRT or \$DET/\$CIDET input. You must still give them correctly.

\* \* \* \* \*

ECP =                effective core potential control.  
          = NONE        all electron calculation (default).  
          = READ        read the potentials in \$ECP group.  
          = SBKJC        use Stevens, Basch, Krauss, Jasien, Cundari potentials for all heavy atoms (Li-Rn are available).  
          = HW          use Hay, Wadt potentials for all the heavy atoms (Na-Xe are available).

\* \* \* \* \*

\* \* \* the next three control molecular geometry \* \* \*

COORD = choice for molecular geometry in \$DATA.  
= UNIQUE only the symmetry unique atoms will be given, in Cartesian coords (default).  
= HINT only the symmetry unique atoms will be given, in Hilderbrandt style internals.  
= CART Cartesian coordinates will be input. Please read the warning just below!!!  
= ZMT GAUSSIAN style internals will be input.  
= ZMTMPC MOPAC style internals will be input.  
= FRAGONLY means no part of the system is treated by ab initio means, hence \$DATA is not given. The system is specified by \$EFRAG.

Note that the CART, ZMT, ZMTMPC choices require input of all atoms in the molecule. These three also orient the molecule, and then determine which atoms are unique. The reorientation is very likely to change the order of the atoms from what you input. When the point group contains a 3-fold or higher rotation axis, the degenerate moments of inertia often cause problems choosing correct symmetry unique axes, in which case you must use COORD=UNIQUE rather than Z-matrices.

Warning: The reorientation into principal axes is done only for atomic coordinates, and is not applied to the axis dependent data in the following groups: \$VEC, \$HESS, \$GRAD, \$DIPDR, \$VIB, nor Cartesian coords of effective fragments in \$EFRAG. COORD=UNIQUE avoids reorientation, and thus is the safest way to read these.

Note that the choices CART, ZMT, ZMTMPC require the use of a \$BASIS group to define the basis set. The first two choices might or might not use \$BASIS, as you wish.

UNITS = distance units, any angles must be in degrees.  
= ANGS Angstroms (default)  
= BOHR Bohr atomic units

NZVAR = 0 Use Cartesian coordinates (default).  
= M If COORD=ZMT or ZMTMPC and a \$ZMAT is not given: the internal coordinates will be those defining the molecule in \$DATA. In this case, \$DATA must not contain any dummy atoms. M is usually 3N-6, or 3N-5 for linear.  
= M For other COORD choices, or if \$ZMAT is given: the internal coordinates will be those defined in \$ZMAT. This allows more sophisticated internal coordinate choices. M is ordinarily 3N-6 (3N-5), unless \$ZMAT has linear bends.

NZVAR refers mainly to the coordinates used by OPTIMIZE or SADPOINT runs, but may also print the internal's values for other run types. You can use internals to define the molecule, but Cartesians during optimizations!

LOCAL = controls orbital localization.  
= NONE Skip localization (default).



= BOYS Do Foster-Boys localization.  
 = RUEDNBRG Do Edmiston-Ruedenberg localization.  
 = POP Do Pipek-Mezey population localization.  
 See the \$LOCAL group. Localization  
 does not work for SCFTYP=GVB or CITYP.

DFTTYP = Functional

pure exchange functionals (no correlation)  
 LDA exchange:

= SLATER Slater exchange, no correlation  
 = LSDA the same as SLATER

GGA exchange:

= B88 Becke 1988 exchange, no correlation  
 = XPW91 Perdew-Wang 1991 exchange, no correlation  
 = GILL96 Gill 1996 exchange, no correlation  
 = XPBE96 Perdew-Burke-Ernzerhof 1996 exchange, no correlation  
 = OPTX Handy-Cohen 2001 OPTX exchange, no correlation

pure correlation functionals (Hartree-Fock exchange)

LDA correlation:

= VWN1RPA Hartree-Fock exchange, VWN formula 1 RPA LDA  
 correlation  
 = VWN5 Hartree-Fock exchange, VWN formula 5 LDA correlation  
 = PW91LDA Hartree-Fock exchange, Perdew-Wang 1991 LDA  
 correlation

GGA correlation:

= LYP Hartree-Fock exchange, Lee-Yang-Parr 1988 correlation  
 = CPBE96 Hartree-Fock exchange, Perdew-Burke-Ernzerhof 1996  
 nonlocal +Perdew-Wang 1991 LDA correlation  
 = CPW91 Hartree-Fock exchange, Perdew 1991 nonlocal +  
 Perdew-Wang 1991 LDA correlation

exchange-correlation functionals

= SLYP Slater exchange, Lee-Yang-Parr 1988 correlation  
 = BLYP Becke 1988 exchange, Lee-Yang-Parr 1988 correlation  
 = GLYP Gill 1996 exchange, Lee-Yang-Parr 1988 correlation  
 = XLYP Extended exchange functional (a combination of Slater,  
 Becke 88, and Perdew-Wang 91 exchange functionals) of  
 Xu and Goddard III, Lee-Yang-Parr 1988 correlation  
 = OLYP OPTX exchange, Lee-Yang-Parr 1988 correlation (  
 = SVWN1RPA Slater exchange, VWN formula 1 RPA correlation  
 = BVWN1RPA Becke 1988 exchange, VWN formula 1 RPA correlation  
 = SVWN5 Slater exchange, VWN formula 5 correlation  
 = BVWN5 Becke 1988 exchange, VWN formula 5 correlation  
 = PBE96 Perdew-Burke-Ernzerhof 1996 exchange,  
 Perdew-Burke-Ernzerhof 1996 nonlocal +  
 Perdew-Wang 1991 LDA correlation  
 = PBEPW91 Perdew-Burke-Ernzerhof 1996 exchange,  
 Perdew 1991 nonlocal + Perdew-Wang 1991 LDA  
 correlation  
 = PW91 Perdew-Wang 1991 exchange,  
 Perdew 1991 nonlocal + Perdew-Wang 1991 LDA  
 correlation

hybrid functionals

= B3LYP1 B3LYP as implemented in NWCHEM and GAUSSIAN 98,  
using VWN formula 1 RPA correlation

= B3LYP5 B3LYP as implemented in GAMESS (US),  
using VWN formula 5 correlation

= B3LYP either B3LYP1 or B3LYP5 depending on the value of  
B3LYP keyword in the \$DFT group, see later.

= X3LYP Extended exchange functional (a combination of Slater,  
Becke 88, and Perdew-Wang 91 exchange functionals) of  
Xu and Goddard III + Hartree-Fock exchange,

= O3LYP1 Slater + OPTX + Hartree-Fock exchange,  
VWN formula 1 RPA + Lee-Yang-Parr 1988 correlation

= O3LYP5 Slater + OPTX + Hartree-Fock exchange,  
VWN formula 5 + Lee-Yang-Parr 1988 correlation

= O3LYP The synonym of O3LYP5

NOTE: FOR ALL O3LYP FAMILY FUNCTIONALS,  
SEE THE DESCRIPTION OF THE O3LYP OPTION BELOW!

= BHLYP Becke 1988 + Hartree-Fock exchange, Lee-Yang-Parr 1988  
correlation

= PBE0 Perdew-Burke-Ernzerhof 1996 + Hartree-Fock exchange,  
Perdew-Burke-Ernzerhof 1996 nonlocal +  
Perdew-Wang 1991 LDA correlation

= PBE1PW91 Perdew-Burke-Ernzerhof 1996 + Hartree-Fock exchange,  
Perdew 1991 nonlocal + Perdew-Wang 1991 LDA  
correlation

= B3PW91 Slater + Becke 1988 + Hartree-Fock exchange,  
Perdew 1991 nonlocal + Perdew-Wang 1991 LDA  
correlation

\* \* \* interfaces to other programs \* \* \*

MOLPLT = flag that produces an input deck for a molecule  
drawing program distributed with GAMESS.  
(default is .FALSE.)

PLTORB = flag that produces an input deck for an orbital  
plotting program distributed with GAMESS.  
(default is .FALSE.)

AIMPAC = flag to create an input deck for Bader's atoms  
in molecules properties code. (default=.FALSE.)  
For information about this program, contact  
Richard F.W. Bader  
Dept. of Chemistry  
McMaster University  
Hamilton, Ontario L8S-4M1 Canada  
bader@sscvox.cis.mcmaster.ca

RPAC = flag to create the input files for Bouman and  
Hansen's RPAC electronic excitation and NMR  
shieldings program. RPAC works only with  
RHF wavefunctions. (inactive)

FRIEND = string to prepare input to other quantum programs, choose from  
= HONDO for HONDO 8.2  
= MELDF for MELDF  
= GAMESSUK for GAMESS (UK Daresbury version)  
= GAUSSIAN for Gaussian 9x  
= ALL for all of the above

PLTORB, MOLPLT, and AIMPAC decks are written to file PUNCH at the end of the job. The two binary disk files output by RPAC are written at the end of the job. Thus all of these correspond to the final geometry encountered during the job.

In contrast, selecting FRIEND turns the job into a CHECK run only, no matter how you set EXETYP. Thus the geometry is that encountered in \$DATA. The input is added to the PUNCH file, and may require some (usually minimal) massaging.

PLTORB and MOLPLT are written even for EXETYP=CHECK. AIMPAC requires at least RUNTYP=PROP. RPAC requires at least RUNTYP=ENERGY, and you must take action to save the binary files AOINTS and WORK15.

\* \* \* computation control switches \* \* \*

For the most part, the default is the only sensible value, and unless you are sure of what you are doing, these probably should not be touched.

NPRINT = Print/punch control flag  
See also EXETYP for debug info.  
(options -7 to 5 are primarily debug)  
= -7 Extra printing from Boys localization.  
= -6 debug for geometry searches  
= -5 minimal output  
= -4 print 2e-contribution to gradient.  
= -3 print 1e-contribution to gradient.  
= -2 normal printing, no punch file  
= 1 extra printing for basis, symmetry, ZMAT  
= 2 extra printing for MO guess routines  
= 3 print out property and 1e- integrals  
= 4 print out 2e- integrals  
= 5 print out SCF data for each cycle.  
(Fock and density matrices, current MOs same as 7, but wider 132 columns output. This option isn't perfect.  
= 6 normal printing and punching (default)  
= 7 more printout than 7. The extra output is (AO) Mulliken and overlap population analysis, eigenvalues, Lagrangians, ...  
= 8 everything in 8 plus Lowdin population analysis, final density matrix.

NOSYM = 0      the symmetry specified in \$DATA is used  
                 as much as possible in integrals, SCF,  
                 gradients, etc. (this is the default)  
= 1            the symmetry specified in the \$DATA group  
                 is used to build the molecule, then  
                 symmetry is not used again. Some GVB  
                 or MCSCF runs (those without a totally  
                 symmetric charge density) require you  
                 request no symmetry.

INTTYP = POPLE use fast Pople-Hehre routines for sp integral  
                 blocks, and HONDO Rys polynomial code for  
                 all other integrals. (default)  
= HONDO use HONDO/Rys integrals for all integrals.  
                 This option produces slightly more accurate  
                 integrals but is also slower.

When diffuse functions are used, the inaccuracy in Pople/Hehre sp integrals shows up as inaccurate LCAO coefficients in virtual orbitals. This means the error in SCF (meaning RHF to MCSCF) energies is expected to be about 5d-8 Hartree, but the error in computations that OCCUPY the virtual orbitals may be much larger. We have seen an energy error of 1d-4 in an MP2 energy when diffuse functions were used. **We recommend that all MP2 or CI jobs with diffuse functions select INTTYP=HONDO.**

FSTINT=.TRUE./.FALSE. Enables (default)/disables the use of the new direct SCF code. For use with MCSCF algorithms, either \$mcscf SOSCF=.t or FOCAS=.t. may be used, but fullnr method is not supported.

REORDR=.TRUE./.FALSE. Enables (default)/disables shells reordering for even better direct SCF performance.

GENCON=.TRUE./.FALSE. Enables (default)/disables the use of the special version of the fastints code designed for general contraction (GC) type basis sets. It is mainly intended to dramatically speedup calculations involving large GC-type basis sets like ANO basis sets by Roos et al (the example of pure GC basis sets), and to some degree cc-pVXZ basis sets (which are only partially of GC type), and many others. The code is very efficient, but requires some additional amount of memory, as well as has minor addition computational overhead for setup. It can result in slightly different energies than the standard fastints code using the same value of ICUT and ITOL parameters, and does not improve performance for pure segmented contraction basis sets at all. This is why the gencon code automatically disables itself if the basis set is not of GC type. It has no effect at present on QFMM calculations. For use with MCSCF algorithms, either \$mcscf SOSCF=.t or FOCAS=.t. may be used, but fullnr method is not supported.

NORMF = 0     normalize the basis functions (default)  
           = 1     no normalization

NORMP = 0     input contraction coefficients refer to  
                   normalized Gaussian primitives. (default)  
           = 1     the opposite.

ITOL    =     primitive cutoff factor (default=20)  
           = n     products of primitives whose exponential  
                   factor is less than 10\*\*(-n) are skipped.

ICUT    = n     integrals less than 10.0\*\*(-n) are not  
                   saved on disk. (default = 9)

WIDE = True.    Wide format for vectors in \$VEC groups for both INPUT and  
                   PUNCH files Default is .False. .  
           = False. Normal format (which is standard for GAMESS).

\* \* \* restart options \* \* \*

IREST =        restart control options  
                   (for OPTIMIZE run restarts, see \$STATPT)  
                   Note that this option is unreliable!  
           = -1    reuse dictionary file from previous run,  
                   useful with GEOM=DAF and/or GUESS=MOSAVED.  
                   Otherwise, this option is the same as 0.  
           = 0     normal run (default)  
           = 1     2e restart (1-e integrals and MOs saved)  
           = 2     SCF restart (1-,2-e integrals and MOs saved). *Now allows  
                   one to restart with the AOINTS file only, without  
                   DICTNRY. 1-e integrals will be recomputed if the old  
                   DICTNRY file does not exist.*  
           = 3     1e gradient restart  
           = 4     2e gradient restart

GEOM    =        select where to obtain molecular geometry  
           = INPUT from \$DATA input (default for IREST=0)  
           = DAF    read from DICTNRY file (default otherwise)

As noted in the first chapter, binary file restart is

not a well tested option!

## **\$SYSTEM group**                   (optional)

This group provides global control information for your computer's operation. This is system related input, and will not seem particularly chemical to you!

TIMLIM = time limit, in minutes. Set to about 95 percent of the time limit given to the batch job so that GAMESS can stop itself gently. (default=600.0)

MWORDS = the maximum replicated memory which your job can use, on every node. This is given in units of 1,000,000 words (as opposed to 1024\*1024 words), where a word is always a 64 bit quantity. Most systems allocate this memory at run time, but some more primitive systems may have an upper limit chosen at compile time. (default=1)  
In case finer control over the memory is needed, this value can be given in units of words by using the keyword MEMORY instead of MWORDS.

KDIAG = diagonalization control switch  
= 0 selects use of very stable and fast diagonalization routine which requires large amount of extra memory. (default)  
= -1 selects potentially less stable but even faster diagonalization routine which uses less memory than kdiag=0  
= -2 selects a combination of two above mentioned methods which can be more stable than kdiag=-1, is usually as fast as kdiag=-1 but requires as much memory as kdiag=0  
= 1 use EVVRSP diagonalization. This may be more accurate than Gamess (US) KDIAG=0.  
= 2 use GIVEIS diagonalization (not as fast or reliable as EVVRSP)  
= 3 use JACOBI diagonalization (this is the slowest method)

The Fastdiag dynamic library (fastdiag.dll in Windows PC GAMESS distribution and fastdiag.ex in Linux version) contains fast optimized modern algorithms for symmetric matrix diagonalization and inversion and is intended to improve the performance of initial guess generation, DIIS extrapolation, as well as some other computationally-intensive steps. You should put it into the same folder as the PC GAMESS executables (Windows users) and in your working/scratch directory (Linux users). It should be renamed to be all lower-case (Linux users).

Note that fast diagonalization routines uses extra memory which is not taken into account during check runs at present. Thus it is recommended to reserve some additional amount of memory for diagonalization routines. For example, if you have a job with 1000 basis functions, it is a good idea to add ca. 2.5-3MW of memory for diagonalization purposes. Otherwise, the slower built-in routine will be called if the amount of memory is not enough to use the fast routines.

Finally, it should be noted that fastdiag is not compatible with and is not used by the generic Pentium PC GAMESS version.

NOJAC = n never use Jacobi diagonalization for matrices of size n x n and above. Recommended for "large-scale" CI and MCSCF calculations (i.e. systems having large number of basis functions (e.g., 1500) and relatively small active spaces (e.g., 12 electrons/12 orbitals)).

COREFL = a flag to indicate whether or not GAMESS should produce a "core" file for debugging when subroutine ABRT is called to kill a job. This variable pertains only to

UNIX operating systems. (default=.FALSE.)

- FASTF = False. Normal operation .  
= True. Turns on the usage of fast (non-fortran) sequential file access routines. This increases the overall performance, and allows PC GAMESS to handle large (> 2Gb) sequential access files under Windows NT.
- SAFMEM The size of the safety memory pool (in double precision words), which is used by several parts of PC GAMESS in the case when there is not enough dynamic memory to perform current operation. In some cases, the PC GAMESS tries to use slightly more memory than the amount available. Then, increasing SAFMEM can help.
- LDAR The default value of LDAR parameter (the size of direct access file record) for \$INTGRL, \$GUGDM2, \$TRFDM2, \$CISORT, and \$MCSCF groups. Default is 2045
- TRUNCF  
= .False. The existing files of non-zero length are not truncated when being (re)opened for writing. This may be useful in some cases, for (almost) all the needed disk space is permanently (pre)allocated at the beginning of your PC GAMESS run, which prevents other disk-intensive programs from causing GAMESS to die in future if the free disk space is exhausted. It has an effect only if FASTF is set to True.  
= .True. The files are truncated while being (re)opened for writing. This is the default behavior, which results in the faster write operations. Default is .True.
- WSCTL  
= .False. Normal operation. The operating system controls the amount of physical memory (the size of the working set) used by the PC GAMESS.  
= .True. The PC GAMESS tries to allocate the working set as large as the amount of memory needed for the current operation. Roughly speaking, the process working set is the amount of physical (not virtual!) memory that is allotted to the process. Thus, this strategy usually considerably reduces paging in the case of memory-demanded jobs. On the other hand, this can leave less physical memory available to other processes and to the operating system itself. See also FAQ for information concerning the possible problems (and how to solve them). It has an effect when running under WinNT only. Default is .True.
- MAXWS The upper limit for the size of the PC GAMESS working set, in DP words. The PC GAMESS will never try to allocate working set that is larger than this amount. The value of 0 means the automatic selection of an appropriate limit. Namely, the limit will be set to the value smaller by 16 MB than the amount of the physical memory installed (or to the half of the physical memory, if the latter value is larger. It has an effect only when

running under WinNT, and WSCTL is set to True. Default is zero.

#### BLAS3

- = sleep Additional threads that were created by BLAS level 3 routines, are suspended when not in use. (default)
- = nosleep Additional threads are permanently active. Use this option only if you encounter some unexpected SMP-related problems with the default settings. It is supported only by SMP-enabled PC GAMESS version running in SMP environment.

#### DECOMM

- = .False. Old-style memory manager behavior. Use this option only if you encounter some unexpected problems with the default settings.
- = .True. Turns on the advanced memory management feature. The advanced memory management is enabled by default. It is supported under NT, Win95/98, and OS/2.

#### FLUSH

- = .False. Disables the usage of file cache flush operation. It is recommended to disable flushing only in the two following situations:
  - # if the total size of all PC GAMESS working files is less than the available size of the OS file cache;
  - # if the hardware RAID volume is used as the scratch storage.
- = .True. Enables the file cache flush operation. PC GAMESS will flush the cache buffers of its working files into disk, when necessary. File flushing is enabled by default. It is supported under NT, Win95/98, and OS/2.

L2SIZE The size of fast L2 cache per CPU, in Kbytes. This information is used by several parts of the PC GAMESS (e.g., by MP3/MP4 code) in order to select the computational strategy optimal for the specific hardware the PC GAMESS is running on. At present, the default value is 0 for all Pentium-optimized PC GAMESS versions, and 512 for Win32-specific Pentium Pro/II/III-optimized version. Change the defaults appropriately for your hardware configuration.

\* \* \* the next three refer to parallel GAMESS \* \* \*

The next three apply only to parallel runs, and as they are more or less obsolete, their use is discouraged.

BALTYP = Parallel load balance scheme  
LOOP turns off dynamic load balancing (DLB)  
NXTVAL uses dynamic load balancing  
(default = LOOP)

XDR = a flag to indicate whether or not messages should be converted into a generic format known as external data representation. If true, messages can exchange between machines of different vendors, at the cost



of performing the data type conversions.  
(default=.FALSE.) --inactive at present--

PTIME = a logical flag to print extra timing info  
during parallel runs. This is not currently  
implemented.

---

## **\$D5 group**

(optional)

d5=.true.(default)/.false.  
f7=.true.(default)/.false.  
g9=.true.(default)/.false.

The default values of d5, f7, and g9 keywords of the \$d5 group are all .true., i.e., to use pure spherical d, f, and g functions. The \$d5 group has no effect if d5 flag of the \$contrl group is not set, which is the default settings. Thus if you prefer to use cartesian basis set, you shouldn't do anything special. If you want to use pure spherical basis, set \$contrl d5=1 \$end. If you like to use Cartesian d functions but spherical f and g harmonics, you should specify:

```
$contrl d5=.t. $end  
$d5 d5=.f. $end
```

---

## **\$DFT group**

(optional)

NRAD (integer) the default number of radial points per atom used.  
Default value is 63.

NRDATM (integer array, dimension 128) An array to change the actual value of NRAD on per element number basis. For example, setting NRDATM(6)=128 will apply the radial grid of 128 points to all carbon atoms. The entry NRDATM(128) is used for dummy atoms with no charge. The default values are all equal to NRAD.

LMAX (integer) the default order of Lebedev angular grid to be used.  
The default value is 29. Valid values are:

LMAX	Number of points per radial shell
3	6
5	14
7	26
9	38
11	50
13	74
15	86
17	110

19	146
21	170
23	194
25	230
27	266
29	302
31	350
35	434
41	590
47	770
53	974
59	1202
65	1454
71	1730
77	2030
83	2354
89	2702
95	3074
101	3470
107	3890
113	4334
119	4802
125	5294
131	5810

LMXATM (integer array, dimension 128) An array to change the actual value of LMAX on per element number basis. For example, setting NRDATM(1)=25 will apply the angular grid of 25th order to all hydrogen atoms. The entry LMXATM(128) is used for dummy atoms with no charge. The default values are all equal to LMAX.

ANGPRN (logical) flag to activate the angular grid pruning as function of radius, uses the scheme similar to proposed by Murray, Handy and Laming (MHL). Default is .true. Setting it to false will slow down the calculations but will result to the more accurate results.

KAP (double precision) The parameter Ktheta used for angular pruning. The default value is 5.0 as recommended by MHL. Increasing this value to, say, 10.0 will improve precision by the cost of performance. This parameter has no effect if angular pruning is not used.

RADPRN (logical) flag to activate the radial grid pruning. Default is .true. Setting it to false will slow down calculations to some degree but may improve precision. Note that to get correct results you must explicitly disable radial pruning if your system has any basisless atoms!

RMXATM (integer array, dimension 128). An array to change the atomic cutoff radii used during radial pruning on per element number basis. For example, setting RMXATM(8)=6 will set the effective radius of all oxygen atoms to 6 Angstrom. The entry RMXATM(128) is used for dummy atoms with no charge. The default values are probably much larger than is actually necessary and depend on the basis set used.

CUTOFF (double precision). Contributions to the DFT Fock matrix due to batch of angular points, which are smaller than CUTOFF, are ignored. The default value is 1.0d-10.

CUTAO (double precision). If the absolute numerical value of AO is smaller than CUTAO, it will be set to zero during calculations. The default value is 1.0d-10.

CUTWGT (double precision). If the absolute value of weight associated with grid point is less than CUTWGT, this point will not be taken into account during DFT calculations. The default value is 1.0d-20.

CUTORB (double precision). Contributions to the DFT Fock matrix due to batch of orbitals, which are smaller than CUTORB, are ignored. The default value is 1.0d-15.

CUTGG1,  
CUTGG2,  
CUTGG3,  
CUTGG4 (double precision). Various cutoffs used during calculation of grid weights derivatives contributions to the molecular gradients. The default values are 1.0d-13, 1.0d-13, 1.0d-13, and 1.0d-30 correspondingly and are probably too strict.

B3LYP (symbolic). Selects the default implementation of B3LYP functional. The default value is NWCHEM which makes B3LYP to be synonym of B3LYP1. Setting this value to GAMESS will change B3LYP to be the same as B3LYP5.

O3LYP (symbolic). Setting it to GAUSSIAN changes the weight of non-local exchange in all O3LYP functionals so that the resulting functional will be identical to one used in Gaussian 03 Rev D.01 and above. The default value is DEFAULT which makes code compatible with O3LYP implementations in other programs. Note there are some ambiguities in O3LYP-related papers which do not allow to decide which implementation (default or Gaussian) is correct. These two implementations should be considered as just two different xc functionals.

Below is the sample input file.

```
! GAMESS (US) STYLE B3LYP OPTIMIZATION+HESSIAN CALCULATION OF WATER MOLECULE
$CONTRL SCFTYP=RHF DFTTYP=B3LYP5 runtyp=optimize $END
$SYSTEM TIMLIM=3000 MEMORY=3000000 $END
$BASIS GBASIS=n31 ngauss=6 NDFUNC=1 $END
$statpt hssend=.t. nprt=-2 $end
$force nvib=2 $end
$DATA
H2O
CNV 2

O      8.0    0.0000000000    0.0000000000    0.7205815395
H      1.0    0.0000000000    0.7565140024    0.1397092302
```

\$END

=====  
**\$BASIS group** (optional)

This group allows certain standard basis sets to be easily given. If this group is omitted, the basis set must be given instead in the \$DATA group.

GBASIS = Name of the Gaussian basis set.  
= MINI - Huzinaga's 3 gaussian minimal basis set.  
Available H-Rn.  
= MIDI - Huzinaga's 21 split valence basis set.  
Available H-Rn.  
= STO - Pople's STO-NG minimal basis set.  
Available H-Xe, for NGAUSS=2,3,4,5,6.  
= N21 - Pople's N-21G split valence basis set.  
Available H-Xe, for NGAUSS=3.  
Available H-Ar, for NGAUSS=6.  
= N31 - Pople's N-31G split valence basis set.  
Available H-Ne,P-Cl for NGAUSS=4.  
Available H-He,C-F for NGAUSS=5.  
Available H-Ar, for NGAUSS=6.  
For Ga-Kr, N31 selects the BC basis.  
= N311 - Pople's "triple split" N-311G basis set.  
Available H-Ne, for NGAUSS=6.  
Selecting N311 implies MC for Na-Ar.  
= DZV - "double zeta valence" basis set.  
a synonym for DH for H,Li,Be-Ne,Al-Cl.  
(14s,9p,3d)/[5s,3p,1d] for K-Ca.  
(14s,11p,5d)/[6s,4p,1d] for Ga-Kr.  
= DH - Dunning/Hay "double zeta" basis set.  
(3s)/[2s] for H.  
(9s,4p)/[3s,2p] for Li.  
(9s,5p)/[3s,2p] for Be-Ne.  
(11s,7p)/[6s,4p] for Al-Cl.  
= TZV - "triple zeta valence" basis set.  
(5s)/[3s] for H.  
(10s,3p)/[4s,3p] for Li.  
(10s,6p)/[5s,3p] for Be-Ne.  
a synonym for MC for Na-Ar.  
(14s,9p)/[8s,4p] for K-Ca.  
(14s,11p,6d)/[10s,8p,3d] for Sc-Zn.  
= MC - McLean/Chandler "triple split" basis.  
(12s,9p)/[6s,5p] for Na-Ar.  
Selecting MC implies 6-311G for H-Ne.

\* \* \* the next two are ECP bases only \* \* \*

GBASIS = SBKJC- Stevens/Basch/Krauss/Jasien/Cundari  
valence basis set, for Li-Rn. This choice  
implies an unscaled -31G basis for H-He.  
= HW - Hay/Wadt valence basis.

This is a -21 split, available Na-Xe,  
except for the transition metals.  
This implies a 3-21G basis for H-Ne.

\* \* \* semiempirical basis sets \* \* \*

The elements for which these exist can be found  
in the 'further information' section of this  
manual. If you pick one of these, all other data  
in this group is ignored. Semi-empirical runs  
actually use valence-only STO bases, not GTOs.

GBASIS = MNDO - selects MNDO model hamiltonian

= AM1 - selects AM1 model hamiltonian

= PM3 - selects PM3 model hamiltonian

NGAUSS = the number of Gaussians (N). This parameter  
pertains only to GBASIS=STO, N21, N31, or N311.

NDFUNC = number of heavy atom polarization functions to  
be used. These are usually d functions, except  
for MINI/MIDI. The term "heavy" means Na on up  
when GBASIS=STO, HW, or N21, and from Li on up  
otherwise. The value may not exceed 3. The  
variable POLAR selects the actual exponents to  
be used, see also SPLIT2 and SPLIT3. (default=0)

NFFUNC = number of heavy atom f type polarization  
functions to be used on Li-Cl. This may only  
be input as 0 or 1. (default=0)

NPFUNC = number of light atom, p type polarization  
functions to be used on H-He. This may not  
exceed 3, see also POLAR. (default=0)

DIFFSP = flag to add diffuse sp (L) shell to heavy atoms.  
Heavy means Li-F, Na-Cl, Ga-Br, In-I, Tl-At.  
The default is .FALSE.

DIFFS = flag to add diffuse s shell to hydrogens.  
The default is .FALSE.

Warning: if you use diffuse functions, please read  
INTTYP in the \$CONTRL group for numerical concerns.

EXTFIL = a flag to read basis sets from an external file,  
defined by GBASIS=name, rather than from a \$DATA group.  
(default=.false.)

The GBASIS "name" keyword must be an 8 or less character string,  
obviously not using any internally stored names. Every atom must be  
defined in the external file by a line giving the chemical symbol, and  
this chosen string. Following this header line, give the basis in free

format \$DATA style, containing only S, P, D, F, G, and L shells, and terminating each atom by the usual blank line. The external file may have several families of bases in the same file, identified by different GBASIS strings.

By default, PC GAMESS looks for the file called "basis name".lib in the directory where input file resides. If the -b < path to basisfile > option is given, then it looks for file called "path to basisfile" first. If it does not exist, it then tries to find file called basis.lib in the directory "path to basisfile".

POLAR = exponent of polarization functions  
= POPLE (default for all other cases)  
= POPN311 (default for GBASIS=N311, MC)  
= DUNNING (default for GBASIS=DH, DZV)  
= HUZINAGA (default for GBASIS=MINI, MIDI)  
= HONDO7 (default for GBASIS=TZV)

SPLIT2 = an array of splitting factors used when NDFUNC  
or NPFUNC is 2. Default=2.0,0.5

SPLIT3 = an array of splitting factors used when NDFUNC  
or NPFUNC is 3. Default=4.00,1.00,0.25

=====

The splitting factors are from the Pople school, and are probably too far apart. See for example the Binning and Curtiss paper. For example, the SPLIT2 value will usually cause an INCREASE over the 1d energy at the HF level for hydrocarbons.

The actual exponents used for polarization functions, as well as for diffuse sp or s shells, are described in the 'Further References' section of this manual. This section also describes the sp part of the basis set chosen by GBASIS fully, with all references cited.

Note that GAMESS always punches a full \$DATA group. Thus, if \$BASIS does not quite cover the basis you want, you can obtain this full \$DATA group from EXETYP=CHECK, and then change polarization exponents, add Rydbergs, etc.

=====

## **\$DATA group** (required)

This group describes the global molecular data such as point group symmetry, nuclear coordinates, and possibly the basis set. It consists of a series of free format card images.

-----

-1- TITLE a single descriptive title card.

-----  
-2- GROUP, NAXIS

GROUP is the Schoenflies symbol of the symmetry group, you may choose from

C1, CS, CI, CN, S2N, CNH, CNV, DN, DNH, DND,  
T, TH, TD, O, OH.

NAXIS is the order of the highest rotation axis, and must be given when the name of the group contains an N. For example, "Cnv 2" is C2v. "S2n 3" means S6.

For linear molecules, choose either CNV or DNH, and enter NAXIS as 4. Enter atoms as DNH with NAXIS=2. If the electronic state of either is degenerate, check the note about the effect of symmetry in the electronic state in the SCF section of REFS.DOC.

-----  
In order to use GAMESS effectively, you must be able to recognize the point group name for your molecule. This presupposes a knowledge of group theory at about the level of Cotton's "Group Theory", Chapter 3.

Armed with only the name of the group, GAMESS is able to exploit the molecular symmetry throughout almost all of the program, and thus save a great deal of computer time. GAMESS does not require that you know very much else about group theory, although a deeper knowledge (character tables, irreducible representations, term symbols, and so on) is useful when dealing with the more sophisticated wavefunctions.

Cards -3- and -4- are quite complicated, and are rarely given. A \*SINGLE\* blank card may replace both cards -3- and -4-, to select the 'master frame', which is defined on the next page. If you choose to enter a blank card, skip to the bottom of the next page.

Note!

If the point group is C1 (no symmetry), skip over cards -3- and -4- (which means no blank card).

-----  
-3- X1, Y1, Z1, X2, Y2, Z2

For C1 group, there is no card -3- or -4-.  
For CI group, give one point, the center of inversion.  
For CS group, any two points in the symmetry plane.  
For axial groups, any two points on the principal axis.  
For tetrahedral groups, any two points on a two-fold axis.  
For octahedral groups, any two points on a four-fold axis.

-----  
-4- X3, Y3, Z3, DIRECT

third point, and a directional parameter.  
For CS group, one point of the symmetry plane,  
noncollinear with points 1 and 2.  
For CI group, there is no card -4-.

For other groups, a generator sigma-v plane (if any) is  
the (x,z) plane of the local frame (CNV point groups).

A generator sigma-h plane (if any) is the (x,y) plane of  
the local frame (CNH and dihedral groups).

A generator C2 axis (if any) is the x-axis of the local  
frame (dihedral groups).

The perpendicular to the principal axis passing through  
the third point defines a direction called D1. If  
DIRECT='PARALLEL', the x-axis of the local frame coincides  
with the direction D1. If DIRECT='NORMAL', the x-axis of  
the local frame is the common perpendicular to D1 and the  
principal axis, passing through the intersection point of  
these two lines. Thus D1 coincides in this case with the  
negative y axis.

-----  
The 'master frame' is just a standard orientation for  
the molecule. By default, the 'master frame' assumes that  
1. z is the principal rotation axis (if any),  
2. x is a perpendicular two-fold axis (if any),  
3. xz is the sigma-v plane (if any), and  
4. xy is the sigma-h plane (if any).  
Use the lowest number rule that applies to your molecule.

Some examples of these rules:  
Ammonia (C3v): the unique H lies in the XZ plane (R1,R3).  
Ethane (D3d): the unique H lies in the YZ plane (R1,R2).  
Methane (Td): the H lies in the XYZ direction (R2). Since  
there is more than one 3-fold, R1 does not apply.  
HP=O (Cs): the mirror plane is the XY plane (R4).

In general, it is a poor idea to try to reorient the  
molecule. Certain sections of the program, such as the  
orbital symmetry assignment, do not know how to deal with  
cases where the 'master frame' has been changed.

Linear molecules (C4v or D4h) must lie along the z axis,  
so do not try to reorient linear molecules.

You can use EXETYP=CHECK to quickly find what atoms are  
generated, and in what order. This is typically necessary  
in order to use the general \$ZMAT coordinates.



\* \* \* \*

Depending on your choice for COORD in \$CONTROL,

```
if COORD=UNIQUE, follow card sequence U
if COORD=HINT,   follow card sequence U
if COORD=CART,  follow card sequence C
if COORD=ZMT,   follow card sequence G
if COORD=ZMTMPC, follow card sequence M
```

Card sequence U is the only one which allows you to define a completely general basis here in \$DATA.

Recall that UNIT in \$CONTRL determines the distance units.

-----  
-5U- Atom input. Only the symmetry unique atoms are input, GAMESS will generate the symmetry equivalent atoms according to the point group selected above.

```
if COORD=UNIQUE  NAME, ZNUC, X, Y, Z
*****
```

NAME = 10 character atomic name, used only for printout.  
Thus you can enter H or Hydrogen, or whatever.  
ZNUC = nuclear charge. It is the nuclear charge which actually defines the atom's identity.  
X,Y,Z = Cartesian coordinates.

```
if COORD=HINT
*****
```

```
NAME, ZNUC, CONX, R, ALPHA, BETA, SIGN, POINT1, POINT2, POINT3
```

NAME = 10 character atomic name (used only for print out).  
ZNUC = nuclear charge.  
CONX = connection type, choose from  
    'LC' linear conn.                    'CCPA' central conn.  
    'PCC' planar central conn.           with polar atom  
    'NPCC' non-planar central conn.      'TCT' terminal conn.  
    'PTC' planar terminal conn.           with torsion  
R = connection distance.  
ALPHA= first connection angle  
BETA = second connection angle  
SIGN = connection sign, '+' or '-'  
POINT1, POINT2, POINT3 =  
    connection points, a serial number of a previously  
    input atom, or one of 4 standard points: O, I, J, K  
    (origin and unit points on axes of master frame).

defaults: POINT1='O', POINT2='I', POINT3='J'

ref- R.L. Hilderbrandt, J.Chem.Phys. 51, 1654 (1969).  
You cannot understand HINT input without reading this.

Note that if ZNUC is negative, the internally stored  
basis for ABS(ZNUC) is placed on this center, but the  
calculation uses ZNUC=0 after this. This is useful  
for basis set superposition error (BSSE) calculations.

-----  
\* \* \* If you gave \$BASIS, continue entering cards -5U-  
until all the unique atoms have been specified.  
When you are done, enter a " \$END " card.  
\* \* \* If you did not, enter cards -6U-, -7U-, -8U-.

-----  
-6U- GBASIS, NGAUSS, (SCALF(i),i=1,4)

GBASIS has exactly the same meaning as in \$BASIS. You may  
choose from MINI, MIDI, STO, N21, N31, N311, DZV, DH, BC,  
TZV, MC, SBKJC, or HW. In addition, you may choose S, P,  
D, F, G, or L to enter an explicit basis set. Here, L  
means both an s and p shell with a shared exponent.

NGAUSS is the number of Gaussians (N) in the Pople style  
basis, or user input general basis. It has meaning only  
for GBASIS=STO, N21, N31, or N311, and S,P,D,F,G, or L.

Up to four scale factors may be entered. If omitted,  
standard values are used. They are not documented as  
every GBASIS treats these differently. Read the source  
code if you need to know more. They are seldom given.

-----  
\* \* \* If GBASIS is not S,P,D,F,G, or L, either add more  
shells by repeating card -6U-, or go on to -8U-.  
\* \* \* If GBASIS=S,P,D,F,G, or L, enter NGAUSS cards -7U-.

-----  
-7U- IG, ZETA, C1, C2

IG = a counter, IG takes values 1, 2, ..., NGAUSS.  
ZETA = Gaussian exponent of the IG'th primitive.  
C1 = Contraction coefficient for S,P,D,F,G shells,  
and for the s function of L shells.  
C2 = Contraction coefficient for the p in L shells.

-----  
\* \* \* For more shells on this atom, go back to card -6U-.  
\* \* \* If there are no more shells, go on to card -8U-.

-----  
-8U- A blank card ends the basis set for this atom.  
-----

Continue entering atoms with -5U- through -8U- until all are given, then terminate the group with a " \$END " card.

--- this is the end of card sequence U ---

COORD=CART input:

-----

-5C- Atom input.

Cartesian coordinates for all atoms must be entered. They may be arbitrarily rotated or translated, but must possess the actual point group symmetry. GAMESS will reorient the molecule into the 'master frame', and determine which atoms are the unique ones. Thus, the final order of the atoms may be different from what you enter here.

NAME, ZNUC, X, Y, Z

NAME = 10 character atomic name, used only for printout. Thus you can enter H or Hydrogen, or whatever.  
ZNUC = nuclear charge. It is the nuclear charge which actually defines the atom's identity.  
X,Y,Z = Cartesian coordinates.

-----

Continue entering atoms with card -5C- until all are given, and then terminate the group with a " \$END " card.

--- this is the end of card sequence C ---

COORD=ZMT input: (GAUSSIAN style internals)

-----

-5G- ATOM

Only the name of the first atom is required.  
See -8G- for a description of this information.

-----

-6G- ATOM i1 BLENGTH

Only a name and a bond distance is required for atom 2.  
See -8G- for a description of this information.

-----

-7G- ATOM i1 BLENGTH i2 ALPHA

Only a name, distance, and angle are required for atom 3.  
See -8G- for a description of this information.

-----

-8G-        ATOM   i1 BLENGTH   i2 ALPHA   i3 BETA   i4

ATOM        is the chemical symbol of this atom. It can be followed by numbers, if desired, for example Si3. The chemical symbol implies the nuclear charge.

i1           defines the connectivity of the following bond.

BLENGTH    is the bond length "this atom-atom i1".

i2           defines the connectivity of the following angle.

ALPHA       is the angle "this atom-atom i1-atom i2".

i3           defines the connectivity of the following angle.

BETA        is either the dihedral angle "this atom-atom i1-atom i2-atom i3", or perhaps a second bond angle "this atom-atom i1-atom i3".

i4           defines the nature of BETA,  
              If BETA is a dihedral angle, i4=0 (default).  
              If BETA is a second bond angle, i4=+/-1.  
              (sign specifies one of two possible directions).

-----

- o Repeat -8G- for atoms 4, 5, ...
  - o The use of ghost atoms is possible, by using X or BQ for the chemical symbol. Ghost atoms preclude the option of an automatic generation of \$ZMAT.
  - o The connectivity i1, i2, i3 may be given as integers, 1, 2, 3, 4, 5, ... or as strings which match one of the ATOMs. In this case, numbers must be added to the ATOM strings to ensure uniqueness!
  
  - o In -6G- to -8G-, symbolic strings may be given in place of numeric values for BLENGTH, ALPHA, and BETA. The same string may be repeated, which is handy in enforcing symmetry. If the string is preceeded by a minus sign, the numeric value which will be used is the opposite, of course. Any mixture of numeric data and symbols may be given. If any strings were given in -6G- to -8G-, you must provide cards -9G- and -10G-, otherwise you may terminate the group now with a " \$END " card.
- 

-9G-        A blank line terminates the Z-matrix section.

-----

-10G-       STRING VALUE

STRING is a symbolic string used in the Z-matrix.  
VALUE is the numeric value to substitute for that string.

-----

Continue entering -10G- until all STRINGS are defined.  
Note that any blank card encountered while reading -10G- will be ignored. GAMESS regards all STRINGS as variables

(constraints are sometimes applied in \$STATPT). It is not necessary to place constraints to preserve point group symmetry, as GAMESS will never lower the symmetry from that given at -2-. When you have given all STRINGS a VALUE, terminate the group with a " \$END " card.

--- this is the end of card sequence G ---

\* \* \* \*

The documentation for sequence G above and sequence M below presumes you are reasonably familiar with the input to GAUSSIAN or MOPAC. It is probably too terse to be understood very well if you are unfamiliar with these. A good tutorial on both styles of Z-matrix input can be found in Tim Clark's book "A Handbook of Computational Chemistry", published by John Wiley & Sons, 1985.

Both Z-matrix input styles must generate a molecule which possesses the symmetry you requested at -2-. If not, your job will be terminated automatically.

COORD=ZMTMPC input: (MOPAC style internals)

-----  
-5M- ATOM

Only the name of the first atom is required.  
See -8M- for a description of this information.  
-----

-6M- ATOM BLENGTH

Only a name and a bond distance is required for atom 2.  
See -8M- for a description of this information.  
-----

-7M- ATOM BLENGTH j1 ALPHA j2

Only a bond distance from atom 2, and an angle with respect to atom 1 is required for atom 3. If you prefer to hook atom 3 to atom 1, you must give connectivity as in -8M-. See -8M- for a description of this information.  
-----

-8M- ATOM BLENGTH j1 ALPHA j2 BETA j3 i1 i2 i3

ATOM, BLENGTH, ALPHA, BETA, i1, i2 and i3 are as described at -8G-. However, BLENGTH, ALPHA, and BETA must be given as numerical values only. In addition, BETA is always a dihedral angle. i1, i2, i3 must be integers only.

The j1, j2 and j3 integers, used in MOPAC to signal optimization of parameters, must be supplied but are ignored here. You may give them as 0, for example.

-----  
Continue entering atoms 3, 4, 5, ... with -8M- cards until all are given, and then terminate the group by giving a " \$END " card.

--- this is the end of card sequence M ---

=====  
This is the end of \$DATA!

If you have any doubt about what molecule and basis set you are defining, or what order the atoms will be generated in, simply execute an EXETYP=CHECK job to find out!

=====  
**\$ZMAT group** (required if NZVAR is nonzero in \$CONTRL)

This group lets you define the internal coordinates in which the gradient geometry search is carried out. These need not be the same as the internal coordinates used in \$DATA. The coordinates may be simple Z-matrix types, delocalized coordinates, or natural internal coordinates.

You must input a total of  $M=3N-6$  internal coordinates ( $M=3N-5$  for linear molecules). NZVAR in \$CONTRL can be less than M IF AND ONLY IF you are using linear bends. It is also possible to input more than M coordinates if they are used to form exactly M linear combinations for new internals. These may be symmetry coordinates or natural internal coordinates. If  $NZVAR > M$ , you must input IJS and SIJ below to form M new coordinates. See DECOMP in \$FORCE for the only circumstance in which you may enter a larger NZVAR without giving SIJ and IJS.

\*\*\*\* IZMAT defines simple internal coordinates \*\*\*\*

IZMAT is an array of integers defining each coordinate. The general form for each internal coordinate is  
code number, I, J, K, L, M, N

IZMAT =1 followed by two atom numbers. (I-J bond length)  
=2 followed by three numbers. (I-J-K bond angle)  
=3 followed by four numbers. (dihedral angle)  
Torsion angle between planes I-J-K and J-K-L.  
=4 followed by four atom numbers. (atom-plane)  
Out-of-plane angle from bond I-J to plane J-K-L.  
=5 followed by three numbers. (I-J-K linear bend)

Counts as 2 coordinates for the degenerate bend,  
normally J is the center atom. See \$LIBE.  
=6 followed by five atom numbers. (dihedral angle)  
Dihedral angle between planes I-J-K and K-L-M.  
=7 followed by six atom numbers. (ghost torsion)  
Let A be the midpoint between atoms I and J, and  
B be the midpoint between atoms M and N. This  
coordinate is the dihedral angle A-K-L-B. The  
atoms I,J and/or M,N may be the same atom number.  
(If I=J AND M=N, this is a conventional torsion).  
Examples: N2H4, or, with one common pair, H2POH.

Example - a nonlinear triatomic, atom 2 in the middle:

```
$ZMAT IZMAT(1)=1,1,2, 2,1,2,3, 1,2,3 $END
```

This sets up two bonds and the angle between them.  
The blanks between each coordinate definition are  
not necessary, but improve readability mightily.

\*\*\*\* the next define delocalized coordinates \*\*\*\*

DLC is a flag to request delocalized coordinates.  
(default is .FALSE.)

AUTO is a flag to generate all redundant coordinates,  
automatically. The DLC space will consist of all  
non-redundant combinations of these which can be  
found. The list of redundant coordinates will  
consist of bonds, angles, and torsions only.  
(default is .FALSE.)

NONVDW is an array of atom pairs which are to be joined  
by a bond, but might be skipped by the routine  
that automatically includes all distances shorter  
than the sum of van der Waals radii. Any angles  
and torsions associated with the new bond(s) are  
also automatically included.

The format for IXZMAT, IRZMAT, IFZMAT is that of IZMAT:

IXZMAT is an extra array of simple internal coordinates  
which you want to have added to the list generated  
by AUTO. Unlike NONVDW, IXZMAT will add only the  
coordinate(s) you specify.

IRZMAT is an array of simple internal coordinates which  
you would like to remove from the AUTO list of  
redundant coordinates. It is sometimes necessary  
to remove a torsion if other torsions around a bond  
are being frozen, to obtain a nonsingular G matrix.

IFZMAT is an array of simple internal coordinates which  
you would like to freeze. See also FVALUE below.  
Note that IFZMAT/FVALUE work only with DLC, see the  
IFREEZ option in \$STATPT to freeze coordinates if  
you wish to freeze simple or natural coordinates.

FVALUE is an array of values to which the internal coordinates should be constrained. It is not necessary to input \$DATA such that the initial values match these desired final values, but it is helpful if the initial values are not too far away.

AUTOFV is a flag to generate fvalue array automatically. Any user input of fvalue is thus ignored. This option is especially useful for relaxed PES scans in DLC. Default is .true.

DLCTOL is the threshold used at several stages during DLC generation, as well as to impose geometrical constraints. Normally you do not need to alter it, however, in some cases setting it to, say, 1d-7 can help generating required number of linearly independent DLC coordinates. Default is 1d-5.

IFRZAT is the array of atom numbers you want to freeze during geometry optimization using DLC. More precisely, the distances between all chosen atoms will be frozen, however, they will be allowed to rotate and translate as the single, united group (rigid body). Currently no more than 16 atoms may be frozen

\*\*\*\* SIJ,IJS define natural internal coordinates \*\*\*\*

SIJ is a transformation matrix of dimension NZVAR x M, used to transform the NZVAR internal coordinates in IZMAT into M new internal coordinates. SIJ is a sparse matrix, so only the non-zero elements are given, by using the IJS array described below. The columns of SIJ will be normalized by GAMESS. (Default: SIJ = I, unit matrix)

IJS is an array of pairs of indices, giving the row and column index of the entries in SIJ.

example - if the above triatomic is water, using

```
IJS(1) = 1,1, 3,1, 1,2, 3,2, 2,3
SIJ(1) = 1.0, 1.0, 1.0,-1.0, 1.0
```

```
gives the matrix S= 1.0  1.0  0.0
                   0.0  0.0  1.0
                   1.0 -1.0  0.0
```

which defines the symmetric stretch, asymmetric stretch, and bend of water.

references for natural internal coordinates:

P.Pulay, G.Fogarasi, F.Pang, J.E.Boggs  
J.Am.Chem.Soc. 101, 2550-2560(1979)  
G.Fogarasi, X.Zhou, P.W.Taylor, P.Pulay  
J.Am.Chem.Soc. 114, 8191-8201(1992)

reference for delocalized coordinates:

J.Baker, A. Kessi, B.Delley  
J.Chem.Phys. 105, 192-212(1996)

=====



**\$LIBE group** (required if linear bends are used in \$ZMAT)

A degenerate linear bend occurs in two orthogonal planes, which are specified with the help of a point A. The first bend occurs in a plane containing the atoms I,J,K and the user input point A. The second bend is in the plane perpendicular to this, and containing I,J,K. One such point must be given for each pair of bends used.

APTS(1)= x1,y1,z1,x2,y2,z2,... for linear bends 1,2,...

Note that each linear bend serves as two coordinates, so that if you enter 2 linear bends (HCCH, for example), the correct value of NZVAR is M-2, where M=3N-6 or 3N-5, as appropriate.

=====

**\$SCF group** relevant if SCFTYP = RHF, UHF, or ROHF,  
required if SCFTYP = GVB)

This group of parameters provides additional control over the RHF, UHF, ROHF, or GVB SCF steps. It must be used for GVB open shell or perfect pairing wavefunctions.

DIRSCF = a flag to activate a direct SCF calculation, which is implemented for all the Hartree-Fock type wavefunctions: RHF, ROHF, UHF, and GVB. This keyword also selects direct MP2 computation. The default of .FALSE. stores integrals on disk storage for a conventional SCF calculation.

FDIFF = a flag to compute only the change in the Fock matrices since the previous iteration, rather than recomputing all two electron contributions. This saves much CPU time in the later iterations. This pertains only to direct SCF, and has a default of .TRUE. This option is implemented only for the RHF, ROHF, UHF cases.

Cases with many diffuse functions in the basis set sometimes oscillate at the end, rather than converging. Turning this parameter off will normally give convergence.

---- The next flags affect convergence rates.

EXTRAP = controls Pople extrapolation of the Fock matrix.  
DAMP = controls Davidson damping of the Fock matrix.  
SHIFT = controls level shifting of the Fock matrix.  
RSTRCT = controls restriction of orbital interchanges.  
DIIS = controls Pulay's DIIS interpolation.  
SOSCF = controls second order SCF orbital optimization.  
(default=.TRUE. for RHF, Abelian group ROHF, GVB)  
(default=.FALSE. for UHF, non-Abelian group ROHF)

DEM = controls direct energy minimization, which is implemented only for RHF.

defaults for	EXTRAP	DAMP	SHIFT	RSTRCT	DIIS	SOSCF
ab initio:	T	F	F	F	T	T/F
semiempirical:	T	F	F	F	F	F

The above parameters are implemented for all SCF wavefunction types, except that DIIS will work for GVB only for those cases with NPAIR=0 or NPAIR=1. If both DIIS and SOSCF are chosen, SOSCF is stronger than DIIS, and so DIIS will not be used.

Once either DIIS or SOSCF are initiated, any other accelerator in effect is put in abeyance.

---- These parameters fine tune the various convergers.

NCONV = n SCF density convergence criteria.  
Convergence is reached when the density change between two consecutive SCF cycles is less than  $10.0^{**}(-n)$  in absolute value. One more cycle is executed after reaching convergence. Less accuracy in NCONV gives questionable gradients. (default is n=5, except CI or MP2 gradients n=6)

SOGTOL = second order gradient tolerance. SOSCF will be initiated when the orbital gradient falls below this threshold. (default=0.25 au)

ETHRSH = energy error threshold for initiating DIIS. The DIIS error is the largest element of e=FDS-SDF. Increasing ETHRSH forces DIIS on sooner. (default = 0.5 Hartree)

MAXDII = Maximum size of the DIIS linear equations, so that at most MAXDII-1 Fock matrices are used in the interpolation. (default=10)

DEMCUT = Direct energy minimization will not be done once the density matrix change falls below this threshold. (Default=0.5)

DMPCUT = Damping factor lower bound cutoff. The damping factor will not be allowed to drop below this value. (default=0.0)

note: The damping factor need not be zero to achieve valid convergence (see Hsu, Davidson, and Pitzer, J.Chem.Phys., 65, 609 (1976), see especially the section on convergence control), but it should not be astronomical either.

\* \* \* \* \*  
For more info on the convergence methods,  
see the 'Further Information' section.  
\* \* \* \* \*

----- miscellaneous options -----

UHFNOS = flag controlling generation of the natural orbitals of a UHF function. (default=.FALSE.)

MVOQ = 0 Skip MVO generation (default)  
= n Form modified virtual orbitals, using a cation with n electrons removed. Implemented for RHF, ROHF, and GVB. If necessary to reach a closed shell cation, the program might remove n+1 electrons. Typically, n will be about 6.

NPUNCH = SCF punch option  
= 0 do not punch out the final orbitals  
= 1 punch out the occupied orbitals  
= 2 punch out occupied and virtual orbitals  
The default is NPUNCH = 2.

----- options for virial scaling -----

VTSCAL = A flag to request that the virial theorem be satisfied. An analysis of the total energy as an exact sum of orbital kinetic energies is printed. The default is .FALSE.

This option is implemented for RHF, UHF, and ROHF, for RUNTYP=ENERGY, OPTIMIZE, or SADPOINT. Related input is as follows:

SCALF = initial exponent scale factor when VTSCAL is in use, useful when restarting. The default is 1.0.

MAXVT = maximum number of iterations (at a single geometry) to satisfy the energy virial theorem. The default is 20.

VTCONV = convergence criterion for the VT, which is satisfied when  $2\langle T \rangle + \langle V \rangle + R \times dE/dR$  is less than VTCONV. The default is 1.0D-6 Hartree.

For more information on this option, which is most economically employed during a geometry search, see M.Lehd and F.Jensen, J.Comput.Chem. 12, 1089-1096(1991).

The next parameters define the GVB wavefunction. Note that ALPHA and BETA also have meaning for ROHF. See also MULT in the \$CONTRL group. The GVB wavefunction assumes orbitals are in the order core, open, pairs.

NCO = The number of closed shell orbitals. The default almost certainly should be changed! (default=0).

NSETO = The number of sets of open shells in the function. Maximum of 10. (default=0)

NO = An array giving the degeneracy of each open shell set. Give NSETO values. (default=0,0,0,...).

NPAIR = The number of geminal pairs in the -GVB- function. Maximum of 12. The default corresponds to open shell SCF (default=0).

CICOEF = An array of ordered pairs of CI coefficients for the -GVB- pairs. For example, a two pair case for water, say, might be CICOEF(1)=0.95,-0.05,0.95,-0.05. If not normalized, as in the default, they will be. This parameter is useful in restarting a GVB run, with the current CI coefficients. (default = 0.90,-0.20,0.90,-0.20,...)

COUPLE = A switch controlling the input of F, ALPHA, and BETA. The default is to use internally stored values for these variables. Note ALPHA and BETA can be given for -ROHF-, as well as -GVB-. (Default=.FALSE.)

F = An vector of fractional occupations.

ALPHA = An array of A coupling coefficients given in lower triangular order.

BETA = An array of B coupling coefficients given in lower triangular order.

Note: The default for F, ALPHA, and BETA depends on the state chosen. Defaults for the most commonly occurring cases are internally stored.

\* \* \* \* \*  
 For more discussion of GVB/ROHF input  
 see the 'further information' section  
 \* \* \* \* \*

=====

***\$SCFMI group*** (optional, relevant if SCFTYP=RHF)

The SCF-MI method is a modification of the Roothaan equations that avoids basis set superposition error (BSSE) in intermolecular interaction calculations, by expanding each monomer's orbitals using only its own basis set. Thus, the resulting orbitals are not orthogonal. The

presence of a \$SCFMI group in the input triggers the use of this option.

The implementation is limited to two monomers, treated at the RHF level. The energy, gradient, and therefore numerical hessian are available. The SCF step may be run in direct SCF mode. The first 4 parameters must be given. All atoms of monomer A must be given in \$DATA before the atoms of monomer B.

NA = number of doubly occupied MOs on fragment A.  
NB = number of doubly occupied MOs on fragment B.  
MA = number of basis functions on fragment A.  
MB = number of basis functions on fragment B.

ITER = maximum number of SCF-MI cycles, overriding the usual MAXIT value. (default is 50).

DTOL = SCF-MI density convergence criteria. (default is 1.0d-10)

ALPHA = possible level shift parameter. (default is 0.0, meaning shifting is not used)

IOPT = prints additional debug information.  
= 0 standard outout (default)  
= 1 print for each SCF-MI cycle MOs, overlap between the MOs, CPU times.  
= 2 print some extra informations in secular systems solution.

MSHIFT = debugging option that permits to shift all the memory pointer of the SCF-MI section of code of the quantity MSHIFT (default is 0).

=====

"Modification of Roothan Equations to Exclude BSSE from Molecular Interaction Calculations"

E. Gianinetti, M. Raimondi, E. Tornaghi  
Int. J. Quantum Chem. 60, 157 (1996)

A. Famulari, E. Gianinetti, M. Raimondi, and M. Sironi  
Int. J. Quantum Chem. (1997), submitted.

=====

**\$MP2 group** (relevant to SCFTYP=RHF,UHF,ROHF if MPLEVL=2)

Controls 2<sup>nd</sup> order Moller-Plesset perturbation runs, if requested by MPLEVL in \$CONTRL. See also the DIRSCF keyword in \$SCF to select direct MP2. MP2 is implemented for RHF, high spin ROHF, or UHF wavefunctions. Analytic gradients and the first order correction to the wavefunction (i.e. properties) are available only for RHF. The \$MP2 group is usually not given. See also \$MCDPT.

METHOD = 2      Selects old style sequential UHF/ROHF MP2 energy integral transformation.

                  = -2      Activates a new sequential UHF/ROHF MP2 energy integral transformation code that requires approximately a half of memory used in method 2. Default is -2.

METHOD = 1      Selects the usage of the new RHF/ROHF/UHF MP2 energy program. This program is intended to handle large systems (e.g., 500 AOs or more). It is direct, very fast, and requires much less memory as compared to other MP2 methods. **It is definitely the method that is to be chosen for the large jobs. On the other hand, it requires the 2-electron AO integrals to be reevaluated four times. Thus, there can be a considerable overhead for the small jobs. This is the only reason why this method is not turned on by default.**

NCORE = n      Omits the first n occupied orbitals from the calculation. The default for n is the number of chemical core orbitals.

MP2PRP=      a flag to turn on property computation for RHF MP2 jobs with RUNTYP=ENERGY. This is appreciably more expensive than just evaluating the 2nd order energy correction alone, so the default is .FALSE. Properties are always computed during gradient runs, when they are an almost free byproduct.

LMOMP2=      a flag to analyze the closed shell MP2 energy in terms of localized orbitals. Any type of localized orbital may be used. This option is implemented only for RHF, and its selection forces use of the METHOD=3 transformation. The default is .FALSE.

OSPT=      selects open shell spin-restricted perturbation. This parameter applies only when SCFTYP=ROHF. Please see the 'further information' section for more information about this choice.

                  = ZAPT picks Z-averaged perturbation theory. (default)

                  = RMP picks RMP (aka ROHF-MBPT) perturbation theory.

CUTOFF=      transformed integral retention threshold, the default is 1.0d-9.

GRDMET = 1      Selects old style MP2 gradient calculations

                  = 2      Activates the new MP2 gradient code, which is generally much faster. (default)

DM2MET = -3,-2,-1,0,1,2,3      Selects one of seven(!) possible methods of MP2 DM2 calculations during MP2 gradient run. Methods -3, -2, and -1 are precisely the same as methods 3, 2, and 1, respectively, but they use an asynchronous disk I/O to eliminate I/O latency and to reduce the total execution time; for this reason, these methods are not

supported under DOS. Method 0 is definitely the slowest one; which one is the fastest depends on the amount of the physical memory installed. Methods 2(-2) require the least CPU time, but probably not the least total time. Methods 3(-3) need the least memory. By default, method 1 is chosen, which seems to be a quite reasonable compromise. You can experiment with this option to find optimal settings for your particular task and environment. Relevant only when GRDMET=2. Default is 1.

MEMGRD = The maximum number of words of the memory to be used in the MP2 gradient calculation. It controls only the last stage of calculations, namely, the evaluation of the gradient integrals themselves. Zero means the usage of all available memory. Relevant only when GRDMET=2. The default value is 0.

MXCPIT = Controls the maximum number of AO CPHF iterations during the MP2 gradient runs. The default value is 100.

CPTOL = Controls the maximum allowed RMS error of the approximate solution of CPHF equations during MP2 gradient runs. The default value is 10<sup>-10</sup>.

MXRECL = The maximum record size for direct access file used during the MP2 energy (method=3) and MP2 gradient runs. The default maximum size is 65536 DP words.

IOPARS = Specifies the set of specific I/O optimization flags for the direct access files (DAF) used during MP2/MP2 GRADIENT runs. Actually, this is the 2-digit octal value. The least significant digit controls the general DAF writing operations, while the most significant one controls the behavior of 'READ-THEN-WRITE-TO-THE-SAME-LOCATION' operation. Each digit can have one of the following values:

- \* 0 - no particular I/O optimization;
- \* 1 - no particular I/O optimization, file cache write through mode;
- \* 2 - optimizes file I/O for the sequential access;
- \* 3 - optimizes file I/O for the sequential access, file cache write through mode;
- \* 4 - optimizes file I/O for the random access;
- \* 5 - optimizes file I/O for the random access, file cache write through mode;
- \* 6 - optimizes file I/O for the random access with some locality;
- \* 7 - optimizes file I/O for the random access with some locality, file cache write through mode;

The default value for this option is 74 in the case of non-negative DM2MET, and 75 otherwise. If you have a plenty of RAM, or fast RAID, or your job creates DAF of a (very) small size, try to set IOPARS to 64 and see if the I/O speed changes. The optimal value may also depend on the OS used.

DIRECT = .False. Selects the semidirect evaluation of the AO

integrals during METHOD=1 MP2 calculations.  
(default)

= .True. Selects the fully direct evaluation of AO integrals during METHOD=1 MP2 calculations. This somewhat reduces the amount of disk space needed, by the cost of the additional CPU time (which may be considerable). Relevant only for the METHOD=1 case.

PACKAO =..False. Normal operation.  
= .True. Activates packing of the AO integrals during METHOD=1 semidirect MP2 calculations, thus (slightly) reducing the amount of disk space needed. Relevant only for the METHOD=1 semidirect case. Default is .True.

MNRECL = The minimum allowed record size (in terms of 12-byte elements) for the direct access file used during METHOD=1 MP2 calculations. Reducing the MNRECL value increases the job execution time to some degree, but decreases the amount of memory needed for calculations. It is not recommended to set MNRECL below 500. Relevant only for the METHOD=1 case. Default is 20000.

SVDISK = .False. This allows the METHOD=1 ROHF MP2 program to use (in some rare cases) additional disk space in order to reduce the job CPU time. (default)  
= .True. Disables the extra disk space usage. Relevant only for the METHOD=1 ROHF MP2 case.

SPARSE =.False. This options controls how the matrix-matrix multiplication is performed during the MP2 METHOD=1 runs. False means the usage of standard BLAS level 3 routines (the only strategy, which was implemented in the PC GAMESS 4.4 and earlier). This is the best choice when the list of two-electron integrals is dense enough, i.e., when the total number of nonzero integrals is less than the maximum of  $N^4/8$  only several times (up to 3-5). Relevant only for the METHOD=1 MP2 case.  
= .True. Selects the usage of special matrix-matrix multiplication routines. This strategy is optimal for sparse lists of two-electron integrals. By default, the PC GAMESS tries to choose the most appropriate strategy automatically. Note, the program choice is in some situations not the best one. This is especially true for SMP systems, or for the systems with the slow access to the main memory

The last 3 input variables apply only to machines with an incomplete DDI implementation, i.e. those computers using ddi.src instead of a proper implementation. At present, only IBM SP & Cray T3E use the good parallel MP2.

NWORD = controls memory usage. The default uses all available memory. (default=0)



METHOD= n selects transformation method, 2 being the segmented transformation, and 3 being a more conventional two phase bin sort implementation. 3 requires more disk, but less memory. The default is to attempt method 2 first, and method 3 second.

AOINTS= defines AO integral storage during conventional integral transformations, during parallel runs. DUP stores duplicated AO lists on each node, and is the default for parallel computers with slow interprocessor communication, e.g. ethernet. DIST distributes the AO integral file across all nodes, and is the default for parallel computers with high speed communications.

Main characteristics of the MP2 implementation in PcGamess/Firefly:

Memory requirements scale as approximately  $N^2$ .

The other MP2 programs currently implemented in the GAMESS scale as at least  $N^3$  for the segmented transformation and as  $A N^2$  for the alternative integral transformation. Here, A and N are the number of active orbitals and the total number of MOs, respectively.

Disk requirements scale as  $A^2 N^2$ . They scale as  $A^2 (N-A)^2$  for the alternative integral transformation. The segmented transformation does not use temporary disk storage. The disk I/O is used, but to a very limited degree. Therefore, the CPU utilization is usually 90% or even better.

The CPU utilization is usually less than 50% for other MP2 transformation methods working in the conventional mode, while, in the direct mode, there is a very serious overhead because of the multiple reevaluation of 2-electron integrals.

Asymptotically, the FLOPs count is about a half or even better as compared to other MP2 energy transformation methods.

It uses SMP more efficiently.

On the other hand, it requires the 2-electron AO integrals to be reevaluated four times. This cost is fixed and does not depend on the details of the MP2 calculation performed.

=====

### **\$MP2GRD group**

(requested by \$MP2 GRDMET=1 or GRDMET=2 )

New semidirect MP2 properties/gradient code was developed to speed up MP2 gradient and related jobs, to seriously reduce memory demands and to allow efficient parallel execution. It is based on FASTINTS code and runs in parallel using P2P interface. It has very modest memory demands, namely, asymptotically they are proportional to  $N_{occ} \cdot (N_{occ} + N_{core}) \cdot N_{vir}$ , where  $N_{occ}$  is the number of active (not counting frozen core) occupied orbitals,  $N_{core}$  is the number of frozen core occupied orbitals, and  $N_{vir}$  is the number of virtual orbitals. Intrinsically, it is very fast and has very good parallel scalability, with most of the memory

demands reducing linearly with the number of used computing nodes. At present, it is limited to RMP2 case only, but will be extended for ZAPT2 in the future.

New MP2 gradient code uses disk I/O and (large) temporary files to store half-transformed 2-electron integrals and non-separable part of MP2 2-particle density matrix (DM2) and thus can be disk I/O intensive. When running in parallel, the contents of these files is evenly distributed across all nodes. If one is running this code in parallel on SMP or multicore system, it is recommended to assign a separate physical disk to each of the parallel PC GAMESS processes.

The new code is controlled by two keywords of the \$MP2 group and by several additional keywords of \$MP2GRD group. New code is triggered on by the presence of METHOD=1 keyword in \$MP2 group and its behavior is further controlled by either \$MP2 GRDMET=1 \$END or \$MP2 GRDMET=2 \$END (the default) switch. GRDMET=2 uses less memory and is usually faster, but for some combinations of computer systems and particular tasks GRDMET=1 can be the preferred option.

The fine tuning of the new properties/gradient module is possible via the following keywords of the \$MP2GRD group:

CACHE (logical, default value depends on the particular task). If turned on, reduces disk reads during second MP2 gradient 2-electron integral half-transformation by using some extra memory.

ASYNCR (logical, default is true). If turned on, activates asynchronous disk I/O during 2-e gradient computation by using some extra memory

ASYSNCR (logical, default is false). When turned on, activates asynchronous "sync" (e.g., "disk flush") calls.

FUSED (logical, default is true). Changes the ordering of loops so that the disk I/O becomes less irregular.

DBLBF (logical, default is true). Enables double buffering and asynchronous I/O for disk writes during second MP2 gradient 2-electron integral half-transformation. Causes minor memory overhead. See also the discussion of ioflgs(30)=1 option below.

GLBLSN (logical, default is false). Enables global synchronization of disk I/O during second MP2 gradient 2-electron integral half-transformation when running in parallel.

MXI (integer, default is  $\min(N_{occ}, 256)$ ). The maximum number of occupied orbitals to be processed in one slice during first and second integral half-transformation. Seriously affects amount of memory required by both first and second half-transformation. Cannot be larger than 256 by design.

MXPQ (integer, default is 225). The maximum number of AO orbital pairs to be processed in one slice during second 2-e integral half-transformation and 2-e gradient computation. Seriously affects amount of memory required by both second half-transformation and gradient code itself

RSIZE1 (integer, default value is 32768). Record size (in elements) used to store half-transformed 2-e integrals. Seriously affects amount of memory required by the first half-transformation, as well as disk I/O performance.

RSIZE2 (integer, default value is 32768). Record size (in elements) used to store half-transformed DM2 elements. Seriously affects amount of memory required by the second half-transformation, as well as disk I/O performance.

NBUFS (integer, default value is calculated to be close to the optimum). Number of buffers used for double buffered asynchronous I/O (see DBLBF option).

TOL1 (double precision, default value is  $10^{-9}$ ). Threshold to store half-transformed 2-e integrals.

TOL2 (double precision, default value is  $10^{-9}$ ). Threshold to store half-transformed DM2 elements.

RANDOM (double precision, default value is 1/3). The random seed to initialize random number generator used while distributing AO shell pairs over nodes.

=====

**\$MP3 group** (relevant to SCFTYP=RHF,UHF,ROHF if MPLEVEL=2)

**\$MP4 group** (relevant to SCFTYP=RHF,UHF,ROHF if MPLEVEL=2)

NCORE = Omits first NCORE occupied orbitals from the calculation. The default for NCORE is the number of chemical core orbitals.

NWORD = Controls the memory usage during MP3/MP4 calculations. The default value for NWORD is 0 which means that all available memory is used.

CUTOFF = This parameter controls the precision, to which the contributions to the MP3 and MP4(D) energies due to the external exchange operator, are calculated. The default value for CUTOFF is  $10^{-9}$ . For MP3 energy, it is guaranteed that the absolute value of the cumulative error will be less than the value of CUTOFF. For the MP4(D) part, the situation is usually the same, but, unlike MP3 energy, this is not always true, especially for the basis sets with partial linear dependence.

Due to this reason, it is recommended to use stronger CUTOFF (e.g.,  $10^{-15}$ ), if the high-precision MP4 energy is required. This will increase the precision by the cost of additional CPU time.

SMPMET = Selects one of the three SMP-based strategies available for the calculation of the external exchange operator contribution to the MP4 energy (this option has no effect in the case of MP3 calculations).

= 1 Method 1 is probably the best one, as it has very good SMP scaling properties. On the other hand, its memory

requirements are proportional to the number of CPUs used.  
 (default)

- = 2 Method 2 does not require extra memory, but can be used only if the number of CPUs is even. Furthermore, its SMP scaling is generally worse than that of Method 1.
- = 3 Method 3 also does not require extra memory. It usually has the worst SMP scaling properties.

SDTQ = .False. Performs MP4-SDQ calculation.  
 = .True. Performs MP4-SDTQ (i.e., full MP4) calculation. Default is False, i.e., only MP4-SDQ energy is calculated.

=====

### **\$GUESS group** (optional, relevant for all SCFTYP's)

This group controls the selection of initial molecular orbitals.

GUESS = Selects type of initial orbital guess.

- = HUCKEL Carry out an extended Huckel calculation using a Huzinaga MINI basis set, and project this onto the current basis. This is implemented for atoms up to Rn, and will work for any all electron or ECP basis set. (default for most runs)
- = HCORE Diagonalize the one electron Hamiltonian to obtain the initial guess orbitals. This method is applicable to any basis set, but does not work as well as the HUCKEL guess.
- = MOREAD Read in formatted vectors punched by an earlier run. This requires a \$VEC group, and you MUST pay attention to NORB below.
- = MOSAVED (default for restarts) The initial orbitals are read from the DICTNRY file of the earlier run.
- = SKIP Bypass initial orbital selection. The initial orbitals and density matrix are assumed to be in the DICTNRY file.

All GUESS types except 'SKIP' permit reordering of the orbitals, carry out an orthonormalization of the orbitals, and generate the correct initial density matrix. The initial density matrix cannot be generated for -CI- and -MCSCF-, so property restarts for these wavefunctions will require 'SKIP' which is an otherwise seldom used option. Note that correct computation of a -GVB- density matrix requires CICOEF in \$SCF. Another possible use for 'SKIP' is to speed up a EXETYP=CHECK job, or a RUNTYP=HESSIAN job where the hessian is supplied.

PRTMO = a flag to control printing of the initial guess.  
 (default=.FALSE.)

## \$VEC

NORB = The number of orbitals to be read in the \$VEC group. This applies only to GUESS=MOREAD.

For -RHF-, -UHF-, -ROHF-, and -GVB-, NORB defaults to the number of occupied orbitals. NORB must be given for -CI- and -MCSCF-. For -UHF-, if NORB is not given, only the occupied alpha and beta orbitals should be given, back to back. Otherwise, both alpha and beta orbitals must consist of NORB vectors.

NORB may be larger than the number of occupied MOs, if you wish to read in the virtual orbitals. If NORB is less than the number of atomic orbitals, the remaining orbitals are generated as the orthogonal complement to those read.

NORDER = Orbital reordering switch.  
= 0 No reordering (default)  
= 1 Reorder according to IORDER and JORDER.

IORDER = Reordering instructions.  
Input to this array gives the new molecular orbital order. For example, IORDER(3)=4,3 will interchange orbitals 3 and 4, while leaving the other MOs in the original order. This parameter applies to all orbitals (alpha and beta) except for -UHF-, where it only affects the alpha MOs. (default is IORDER(i)=i )

JORDER = Reordering instructions.  
Same as IORDER, but for the beta MOs of -UHF-.

TOLZ = level below which MO coefficients will be set to zero. (default=1.0E-7)

TOLE = level at which MO coefficients will be equated. This is a relative level, coefficients are set equal if one agrees in magnitude to TOLE times the other. (default=5.0E-5)

SYMDEN = call a routine to project the totally symmetric component out of a density matrix. This may be useful if the HUCKEL or HCORE give orbitals with inexact symmetry. Since the density matrix is not idempotent, this can generate a non-variational energy on the first iteration. For the same reason, this should never be used with orbitals of MOREAD quality. (default=.FALSE.)

MIX = rotate the alpha and beta HOMO and LUMO orbitals so as to generate inequivalent alpha and beta orbital spaces. This pertains to UHF singlets only. This may require use of NOSYM=1 in \$CONTRL depending on your situation. (default=.FALSE.)

KDIAG = diagonalization control switch

- = 0 selects use of very stable and fast diagonalization routine which requires large amount of extra memory.
- = -1 selects potentially less stable but even faster diagonalization routine which uses less memory than kdiag=0
- = -2 selects a combination of two above mentioned methods which can be more stable than kdiag=-1, is usually as fast as kdiag=-1 but requires as much memory as kdiag=0
- = 1 use EVVRSP diagonalization. This may be more accurate than Gamess (US) KDIAG=0.
- = 2 use GIVEIS diagonalization (not as fast or reliable as EVVRSP)
- = 3 use JACOBI diagonalization (this is the slowest method) (DEFAULT) This default is not

sensible, but assures compatibility with Gamess (US). It is generally recommended to set kdiag=0 in the \$guess group in all the PC GAMESS input files, especially if the number of basis functions is large enough.

The Fastdiag dynamic library (fastdiag.dll in Windows PC GAMESS distribution and fastdiag.ex in Linux version) contains fast optimized modern algorithms for symmetric matrix diagonalization and inversion and is intended to improve the performance of initial guess generation, DIIS extrapolation, as well as some other computationally-intensive steps. You should put it into the same folder as the PC GAMESS executables (Windows users) and in your working/scratch directory (Linux users). It should be renamed to be all lower-case (Linux users).

Note that fast diagonalization routines uses extra memory which is not taken into account during check runs at present. Thus it is recommended to reserve some additional amount of memory for diagonalization routines. For example, if you have a job with 1000 basis functions, it is a good idea to add ca. 2.5-3MW of memory for diagonalization purposes. Otherwise, the slower built-in routine will be called if the amount of memory is not enough to use the fast routines.

Finally, it should be noted that fastdiag is not compatible with and is not used by the generic Pentium PC GAMESS version.

=====

**\$VEC group** (optional, relevant for all SCFTYP's)  
(required if GUESS=MOREAD)

This group consists of formatted vectors, as written onto file PUNCH in a previous run. It is considered good form to retain the titling comment cards punched before the \$VEC card, as a reminder to yourself of the origin of the orbitals.

For Morokuma decompositions, the names of this group are \$VEC1, \$VEC2, ... for each monomer, computed in the identical orientation as the supermolecule. For transition moment or spin-orbit coupling runs, orbitals for states one and possibly two are \$VEC1 and \$VEC2.

=====

**\$STATPT group** (optional, for RUNTYP=OPTIMIZE or SADPOINT)

This group controls the search for stationary points. Note that NZVAR in \$CONTRL determines if the geometry search is conducted in Cartesian or internal coordinates.

METHOD = optimization algorithm selection. Pick from

NR Straight Newton-Raphson iterate. This will attempt to locate the nearest stationary point, which may be of any order. There is no steplength control. RUNTYP can be either OPTIMIZE or SADPOINT

RFO Rational Function Optimization. This is one of the augmented Hessian techniques where the shift parameter(s) is(are) chosen by a rational function approximation to the PES. For SADPOINT searches it involves two shift parameters. If the calculated stepsize is larger than DXMAX the step is simply scaled down to size.

QA Quadratic Approximation. This is another version of an augmented Hessian technique where the shift parameter is chosen such that the steplength is equal to DXMAX. It is completely equivalent to the TRIM method. (default)

SCHLEGEL The quasi-NR optimizer by Schlegel.

CONOPT, CONstrained OPTimization. An algorithm which can be used for locating TSs. The starting geometry MUST be a minimum! The algorithm tries to push the geometry uphill along a chosen Hessian mode (IFOLLOW) by a series of optimizations on hyperspheres of increasingly larger radii. Note that there currently are no restart capabilities for this method, not even manually.

GDIIS Geometry optimisation using Direct Inversion in the Iterative Subspace (highly recommended)

OPTTOL = gradient convergence tolerance, in Hartree/Bohr. Convergence of a geometry search requires the largest component of the gradient to be less than OPTTOL, and the root mean square gradient less than 1/3 of OPTTOL. (default=0.0001)

NSTEP = maximum number of steps to take. Restart data is punched if NSTEP is exceeded. (default=20)

NOREG = number of steps before delocalized coordinates are regenerated.

Selecting a low value for NOREG is advised when your DLC based optimization aborts after a few steps due to failure to generate good delocalized coordinates.

--- the next four control the step size ---

DXMAX = initial trust radius of the step, in Bohr.  
For METHOD=RFO, QA, or SCHLEGEL, steps will be scaled down to this value, if necessary. (default=0.3 for OPTIMIZE and 0.2 for SADPOINT)  
For METHOD=NR, DXMAX is inoperative.  
For METHOD=CONOPT, DXMAX is the step along the previous two points to increment the hypersphere radius between constrained optimizations. (default=0.1)

the next three apply only to METHOD=RFO or QA:

TRUPD = a flag to allow the trust radius to change as the geometry search proceeds. (default=.TRUE.)  
TRMAX = maximum permissible value of the trust radius. (default=0.5 for OPTIMIZE and 0.3 for SADPOINT)  
TRMIN = minimum permissible value of the trust radius. (default=0.05)

--- the next three control mode following ---

IFOLLOW = Mode selection switch, for RUNTYP=SADPOINT.  
For METHOD=RFO or QA, the mode along which the energy is maximized, other modes are minimized. Usually referred to as "eigenvector following".  
For METHOD=SCHLEGEL, the mode whose eigenvalue is (or will be made) negative. All other curvatures will be made positive.  
For METHOD=CONOPT, the mode along which the geometry is initially perturbed from the minima. (default is 1)  
In Cartesian coordinates, this variable doesn't count the six translation and rotation degrees. Note that the "modes" aren't from mass-weighting.

STPT = flag to indicate whether the initial geometry is considered a stationary point. If .true. the initial geometry will be perturbed by a step along the IFOLLOW normal mode with stepsize STSTEP. (default=.false.)  
The positive direction is taken as the one where the largest component of the Hessian mode is positive. If there are more than one largest component (symmetry), the first is taken as positive.  
Note that STPT=.TRUE. has little meaning with HESS=GUESS as there will be many degenerate eigenvalues.



STSTEP = Stepsize for jumping off a stationary point.  
Using values of 0.05 or more may work better.  
(default=0.01)

IFREEZ = array of internal coordinates to freeze.  
For example, IFREEZ(1)=1,3 freezes the two bond  
lengths in the \$ZMAT example, while optimizing  
the angle. You cannot freeze Cartesian coords.

--- The next two control the hessian matrix quality ---

HESS = selects the initial hessian matrix.  
= GUESS chooses a positive definite diagonal  
hessian. (default for RUNTYP=OPTIMIZE)  
= READ causes the hessian to be read from a \$HESS  
group. (default for RUNTYP=SADPOINT)  
= RDAB reads only the ab initio part of the  
hessian, and approximates the effective  
fragment blocks.  
= RDALL reads the full hessian, then converts  
any fragment blocks to 6x6 T+R shape.  
(this option is seldom used).  
= CALC causes the hessian to be computed, see  
the \$FORCE group.

IHREP = the number of steps before the hessian is  
recomputed. If given as 0, the hessian will  
be computed only at the initial geometry if  
you choose HESS=CALC, and never again. If  
nonzero, the hessian is recalculated every  
IHREP steps, with the update formula used on  
other steps. (default=0)

--- the next two control the amount of output ---

Let 0 mean the initial geometry, L mean the last  
geometry, and all mean every geometry.

Let INTR mean the internuclear distance matrix.

Let HESS mean the approximation to the hessian.

Note that a directly calculated hessian matrix  
will always be punched, NPUN refers only to the  
updated Hessians used by the quasi-Newton step.

NPRT = 1 Print INTR at all, orbitals at all  
0 Print INTR at all, orbitals at 0+L (default)  
-1 Print INTR at all, orbitals never  
-2 Print INTR at 0+L, orbitals never

NPUN = 3 Punch all orbitals and HESS at all  
2 Punch all orbitals at all  
1 same as 0, plus punch HESS at all  
0 Punch all orbitals at 0+L, otherwise only  
occupied orbitals (default)

- 1 Punch occ orbitals at 0+L only
- 2 Never punch orbitals

HSEND = a flag to control automatic hessian evaluation at the end of a successful geometry search. (default=.FALSE.)

MOVIE = a flag to create a series of structural data which can be show as a movie by the MacIntosh program Chem3D. The data is written to the file IRCDATA. (default=.FALSE.)

---- the following parameters are quite specialized ----

PURIFY = a flag to help eliminate the rotational and translational degrees of freedom from the initial hessian (and possibly initial gradient). This is much like the variable of the same name in \$FORCE, and will be relevant only if internal coordinates are in use. (default=.FALSE.)

PROJCT = a flag to eliminate translation and rotational degrees of freedom from Cartesian optimizations. The default is .TRUE. since this normally will reduce the number of steps, except that this variable is set false when POSITION=FIXED is used during EFP runs.

ITBMAT = number of micro-iterations used to compute the step in Cartesians which corresponds to the desired step in internals. The default is 5.

UPHESS = SKIP do not update Hessian (not recommended)  
BFGS default for OPTIMIZE using RFO or QA  
POWELL default for OPTIMIZE using NR or CONOPT  
POWELL default for SADPOINT  
MSP mixed Murtagh-Sargent/Powell update  
SCHLEGEL only choice for METHOD=SCHLEGEL

RESTAR = Enables restart of an optimization run. This can only be used with IREST .ne. 0 in \$CONTRL. Use of this variable is discouraged.

---- NNEG, RMIN, RMAX, RLIM apply only to SCHLEGEL ----

NNEG = The number of negative eigenvalues the force constant matrix should have. If necessary the smallest eigenvalues will be reversed. The default is 0 for RUNTYP=OPTIMIZE, and 1 for RUNTYP=SADPOINT.

RMIN = Minimum distance threshold. Points whose root mean square distance from the current point is less than RMIN are discarded. (default=0.0015)

RMAX = Maximum distance threshold. Points whose root mean square distance from the current point is greater than RMAX are discarded. (default=0.1)

RLIM = Linear dependence threshold. Vectors from the current point to the previous points must not be colinear. (default=0.07)

=====

\* \* \* \* \*  
See the 'further information' section for  
some help with OPTIMIZE and SADPOINT runs  
\* \* \* \* \*

=====

### ***\$TRUDGE group*** (optional, required for RUNTYP=TRUDGE)

This group defines the parameters for a non-gradient optimization of exponents or the geometry. The TRUDGE package is a modified version of the same code from Michel Dupuis' HONDO 7.0 system, originally written by H.F.King. Presently the program allows for the optimization of 10 parameters.

Exponent optimization works only for uncontracted primitives, without enforcing any constraints. Two non-symmetry equivalent H atoms would have their p function exponents optimized separately, and so would two symmetry equivalent atoms! A clear case of GIGO.

Geometry optimization works only in HINT internal coordinates (see \$CONTRL and \$DATA groups). The total energy of all types of SCF wavefunctions can be optimized, although this would be extremely stupid as gradient methods are far more efficient. The main utility is for open shell MP2 or CI geometry optimizations, which may not be done in any other way with GAMESS.

OPTMIZ = a flag to select optimization of either geometry or exponents of primitive gaussian functions.  
= BASIS for basis set optimization.  
= GEOMETRY for geometry optimization (default).  
This means minima search only, there is no saddle point capability.

NPAR = number of parameters to be optimized.

IEX = defines the parameters to be optimized.

If OPTMIZ=BASIS, IEX declares the serial number of the Gaussian primitives for which the exponents

will be optimized.

If OPTMIZ=GEOMETRY, IEX define the pointers to the HINT internal coordinates which will be optimized. (Note that not all internal coordinates have to be optimized.) The pointers to the internal coordinates are defined as: (the number of atom on the input list)\*10 + (the number of internal coordinate for that atom). For each atom, the HINT internal coordinates are numbered as 1, 2, and 3 for BOND, ALPHA, and BETA, respectively.

P = Defines the initial values of the parameters to be optimized. You can use this to reset values given in \$DATA. If omitted, the \$DATA values are used. If given here, geometric data must be in Angstroms and degrees.

A complete example is a TCSCF multireference 6-31G geometry optimization for methylene,

```
$CONTRL SCFTYP=GVB CITYP=GUGA RUNTYP=TRUDGE
        COORD=HINT $END
$BASIS  GBASIS=N31 NGAUSS=6 $END
$DATA
Methylene TCSCF+CISD geometry optimization
Cnv 2

C   6.    LC  0.00  0.0  0.00  -  O  K
H   1.    PCC 1.00  53.  0.00  +  O  K  I
$END
$SCF    NCO=3 NPAIR=1 $END
$TRUDGE OPTMIZ=GEOMETRY NPAR=2
        IEX(1)=21,22  P(1)=1.08 $END
$CIDRT  GROUP=C2V SOCI=.TRUE. NFZC=1 NDOC=3 NVAL=1
        NEXT=-1 $END
```

using GVB-PP(1), or TCSCF orbitals in the CI. The starting bond length is reset to 1.09, while the initial angle will be 106 (twice 53). Result after 17 steps is R=1.1283056, half-angle=51.83377, with a CI energy of -38.9407538472

Note that you may optimize the geometry for an excited CI state, just specify

```
$GUGDIA  NSTATE=5 $END
$GUGDM   IROOT=3  $END
```

to find the equilibrium geometry of the third state (of five total states) of the symmetry implied by your \$CIDRT.

=====

## **\$TRURST group** (optional, relevant for RUNTYP=TRUDGE)

This group specifies restart parameters for TRUDGE runs and accuracy thresholds.

KSTART indicates the conjugate gradient direction in which the optimization will proceed. ( default = -1 )

-1 .... indicates that this is a non-restart run.  
0 .... corresponds to a restart run.

FNOISE accuracy of function values.  
Variation smaller than FNOISE are not considered to be significant (Def. 0.0005)

TOLF accuracy required of the function (Def. 0.001)

TOLR accuracy required of conjugate directions (Def. 0.05)

For geometry optimization, the values which give better results (closer to the ones obtained with gradient methods) are: TOLF=0.0001, TOLR=0.001, FNOISE=0.00001

=====

## **\$FORCE group**

(optional, relevant for RUNTYP=HESSIAN,OPTIMIZE,SADPOINT)

This group controls the computation of the hessian matrix (the energy second derivative tensor, also known as the force constant matrix), and an optional harmonic vibrational analysis. This can be a very time consuming calculation. However, given the force constant matrix, the vibrational analysis for an isotopically substituted molecule is very cheap. Related input is HESS= in \$STATPT, and the \$MASS, \$HESS, \$GRAD, \$DIPDR, \$VIB groups.

METHOD = chooses the computational method.  
= ANALYTIC is implemented only for SCFTYPs RHF, ROHF, and GVB (when NPAIR is 0 or 1). This is the default for these cases.  
= NUMERIC is the default for all other cases: UHF, MCSCF, and all MP2 or CI runs.

RDHESS = a flag to read the hessian from a \$HESS group, rather than computing it. This variable pertains only to RUNTYP=HESSIAN. See also HESS= in the \$STATPT group. (default is .FALSE.)

PURIFY = controls cleanup  
Given a \$ZMAT, the hessian and dipole derivative tensor can be "purified" by transforming from

Cartesians to internals and back to Cartesians.  
This effectively zeros the frequencies of the  
translation and rotation "modes", along with  
their IR intensities. The purified quantities  
are punched out. Purification does change the  
Hessian slightly, frequencies at a stationary  
point can change by a wave number or so. The  
change is bigger at non-stationary points.  
(default=.FALSE. if \$ZMAT is given)

PRTIFC = prints the internal coordinate force constants.  
You MUST have defined a \$ZMAT group to use this.  
(Default=.FALSE.)

--- the next four apply only to METHOD=NUMERIC ----

NVIB = Number of displacements in each Cartesian  
direction for force field computation.  
= 1 Move one VIBSIZ unit in each positive  
Cartesian direction. This requires 3N+1  
evaluations of the wavefunction, energy, and  
gradient, where N is the number of SYMMETRY  
UNIQUE atoms given in \$DATA. (default)  
= 2 Move one VIBSIZ unit in the positive direction  
and one VIBSIZ unit in the negative direction.  
This requires 6N+1 evaluations of the  
wavefunction and gradient, and gives a small  
improvement in accuracy. In particular, the  
frequencies will change from NVIB=1 results by  
no more than 10-100 wavenumbers, and usually  
much less. However, the normal modes will be  
more nearly symmetry adapted, and the residual  
rotational and translational "frequencies"  
will be much closer to zero.

VIBSIZ = Displacement size (in Bohrs). Default=0.01

Let 0 mean the Vib0 geometry, and  
D mean all the displaced geometries

NPRT = 1 Print orbitals at 0 and D  
= 0 Print orbitals at 0 only (default)

NPUN = 2 Punch all orbitals at 0 and D  
= 1 Punch all orbitals at 0 and occupied orbs at D  
= 0 Punch all orbitals at 0 only (default)

----- the rest control normal coordinate analysis -----

VIBANL = flag to activate vibrational analysis.  
(the default is .TRUE. for RUNTYP=HESSIAN, and  
otherwise is .FALSE.)

SCLFAC = scale factor for vibrational frequencies, used in calculating the zero point vibrational energy. Some workers correct for the usual overestimate in SCF frequencies by a factor 0.89. ZPE or other methods might employ other factors, see A.P.Scott, L.Radom J.Phys.Chem. 100, 16502-16513 (1996). The output always prints unscaled frequencies, so this value is used only during the thermochemical analysis. (Default is 1.0)

TEMP = an array of up to ten temperatures at which the thermochemistry should be printed out. The default is a single temperature, 298.15 K. To use absolute zero, input 0.001 degrees.

FREQ = an array of vibrational frequencies. If the frequencies are given here, the hessian matrix is not computed or read. You enter any imaginary frequencies as negative numbers, omit the zero frequencies corresponding to translation and rotation, and enter all true vibrational frequencies. Thermodynamic properties will be printed, nothing else is done by the run.

PRTSCN = flag to print contribution of each vibrational mode to the entropy. (Default is .FALSE.)

DECOMP = activates internal coordinate analysis. Vibrational frequencies will be decomposed into "intrinsic frequencies", by the method of J.A.Boatz and M.S.Gordon, J.Phys.Chem., 93, 1819-1826(1989). If set .TRUE., the \$ZMAT group may define more than 3N-6 (3N-5) coordinates. (default=.FALSE.)

PROJCT = controls the projection of the hessian matrix. The projection technique is described by W.H.Miller, N.C.Handy, J.E.Adams in J. Chem. Phys. 1980, 72, 99-112. At stationary points, the projection simply eliminates rotational and translational contaminants. At points with non-zero gradients, the projection also ensures that one of the vibrational modes will point along the gradient, so that there are a total of 7 zero frequencies. The other 3N-7 modes are constrained to be orthogonal to the gradient. Because the projection has such a large effect on the hessian, the hessian punched is the one BEFORE projection. For the same reason, the default is .FALSE. to skip the projection, which is mainly of interest in dynamical calculations.

=====

There is a set of programs for the calculation of kinetic or equilibrium isotope effects from the group of Piotr Paneth at the University of Lodz. This ISOEFF package will accept data computed by GAMESS, and can be downloaded at <http://ck-sg.p.lodz.pl/isoeff/isoeff.html>

=====

### **\$CPHF group** (relevant for analytic RUNTYP=HESSIAN)

This group controls the solution of the response equations, also known as coupled Hartree-Fock.

POLAR = a flag to request computation of the static polarizability, alpha. Because this property needs 3 additional response vectors, beyond those needed for the hessian, the default is to skip the property. (default = .FALSE.)

NWORD = controls memory usage for this step. The default uses all available memory. (default=0)

MXCPIT = When solving **CPHF** equations iteratively, **MXCPIT** controls the maximum allowed number of iterations. The default value is 50.

CPTOL = When solving **CPHF** equations iteratively, **CPTOL** controls the maximum allowed relative error of the approximate solution. Default value is  $10^{-5}$

=====

### **\$HESS group** (relevant for RUNTYP=HESSIAN if RDHESS=.TRUE.) (relevant for RUNTYP=IRC if FREQ,CMODE not given) (relevant for RUNTYP=OPTIMIZE,SADPOINT if HESS=READ)

Formatted force constant matrix (FCM), i.e. hessian matrix. This data is punched out by a RUNTYP=HESSIAN job, in the correct format for subsequent runs. The first card in the group must be a title card.

A \$HESS group is always punched in Cartesians. It will be transformed into internal coordinate space if a

geometry search uses internals. It will be mass weighted (according to \$MASS) for IRC and frequency runs.

The initial FCM is updated during the course of a geometry optimization or saddle point search, and will be punched if a run exhausts its time limit. This allows restarts where the job leaves off. You may want to read this FCM back into the program for your restart, or you may prefer to regenerate a new initial hessian. In any case, this updated hessian is absolutely not suitable for frequency prediction!



=====  
**\$GRAD group** (relevant for RUNTYP=OPTIMIZE or SADPOINT)  
(relevant for RUNTYP=HESSIAN when RDHESS=.TRUE.)

Formatted gradient vector at the \$DATA geometry. This data is read in the same format it was punched out.

For RUNTYP=HESSIAN, this information is used to determine if you are at a stationary point, and possibly for projection. If omitted, the program pretends the gradient is zero, and otherwise proceeds normally.

For geometry searches, this information (if known) can be read into the program so that the first step can be taken instantly.

=====  
**\$DIPDR group** (relevant for RUNTYP=HESSIAN if RDHESS=.T.)

Formatted dipole derivative tensor, punched in a previous RUNTYP=HESSIAN job. If this group is omitted, then a vibrational analysis will be unable to predict the IR intensities, but the run can otherwise proceed.

=====  
**\$VIB group** (relevant for RUNTYP=HESSIAN, METHOD=NUMERIC)

Formatted card image -restart- data. This data is read in the format it was punched by a previous HESSIAN job to the file IRCDATA. Just add a "\$END" card, and if the final gradient was punched as zero, delete the last set of data. Normally, IREST in \$CONTRL will NOT be used in conjunction with a HESSIAN restart. The mere presence of this deck triggers the restart from cards. This deck can also be used to turn a single point differencing run into double differencing, as well as recovering from time limits, or other bombouts.

=====  
**\$MASS group** (relevant for RUNTYP=HESSIAN, IRC, or DRC)

This group permits isotopic substitution during the computation of mass weighted Cartesian coordinates. Of course, the masses affect the frequencies and normal modes of vibration.

AMASS = An array giving the atomic masses, in amu. The default is to use the mass of the most abundant

isotope. Masses through element 104 are stored.

example - \$MASS AMASS(3)=2.0140 \$END

will make the third atom in the molecule a deuterium.

=====

## **\$IRC group**

(relevant for RUNTYP=IRC)

This group governs the location of the intrinsic reaction coordinate, a steepest descent path in mass weighted coordinates, that connects the saddle point to reactants and products.

----- there are five integration methods chosen by PACE.

PACE = GS2 selects the Gonzalez-Schlegel second order method. This is the default method.  
Related input is:

GCUT cutoff for the norm of the mass-weighted gradient tangent (the default is chosen in the range from 0.00005 to 0.00020, depending on the value for STRIDE chosen below.)  
RCUT cutoff for Cartesian RMS displacement vector. (the default is chosen in the range 0.0005 to 0.0020 Bohr, depending on the value for STRIDE)  
ACUT maximum angle from end points for linear interpolation (default=5 degrees)  
MXOPT maximum number of constrained optimization steps for each IRC point (default=20)  
IHUPD is the hessian update formula. 1 means Powell, 2 means BFGS (default=2)  
GA is a gradient from the previous IRC point, and is used when restarting.  
OPTTOL is a gradient cutoff used to determine if the IRC is approaching a minimum. It has the same meaning as the variable in \$STATPT. (default=0.0001)

PACE = LINEAR selects linear gradient following (Euler's method). Related input is:

STABLZ switches on Ishida/Morokuma/Komornicki reaction path stabilization. The default is .TRUE.  
DELTA initial step size along the unit bisector, if STABLZ is on. Default=0.025 Bohr.  
ELBOW is the collinearity threshold above which the stabilization is skipped. If the mass weighted gradients at QB and QC are almost collinear, the reaction path is deemed to be curving very little, and stabilization isn't needed. The default is 175.0 degrees. To always perform stabilization, input 180.0.  
READQB, EB, GBNORM, GB are energy and gradient data already known at the current IRC point. If it

happens that a run with STABLZ on decides to skip stabilization because of ELBOW, this data will be punched to speed the restart.

PACE = QUAD selects quadratic gradient following.  
Related input is:

SAB distance to previous point on the IRC.  
GA gradient vector at that historical point.

PACE = AMPC4 selects the fourth order Adams-Moulton variable step predictor-corrector.  
Related input is:

GA0,GA1,GA2 which are gradients at previous points.

PACE = RK4 selects the 4th order Runge-Kutta variable step method. There is no related input.

----- The next two are used by all PACE choices -----

STRIDE = Determines how far apart points on the reaction path will be. STRIDE is used to calculate the step taken, according to the PACE you choose. The default is good for the GS2 method, which is very robust. Other methods should request much smaller step sizes, such as 0.10 or even 0.05.  
(default = 0.30 sqrt(amu)-Bohr)

NPOINT = The number of IRC points to be located in this run. The default is to find only the next point.  
(default = 1)

----- The next two let you choose your output volume -----

Let F mean the first IRC point found in this run,  
and L mean the final IRC point of this run.  
Let INTR mean the internuclear distance matrix.

NPRT = 1 Print INTR at all, orbitals at all IRC points  
0 Print INTR at all, orbitals at F+L (default)  
-1 Print INTR at all, orbitals never  
-2 Print INTR at F+L, orbitals never

NPUN = 1 Punch all orbitals at all IRC points  
0 Punch all orbitals at F+L, only occupied orbitals at IRC points between (default)  
-1 Punch all orbitals at F+L only  
-2 Never punch orbitals

----- The next two tally the reaction path results. The defaults are appropriate for starting from a saddle point, restart values are automatically punched out.

NEXTPT = The number of the next point to be computed.  
STOTAL = Total distance along the reaction path to next IRC point, in mass weighted Cartesian space.

----- The following controls jumping off the saddle point. If you give a \$HESS group, FREQ and CMODE will be generated automatically.

SADDLE = A logical variable telling if the coordinates given in the \$DATA deck are at a saddle point (.TRUE.) or some other point lying on the IRC (.FALSE.). If SADDLE is true, either a \$HESS group or else FREQ and CMODE must be given. (default = .FALSE.) Related input is:

TSENGY = A logical variable controlling whether the energy and wavefunction are evaluated at the transition state coordinates given in \$DATA. Since you already know the energy from the transition state search and force field runs, the default is .F.

FORWRD = A logical variable controlling the direction to proceed away from a saddle point. The forward direction is defined as the direction in which the largest magnitude component of the imaginary normal mode is positive. (default = .TRUE.)

EVIB = Desired decrease in energy when following the imaginary normal mode away from a saddle point. (default=0.0005 Hartree)

FREQ = The magnitude of the imaginary frequency, given in  $\text{cm}^{-1}$ .

CMODE = An array of the components of the normal mode whose frequency is imaginary, in Cartesian coordinates. Be careful with the signs!

You must give FREQ and CMODE if you don't give a \$HESS group, when SADDLE=.TRUE. The option of giving these two variables instead of a \$HESS does not apply to the GS2 method, which must have a hessian input, even for restarts. Note also that EVIB is ignored by GS2 runs.

=====

\* \* \* \* \*  
For hints about IRC tracking, see  
the 'further information' section.  
\* \* \* \* \*

=====  
**\$GRADEX group** (optional, for RUNTYP=GRADEXTR)

This group controls the gradient extremal following algorithm. The GEs leave stationary points parallel to each of the normal modes of the hessian. Sometimes a GE leaving a minimum will find a transition state, and thus provides us with a way of finding that saddle point. GEs have many unusual mathematical properties, and you should be aware that they normally differ a great deal from IRCs.

The search will always be performed in cartesian coordinates, but internal coordinates along the way may be printed by the usual specification of NZVAR and \$ZMAT.

METHOD = algorithm selection.

SR A predictor-corrector method due to Sun and Ruedenberg (default).

JJH A method due to Jorgensen, Jensen and Helgaker.

NSTEP = maximum number of predictor steps to take.

(default=50)

DPRED = the stepsize for the predictor step.

(default = 0.10)

STPT = a flag to indicate whether the initial geometry is considered a stationary point. If .TRUE., the geometry will be perturbed by STSTEP along the IFOLLOW normal mode.

(default = .TRUE.)

STSTEP = the stepsize for jumping away from a stationary point. (default = 0.01)

IFOLLOW = Mode selection option. (default is 1)

If STPT=.TRUE., the initial geometry will be perturbed by STSTEP along the IFOLLOW normal mode. Note that IFOLLOW can be positive or negative, depending on the direction the normal mode should be followed in. The positive direction is defined as the one where the largest component of the Hessian eigenvector is positive.

If STPT=.FALSE. the sign of IFOLLOW determines which direction the GE is followed in. A positive value will follow the GE in the uphill direction. The value of IFOLLOW should be set to the Hessian mode which is parallel to the gradient to avoid miscellaneous warning messages.

GOFIRST = a flag to indicate whether the algorithm should attempt to locate a stationary point. If .TRUE., a straight NR search is performed once the NR step length drops below SNRMAX. 10 NR steps are then allowed, a value which cannot be changed. (default = .TRUE.)

SNRMAX = upper limit for switching to straight NR search for stationary point location. (default = 0.10 or DPRED, whichever is smallest)

OPTTOL = gradient convergence tolerance, in Hartree/Bohr. Used for optimizing to a stationary point. Convergence of a geometry search requires the

rms gradient to be less than OPTTOL.  
(default=0.0001)

HESS = selection of the initial hessian matrix, if STPT=.TRUE.  
= READ causes the hessian to be read from a \$HESS group.  
= CALC causes the hessian to be computed. (default)

---- parameters on this page apply only to METHOD=SR ----

DELCOR = the corrector step should be smaller than this value before the next predictor step is taken. (default = 0.001)

MYSTEP = maximum number of micro iteration allowed to bring the corrector step length below DELCOR. (default=20)

SNUMH = stepsize used in the numerical differentiation of the Hessian to produce third derivatives. (default = 0.0001)

HSDFDB = flag to select determination of third derivatives. At the current geometry we need the gradient, the Hessian, and the partial third derivative matrix in the gradient direction.

If .TRUE., the gradient is calculated at the current geometry, and two Hessians are calculated at SNUMH distance to each side in the gradient direction. The Hessian at the geometry is formed as the average of the two displaced Hessians.

If .FALSE., both the gradient and Hessian are calculated at the current geometry, and one additional Hessian is calculated at SNUMH in the gradient direction.

The default double-sided differentiation produces a more accurate third derivative matrix, at the cost of an additional wave function and gradient. (default = .TRUE.)

\* \* \* \* \*

See the 'further information' section for some help with GRADEXTR runs.  
\* \* \* \* \*

=====

### **\$DRC group**

(relevant for RUNTYP=DRC)

This group governs the dynamical reaction coordinate, a classical trajectory method based on quantum chemical potential energy surfaces. In GAMESS these may be either ab initio or semi-empirical. Because the vibrational period of a normal mode with frequency 500 wavenumbers is 67 fs, a DRC needs to run for many steps in order to sample a representative portion of phase space. Almost all DRCs break molecular symmetry, so build your molecule with C1 symmetry in \$DATA, or specify NOSYM=1 in \$CONTRL. Restart data can be found in the job's OUTPUT file, with important results summarized to the IRCDATA file.

NSTEP = The number of DRC points to be calculated, not including the initial point. (default = 1000)

DELTAT = is the time step. (default = 0.1 fs)

TOTIME = total duration of the DRC computed in a previous job, in fs. The default is the correct value when initiating a DRC. (default=0.0 fs)

\* \* \*

In general, a DRC can be initiated anywhere, so \$DATA might contain coordinates of the equilibrium geometry, or a nearby transition state, or something else. You must also supply an initial kinetic energy, and the direction of the initial velocity, for which there are a number of options:

EKIN = The initial kinetic energy (default = 0.0 kcal/mol)  
See also ENM, NVEL, and VIBLVL regarding alternate ways to specify the initial value.

VEL = an array of velocity components, in Bohr/fs.  
When NVEL is false, this is simply the direction of the velocity vector. Its magnitude will be

automatically adjusted to match the desired initial kinetic energy, and it will be projected so that the translation of the center of mass is removed. Give in the order vx1, vy1, vz1, vx2, vy2, ...

NVEL = a flag to compute the initial kinetic energy from the input VEL using the sum of mass\*VEL\*VEL/2. This flag is usually selected only for restarts. (default=.FALSE.)

\* \* \*

The next two allow the kinetic energy to be partitioned over all normal modes. The coordinates in \$DATA are likely to be from a stationary point! You must also supply a \$HESS group.

VIBLVL = a flag to turn this option on (default=.FALSE.)

VIBENG = an array of energies (in units of multiples of the hv of each mode) to be imparted along each normal mode. The default is to assign the zero point energy only, VIBENG(1)=0.5, 0.5, ..., 0.5. If given as a negative number, the initial direction of the velocity vector is along the reverse direction of the mode. "Reverse" means the phase of the normal mode is chosen such that the largest magnitude component is a negative value. An example might be VIBENG(4)=2.5 to add two quanta to mode 4, along with zero point energy in all modes.

\* \* \*

The next three pertain to initiating the DRC along a single normal mode of vibration. No kinetic energy is assigned to the other modes. You must also supply a \$HESS group.

NNM = The number of the normal mode to which the initial kinetic energy is given. The absolute value of NNM must be in the range 1, 2, ..., 3N-6. If NNM is a positive/negative value, the initial velocity will lie in the forward/reverse direction of the mode. "Forward" means the largest component of the normal mode is a positive value. (default=0)

ENM = the initial kinetic energy given to mode NNM, in units of vibrational quanta hv, so the amount depends on mode NNM's vibrational frequency, v. If you prefer to impart an arbitrary initial kinetic energy to mode NNM, specify EKIN instead. (default = 0.0 quanta)



\* \* \*

To summarize, there are five different ways to specify the DRC trajectory:

1. VEL vector with NVEL=.TRUE. This is difficult to specify at your initial point, and so this option is mainly used when restarting your trajectory. The restart information is always in this format.
2. VEL vector and EKIN with NVEL=.FALSE. This will

give a desired amount of kinetic energy in the direction of the velocity vector.

3. VIBLVL and VIBENG selected, to give initial kinetic energy to all of the normal modes.
4. NNM and ENM to give quanta to a single normal mode.
5. NNM and EKIN to give arbitrary kinetic energy to a single normal mode.

\* \* \*

The most common use of the next two is to analyze a trajectory with respect to the minimum energy geometry the trajectory is traveling around.

NMANAL = a flag to select mapping of the mass-weighted Cartesian DRC coordinates and velocity (conjugate momentum) in terms of normal modes step by step. If you choose this option, you must supply both C0 and a \$HESS group from the stationary point. (default=.FALSE.)

C0 = an array of the coordinates of the stationary point (the coordinates in \$DATA might well be some other coordinates). Give in the order x1,y1,z1,x2,y2,...

\* \* \*

The next option applies to all input paths which read a hessian: NMANAL, NNM, or VIBLVL. After the translations and rotations have been dropped, the normal modes are renumbered 1, 2, ..., 3N-6.

HESSTS = a flag to say if the hessian corresponds to a transition state or a minimum. This parameter controls deletion of the translation and rotation degrees of freedom, i.e. the default is to drop the first six "modes", while setting this flag on drops modes 2 to 7 instead. (default=.FALSE.)

\* \* \*

The final variables control the volume of output.  
Let F mean the first DRC point found in this run,  
and L mean the last DRC point of this run.

NPRTSM = summarize the DRC results every NPRTSM steps,  
to the file IRCDATA. (default = 1)

NPRT = 1 Print orbitals at all DRC points  
0 Print orbitals at F+L (default)  
-1 Never print orbitals

NPUN = 2 Punch all orbitals at all DRC points  
1 Punch all orbitals at F+L, and occupied  
orbitals at DRC points between  
0 Punch all orbitals at F+L only (default)  
-1 Never punch orbitals

=====

#### References:

J.J.P.Stewart, L.P.Davis, L.W.Burggraf,

J.Comput.Chem. 8, 1117-1123 (1987)

S.A.Maluendes, M.Dupuis, J.Chem.Phys. 93, 5902-5911 (1990)

T.Taketsugu, M.S.Gordon, J.Phys.Chem. 99, 8462-8471 (1995)

T.Taketsugu, M.S.Gordon, J.Phys.Chem. 99, 14597-604 (1995)

T.Taketsugu, M.S.Gordon, J.Chem.Phys. 103, 10042-9 (1995)

T.Taketsugu, M.S.Gordon, J.Chem.Phys. 104, 2834-40 (1996)

M.S.Gordon, G.Chaban, T.Taketsugu  
J.Phys.Chem. 100, 11512-11525 (1996)

=====

#### **\$SURF group** (relevant for RUNTYP=SURFACE and RUNTYP=RSURFACE)

This group allows you to probe a potential energy  
surface along a small grid of points. Note that there is  
no option to vary angles, only distances. The scan can  
be made for any SCFTYP, or for the MP2 or CI surface.

IVEC1 = an array of two atoms, defining a coordinate from

the first atom given, to the second.

IGRP1 = an array specifying a group of atoms, which must include the second atom given in IVEC1. The entire group will be translated (rigidly) along the vector IVEC1, relative to the first atom given in IVEC1.

ORIG1 = starting value of the coordinate, which may be positive or negative. Zero corresponds to the distance given in \$DATA.

DISP1 = step size for the coordinate.

NDISP1 = number of steps to take for this coordinate.

There are no reasonable defaults for these keywords, so you should input all of them. ORIG1 and DISP1 should be given in Angstrom.

IVEC2, IGRP2, ORIG2, DISP2, NDISP2 = have the identical meaning as their "1" counterparts, and permit you to make a two dimensional map along two displacement coordinates. If the "2" data are not input, the surface map proceeds in only one dimension.

Note that properties are not computed at these points, other than the energy.

REUSE = .t. reuses approximate hessian information during relaxed scans. Failure to add this keyword may compromise the completion of your relaxed scan job.

NORMV is seldom used and means to renorm vectors defined via VECT1 or VECT2 to unity.

NSURF controls multiple PES construction for multistate jobs like state-averaged MCSCF.

VECT1 and  
VECT2 are double precision arrays containing coefficients of the Cartesian or internal coordinates forming displacements to scan, with the numbering scheme the same as of IFREEZ array in \$STATPT. The use of these keywords allows the programs to scan angles or dihedrals, rather than distances. VECT1 (and VECT2) runs should be used in conjunction with DISP<sub>i</sub> and NDISP<sub>i</sub>. ORIG<sub>i</sub> are also allowed. RSURFACE runs should use ifreez in \$STATPT to manually freeze the same Cartesian/internal coordinates or their combinations which are selected to scan in \$surf via VECT1/VECT2. This is not required for non-relaxed surface runs.

=====

**\$LOCAL group** (relevant for LOCAL=RUEDNBRG, BOYS, or POP)

This group allows input of additional data to control the localization methods. If no input is provided, the valence orbitals will be localized as much as possible, while still leaving the wavefunction invariant.

PRTLLOC = a flag to control supplemental printout. The extra output is the rotation matrix to the localized orbitals, and, for the Boys method, the orbital centroids, for the Ruedenberg method, the coulomb and exchange matrices, for the population method, atomic populations. (default=.FALSE.)

MAXLOC = maximum number of localization cycles. This applies to BOYS or POP methods only. If the localization fails to converge, a different order of 2x2 pairwise rotations will be tried. (default=250)

CVGLOC = convergence criterion. The default provides LMO coefficients accurate to 6 figures. (default=1.0E-6)

SYMLOC = a flag to restrict localization so that orbitals of different symmetry types are not mixed. This option is not supported in all possible point groups. The purpose of this option is to give a better choice for the starting orbitals for GVB-PP or MCSCF runs, without destroying the orbital's symmetry. This option is compatible with each of the 3 methods of selecting the orbitals to be included. (default=.FALSE.)

These parameters select the orbitals which are to be included in the localization. You may select from FCORE, NOUTA/NOUTB, or NINA/NINB, but may choose only one of these.

FCORE = flag to freeze all the chemical core orbitals present. All the valence orbitals will be localized. (default=.TRUE.)

\* \* \*

NOUTA = number of alpha orbitals to hold fixed in the localization. (default=0)

MOOUTA = an array of NOUTA elements giving the numbers of the orbitals to hold fixed. For example, the input NOUTA=2 MOOUTA(1)=8,13 will freeze only orbitals 8 and 13. You must enter all the

orbitals you want to freeze, including any cores.  
This variable has nothing to do with cows.

NOUTB = number of beta orbitals to hold fixed in -UHF-  
localizations. (default=0)

MOOUTB = same as MOOUTA, except that it applies to the  
beta orbitals, in -UHF- wavefunctions only.

\* \* \*

NINA = number of alpha orbitals which are to be  
included in the localization. (default=0)

MOINA = an array of NINA elements giving the numbers of  
the orbitals to be included in the localization.  
Any orbitals not mentioned will be frozen.

NINB = number of -UHF- beta MOs in the localization.  
(default=0)

MOINB = same as MOINA, except that it applies to the  
beta orbitals, in -UHF- wavefunctions only.

N.B. Since Boys localization needs the dipole integrals,  
do not turn off dipole moment calculation in \$ELMOM.

----- The following keywords are used for the localized  
charge distribution (LCD) energy decomposition.

EDCOMP = flag to turn on LCD energy decomposition.  
Note that this method is currently implemented  
for SCFTYP=RHF and ROHF and LOCAL=RUEDNBRG only.  
The SCF LCD forces all orbitals to be localized,  
overriding input on the previous page. See also  
LMOMP2 in the \$MP2 group. (default = .FALSE.)

MOIDON = flag to turn on LMO identification and subsequent  
LMO reordering, and assign nuclear LCD automat-  
ically. (default = .FALSE.)

DIPDCM = flag for LCD molecular dipole decomposition.  
(default = .FALSE.)

QADDCM = flag for LCD molecular quadrupole decomposition.  
(default = .FALSE.)

POLDPCM = flag to turn on LCD polarizability decomposition.  
This method is implemented for SCFTYP=RHF or ROHF  
and LOCAL=BOYS or RUEDNBRG. (default=.FALSE.)

POLANG = flag to choose units of localized polarizability  
output. The default is Angstroms\*\*3, while false  
will give Bohr\*\*3. (default=.TRUE.)

ZDO = flag for LCD analysis of a composite wave function,

given in a \$VEC group of a van der Waals complex, within the zero differential overlap approximation. The MOs are not orthonormalized and the intermolecular electron exchange energy is neglected. In addition, the molecular overlap matrix is printed out. This is a very specialized option. (default = .FALSE.)

----- The remaining keywords can be used to define the nuclear part of an LCD. They are usually used to rectify mistakes in the automatic definition made when MOIDON=.TRUE. The index defining the LMO number then refers to the reordered list of LMOs.

NNUCMO = array giving the number of nuclei assigned to a particular LMO.

IJMO = is an array of pairs of indices (I,J), giving the row (nucleus I) and column (orbital J) index of the entries in ZIJ and MOIJ.

MOIJ = arrays of integers K, assigning nucleus K as the site of the Ith charge of LCD J.

ZIJ = array of floating point numbers assigning a charge to the Ith charge of LCD J.

IPROT = array of integers K, defining nucleus K as a proton.

DEPRNT = a flag for additional decomposition printing, such as pair contributions to various energy terms, and centroids of the Ruedenberg orbitals. (default = .FALSE.)

=====

\* \* \* \* \*  
For hints about localizations, and  
the LCD energy decomposition, see  
the 'further information' section.  
\* \* \* \* \*

=====

**\$TWOEI group** (relevant for EDCOMP=.TRUE. in \$LOCAL)

Formatted transformed two-electron Coulomb and Exchange integrals as punched during a LOCAL=RUEDNBRG run. If this group is present it will automaticall be read in during such a run and the two-electron integrals do not have to be re-transformed. This group is especially useful for

EDCOMP=.TRUE. runs when the localization has to be repeated for different definitions of nuclear LCDs.

=====

**\$ELMOM group** (not required)

This group controls electrostatic moments calculation.

IEMOM = 0 - skip this property  
1 - calculate monopole and dipole (default)  
2 - also calculate quadrupole moments  
3 - also calculate octupole moments

WHERE = COMASS - center of mass (default)  
NUCLEI - at each nucleus  
POINTS - at points given in \$POINTS.

OUTPUT = PUNCH, PAPER, or BOTH (default)

IEMINT = 0 - skip printing of integrals (default)  
1 - print dipole integrals  
2 - also print quadrupole integrals  
3 - also print octupole integrals  
-2 - print quadrupole integrals only  
-3 - print octupole integrals only

The quadrupole and octupole tensors on the printout are formed according to the definition of Buckingham. Caution: only the first nonvanishing term in the multipole charge expansion is independent of the coordinate origin chosen, which is normally the center of mass.

=====

**\$ELPOT group** (not required)

This group controls electrostatic potential calculation.

IEPOT = 0 skip this property (default)  
1 calculate electric potential

WHERE = COMASS - center of mass  
NUCLEI - at each nucleus (default)  
POINTS - at points given in \$POINTS  
GRID - at grid given in \$GRID  
PDC - at points controlled by \$PDC.  
CUBE - grid given in \$CUBE. This is the default if \$CUBE is given

OUTPUT = PUNCH, PAPER, or BOTH (default)

This property is the electrostatic potential  $V(a)$  felt by a test positive charge, due to the molecular charge density. A nucleus at the evaluation point is ignored. If this property is evaluated at the nuclei, it obeys the

equation

$$\sum \text{on nuclei (a)} \quad Z(a)*V(a) = 2*V(nn) + V(ne).$$

The electronic portion of this property is called the diamagnetic shielding.

=====  
**\$ELDENS group** (not required)

This group controls electron density calculation.

IEDEN = 0 skip this property (default)  
      = 1 compute the electron density.

MORB = The molecular orbital whose electron density is  
      to be computed. If zero, the total density is  
      computed. (default=0)

WHERE = COMASS - center of mass  
      NUCLEI - at each nucleus (default)  
      POINTS - at points given in \$POINTS  
      GRID - at grid given in \$GRID  
      CUBE - grid given in \$CUBE. This is the default if  
           \$CUBE is given

OUTPUT = PUNCH, PAPER, or BOTH (default)

IEDINT = 0 - skip printing of integrals (default)  
      1 - print the electron density integrals

SPIND =.f.(default)/.t. - calculate spin density if available

DIFFD =.f.(default)/.t. - calculate delta density

SKIPHF=.t.(default)/.f. - skip (HF - initial guess) delta density  
                          calculation

DERIVS(1)=0,0,0 (default) - any combination of up to three numbers 0,1 &  
                          2. Defines the level of derivatives  
                          required. 0 means density, 1 means the  
                          density gradient (or its norm), 2 - density  
                          Laplacian.

GRDFLD= .f.(default)/.t. - modifies density gradient calculations so  
                          that the 3-component vector gradient field is  
                          written (by default, the norm of the gradient  
                          is written).

ICORBS - integer array - indices of alpha MOs (positive) and (if  
          available) beta MOs/NOs (negative) for which to calculate the  
          values of orbitals (not the density!). Most of other \$cube-  
          related options are disabled in the presence of ICORBS array.  
          In the punch file, the inner loop of the printout is over  
          different MOs, then over z coordinate, etc...



UHFNOS= .f.(default)/.t. modifies icorbs array entries so that negative entries are treated as requests to calculate UHF NOs, not UHF beta orbitals.

=====  
***\$ELFLDG group*** (not required)

This group controls electrostatic field and electric field gradient calculation.

IEFLD = 0 - skip this property (default)  
1 - calculate field  
2 - calculate field and gradient

WHERE = COMASS - center of mass  
NUCLEI - at each nucleus (default)  
POINTS - at points given in \$POINTS

OUTPUT = PUNCH, PAPER, or BOTH (default)

IEFINT = 0 - skip printing these integrals (default)  
1 - print electric field integrals  
2 - also print field gradient integrals  
-2 - print field gradient integrals only

The Hellman-Feynman force on a nucleus is the nuclear charge multiplied by the electric field at that nucleus. The electric field is the gradient of the electric potential, and the field gradient is the hessian of the electric potential. The components of the electric field gradient tensor are formed in the conventional way, i.e. see D.Neumann and J.W.Moskowitz.

=====  
***\$POINTS group*** (not required)

This group is used to input points at which properties will be computed. This first card in the group must contain the string ANGS or BOHR, followed by an integer NPOINT, the number of points to be used. The next NPOINT cards are read in free format, containing the X, Y, and Z coordinates of each desired point.

=====  
***\$GRID group*** (not required)

This group is used to input a grid (plane through the molecule) on which properties will be calculated.

ORIGIN(i) = coordinates of the lower left corner of  
the plot.  
XVEC(i) = coordinates of the lower right corner of  
the plot.  
YVEC(i) = coordinates of the upper left corner of  
the plot.  
SIZE = grid increment, default is 0.25.  
UNITS = units of the above four values, it can be  
either BOHR or ANGS (the default).

Note that XVEC and YVEC are not necessarily parallel to  
the X and Y axes, rather they are the axes which you  
desire to see plotted by the MEPMAP contouring program.

=====

\* \* \* \* \*  
For conversion factors, and references  
see the 'further information' section.  
\* \* \* \* \*

=====

## **\$PDC group**

(relevant if WHERE=PDC in \$ELPOT)

This group determines the points at which to compute  
the electrostatic potential, for the purpose of fitting  
atomic charges to this potential. Constraints on the fit  
which determines these "potential determined charges" can  
include the conservation of charge, the dipole, and the  
quadrupole.

PTSEL = determines the points to be used, choose from  
    GEODESIC to use a set of points on several fused  
          sphere van der Waals surfaces, with points  
          selected using an algorithm due to Mark  
          Spackman. The results are similar to those  
          from the Kollman/Singh method, but are  
          less rotation dependent. (default)  
    CONNOLLY to use a set of points on several fused  
          sphere van der Waals surfaces, with points  
          selected using an algorithm due to Michael  
          Connolly. This is identical to the method  
          used by Kollman & Singh (see below)  
    CHELPG to use a modified version of the CHELPG  
          algorithm, which produces a symmetric  
          grid of points for a symmetric molecule.

CONSTR = NONE - no fit is performed. The potential at  
          the points is instead output according  
          to OUTPUT in \$ELPOT.

CHARGE - the sum of fitted atomic charges is constrained to reproduce the total molecular charge. (default)  
DIPOLE - fitted charges are constrained to exactly reproduce the total charge and dipole.  
QUPOLE - fitted charges are constrained to exactly reproduce the charge, dipole, and quadrupole.

Note: the number of constraints cannot exceed the number of parameters, which is the number of nuclei. Planar molecules afford fewer constraint equations, namely two dipole constraints and three quadrupole constraints, instead of three and five, respectively.

\* \* \* the next 5 pertain to PTSEL=GEODESIC or CONNOLLY \* \* \*

VDWSCL = scale factor for the first shell of VDW spheres. The default of 1.4 seems to be an empirical best value. Values for VDW radii for most elements up to Z=36 are internally stored.

VDWINC = increment for successive shells (default = 0.2). The defaults for VDWSCL and VDWINC will result in points chosen on layers at 1.4, 1.6, 1.8 etc times the VDW radii of the atoms.

LAYER = number of layers of points chosen on successive fused sphere VDW surfaces (default = 4)

NFREQ = flag for particular geodesic tessellation of points. Only relevant if PTSEL=GEODESIC.  
Options are:  
(10\*h + k) for {3,5}h,k tessellations  
-(10\*h + k) for {5,3}h,k tessellations  
(of course both nh and nk must be less than 10, so NFREQ must lie within the range -99 to 99)  
The default value is NFREQ=30 (=03)

PTDENS = density of points on the surface of each scaled VDW sphere (in points per square au). Only relevant if PTSEL=CONNOLLY. Default is 0.28 per au squared, which corresponds to 1.0 per square Angstrom, the default recommended by Kollman & Singh.

\* \* \* the next two pertain to PTSEL=CHELPG \* \* \*

RMAX = maximum distance from any point to the closest atom. (default=3.0 Angstroms)

DELR = distance between points on the grid. (default=0.8 Angstroms)

MAXPDC = an estimate of the total number of points whose electrostatic potential will be included in the

fit. (default=10000)

\* \* \*

CENTER = an array of coordinates at which the moments were computed.

DPOLE = the molecular dipole.

QPOLE = the molecular quadrupole.

PDUNIT = units for the above values. ANGS (default) will mean that the coordinates are in Angstroms, the dipole in Debye, and quadrupole in Buckingham's. BOHR implies atomic units for all 3.

Note: it is easier to compute the moments in the current run, by setting IEMOM to at least 2 in \$ELMOM. However, you could fit experimental data, for example, by reading it in here.

=====

There is no unique way to define fitted atomic charges. Smaller numbers of points at which the electrostatic potential is fit, changes in VDW radii, asymmetric point location, etc. all affect the results. A useful bibliography is

U.C.Singh, P.A.Kollman, J.Comput.Chem. 5, 129-145(1984)  
L.E.Chirlain, M.M.Francl, J.Comput.Chem. 8, 894-905(1987)  
R.J.Woods, M.Khalil, W.Pell, S.H.Moffatt, V.H.Smith,  
J.Comput.Chem. 11, 297-310(1990)  
C.M.Breneman, K.B.Wiberg, J.Comput.Chem. 11, 361-373(1990)  
K.M.Merz, J.Comput.Chem. 13, 749(1992)  
M.A.Spackman, J.Comput.Chem. 17, 1-18(1996)

=====

**\$MOLGRF group** (relevant only if you have MOLGRAPH)

This option provides an interface for viewing orbitals through a commercial package named MOLGRAPH, from Daikin Industries. Note that this option uses three disk files which are not defined in the GAMESS execution scripts we provide, since we don't use MOLGRAPH ourselves. You will need to define files 28, 29, 30, as generic names PRGRID, COGRID, MOGRID, of which the latter is passed to MOLGRAPH.

GRID3D = a flag to generate 3D grid data.  
(default is .false.).

TOTAL = a flag to generate a total density grid data.  
"Total" means the sum of the orbital densities

given by NPLT array. (default is .false.).

MESH = numbers of grids. You can use different numbers for three axes. (default is MESH(1)=21,21,21).

BOUND = boundary coordinates of a 3D graphical cell. The default is that the cell is larger than the molecular skeleton by 3 bohr in all directions. E.g., BOUND(1)=xmin,xmax,ymin,ymax,zmin,zmax

NPLOTS = number of orbitals to be used to generate 3D grid data. (default is NPLOTS=1).

NPLT = orbital IDs. The default is 1 orbital only, the HOMO or SOMO. If the LOCAL option is given in \$CONTRL, localized orbital IDs should be given. For example, NPLT(1)=n1,n2,n3,...

CHECK = debug option, printing some of the grid data.

If you are interested in graphics, look at the WWW page for information about other graphics packages with GAMESS.

=====

### **\$STONE group** (optional)

This group defines the expansion points for Stone's distributed multipole analysis (DMA) of the electrostatic potential.

The DMA takes the multipolar expansion of each overlap charge density defined by two gaussian primitives, and translates it from the center of charge of the overlap density to the nearest expansion point. Some references for the method are

Stone, Chem.Phys.Lett. 83, 233 (1981)  
Price and Stone, Chem.Phys.Lett. 98, 419 (1983)  
Buckingham and Fowler, J.Chem.Phys. 79, 6426 (1983)  
Stone and Alderton, Mol.Phys. 56, 1047 (1985)

The existence of a \$STONE group in the input is what triggers the analysis. Enter as many lines as you wish, in any order, terminated by a \$END record.

-----

ATOM i name, where

ATOM is a keyword indicating that a particular atom is selected as an expansion center.  
i is the number of the atom  
name is an optional name for the atom. If not entered the name will be set to the name

used in the \$DATA input.

-----  
ATOMS            is a keyword selecting all nuclei in the  
                 molecule as expansion points. No other  
                 input on the line is necessary.  
-----

-----  
BONDS            is a keyword selecting all bond midpoints  
                 in the molecule as expansion points. No  
                 other input on the line is necessary.  
-----

-----  
BOND i j name, where

      BOND        is a keyword indicating that a bond mid-  
                 point is selected as an expansion center.  
      i,j        are the indices of the atoms defining the  
                 bond, corresponding to two atoms in \$DATA.  
      name        is an optional name for the bond midpoint.  
                 If omitted, it is set to 'BOND'.  
-----

-----  
CMASS            is a keyword selecting the center of mass  
                 as an expansion point. No other input on  
                 the line is necessary.  
-----

-----  
POINT x y z name, where

      POINT       is a keyword indicating that an arbitrary  
                 point is selected as an expansion point.  
      x,y,z      are the coordinates of the point, in Bohr.  
      name        is an optional name for the expansion  
                 point. If omitted, it is set to 'POINT'.  
=====

The second and third moments on the printout can be  
converted to Buckingham's tensors by formula 9 of  
A.D.Buckingham, Quart.Rev. 13, 183-214 (1959)  
These can in turn be converted to spherical tensors  
by the formulae in the appendix of  
S.L.Price, et al. Mol.Phys. 52, 987-1001 (1984)  
=====

## **\$MOROKM group**

(relevant for RUNTYP=MOROKUMA)

This group controls how the supermolecule input in the \$DATA group is divided into two or more monomers. Both the supermolecule and its constituent monomers must be well described by RHF wavefunctions.

MOROKM = a flag to request Morokuma-Kitaura decomposition.  
(default is .TRUE.)

RVS = a flag to request "reduced variation space" decomposition. This differs from the Morokuma option, and one or the other or both may be requested in the same run. (default is .FALSE.)

BSSE = a flag to request basis set superposition error be computed. You must ensure that CTPSPL is selected. This option applies only to MOROKM decompositions, as a basis superposition error is automatically generated by the RVS scheme. This is not the full Boys counterpoise correction, as explained in the reference. (default is .FALSE.)

\* \* \*

IATM = An array giving the number of atoms in each of the monomer. Up to ten monomers may be defined. Your input in \$DATA must have all the atoms in the first monomer defined before the atoms in the second monomer, before the third monomer... The number of atoms belonging to the final monomer can be omitted. There is no sensible default for IATM, so don't omit it from your input.

ICHM = An array giving the charges of the each monomer. The charge of the final monomer may be omitted, as it is fixed by ICH in \$CONTRL, which is the total charge of the supermolecule. The default is neutral monomers, ICHM(1)=0,0,0,...

EQUM = a flag to indicate all monomers are equivalent by symmetry (in addition to containing identical atoms). If so, which is not often true, then only the unique computations will be done.  
(default is .FALSE.)

CTPSPL = a flag to decompose the interaction energy into charge transfer plus polarization terms. This is most appropriate for weakly interacting monomers. (default is .TRUE.)

CTPLX = a flag to combine the CT and POL terms into a single term. If you select this, you might want to turn CTPSPL off to avoid the extra work that that decomposition entails, or you can analyze

both ways in the same run (default=.FALSE.)

RDENG = a flag to enable restarting, by reading the lines containing "FINAL ENERGY" from a previous run. The \$ENERGY group is single lines read under format A16,F20.10 containing the E, and a card \$END to complete. The 16 chars = anything. (default is .FALSE.)

=====

The present implementation has some quirks:

1. The initial guess of the monomer orbitals is not controlled by \$GUESS. The program first looks for a \$VEC1, \$VEC2, ... group for each monomer. If they are found, they will be MOREAD. If any of these are missing, the guess for that monomer will be constructed by HCORE. Check your monomer energies carefully! The initial guess orbitals for the supermolecule are formed by a block diagonal matrix of the monomer orbitals.
2. The use of symmetry is turned off internally.
3. There is no direct SCF option. File ORDINT will be a full C1 list of integrals. File AOINTS will contain whatever subset of these is needed for each particular decomposition step. So extra disk space is needed compared to RUNTYP=ENERGY.
4. This kind of run applies only to ab initio cases.
5. This kind of run will work in parallel.

References:

- C.Coulson in "Hydrogen Bonding", D.Hadzi, H.W.Thompson, Eds., Pergamon Press, NY, 1957, pp 339-360.  
C.Coulson Research, 10, 149-159 (1957).  
K.Morokuma J.Chem.Phys. 55, 1236-44 (1971).  
K.Kitaura, K.Morokuma Int.J.Quantum Chem. 10, 325 (1976).  
K.Morokuma, K.Kitaura in "Chemical Applications of Electrostatic Potentials", P.Politzer, D.G.Truhlar, Eds. Plenum Press, NY, 1981, pp 215-242.

The method coded is the newer version described in the latter two papers. Note that the CT term is computed separately for each monomer, as described in the words below equation 16 of the 1981 paper, not simultaneously.

Reduced Variational Space:

W.J.Stevens, W.H.Fink, Chem.Phys.Lett. 139, 15-22(1987).

A comparison of the RVS and Morokuma decompositions can be found in the review article: "Wavefunctions and Chemical Bonding" M.S.Gordon, J.H.Jensen in "Encyclopedia of Computational Chemistry", volume 5, P.V.R.Schleyer, editor, John Wiley and Sons, Chichester, 1998.

BSSE during Morokuma decomposition:

R.Cammi, R.Bonaccorsi, J.Tomasi  
Theoret.Chim.Acta 68, 271-283(1985).



The present implementation:  
"Energy decomposition analysis for many-body interactions,  
and application to water complexes"  
W.Chen, M.S.Gordon J.Phys.Chem. 100, 14316-14328 (1996)

=====  
**\$FFCALC group**

(relevant for RUNTYP=FFIELD)

This group permits the study of the influence of an applied electric field on the wavefunction. The most common finite field calculation applies a sequence of fields to extract the linear polarizability and first and second order hyperpolarizability. The method is general, and so works for all ab initio wavefunctions in GAMESS.

EFIELD = applied electric field strength  
(default=0.001 a.u.)

IAXIS and JAXIS specify the orientation of the applied field. 1,2,3 mean x,y,z respectively.  
The default is IAXIS=3 and JAXIS=0.

If IAXIS=i and JAXIS=0, the program computes alpha(ii), beta(iii), and gamma(iiii) from the energy changes, and a few more components from the dipole changes. Five wavefunction evaluations are performed.

If IAXIS=i and JAXIS=j, the program computes the cross terms beta(ijj), beta(iij), and gamma(iijj) from the energy changes, and a few more components from the dipole changes. This requires nine evaluations of the wavefunction.

AOFF = a flag to permit evaluation of alpha(ij) when the dipole moment is not available. This is necessary only for MP2, and means the off-axial calculation will do 13, not 9 energy evaluations. Default=.FALSE.

SYM = a flag to specify when the fields to be applied along the IAXIS and/or JAXIS (or according to EONE below) do not break the molecular symmetry. Since most fields do break symmetry, the default is .FALSE.

ONEFLD = a flag to specify a single applied field calculation will be performed. Only the energy and dipole moment under this field are computed. If this option is selected, only SYM and EONE input is heeded. The default is .FALSE.

EONE = an array of the three x,y,z components of the single applied field.

There are notes on RUNTYP=FFIELD on the next page.

Finite field calculations require large basis sets, and extraordinary accuracy in the wavefunction. To converge the SCF to many digits is sometimes problematic, but we suggest you use the input to increase integral accuracy and wavefunction convergence, for example

```
$CONTRL ICUT=20 ITOL=30 INTTYP=HONDO $END
$SCF NCONV=10 FDIFF=.FALSE. $END
```

In many cases, the applied fields will destroy the molecular symmetry. This means the integrals are calculated once with point group symmetry to do the initial field free wavefunction evaluation, and then again with point group symmetry turned off. If the fields applied do not destroy symmetry, you can avoid this second calculation of the integrals by SYM=.TRUE. This option also permits use of symmetry during the applied field wavefunction evaluations.

Examples of fields that do not break symmetry are a Z-axis field for an axial point group which is not centrosymmetric (i.e. C2v). However, a second field in the X or Y direction does break the C2v symmetry. Application of a Z-axis field for benzene breaks D6h symmetry. However, you could enter the group as C6v in \$DATA while using D6h coordinates, and regain the prospect of using SYM=.TRUE. If you wanted to go on to apply a second field for benzene in the X direction, you might want to enter Cs in \$DATA, which will necessitate the input of two more carbon and hydrogen atom, but recovers use of SYM=.TRUE.

Reference: H.A.Kurtz, J.J.P.Stewart, K.M.Dieter  
J.Comput.Chem. 11, 82-87 (1990).

For analytic computation of static and also frequency dependent NLO proerties, for closed shell cases, see the \$TDHF group.

=====

**\$TDHF group** (relevant for SCFTYP=RHF if RUNTYP=TDHF)

This group permits the analytic calculation of various static and/or frequency dependent polarizabilities, with an emphasis on important NLO properties such as second and

third harmonic generation. The method is programmed only for closed shell wavefunctions, at the semi-empirical or ab initio level. Ab initio calculations may be direct SCF, or parallel, if desired.

Because the Fock matrices computed during the time-dependent Hartree-Fock CPHF are not symmetric, you may not use symmetry. You must enter NOSYM=1 in \$CONTRL!

For a more general numerical approach to the static properties, see \$FFCALC.

NFREQ = Number of frequencies to be used. (default=1)

FREQ = An array of energy values in atomic units. For example: if NFREQ=3 then FREQ(1)=0.0,0.1,0.25. By default, only the static polarizabilities are computed. (default is freq(1)=0.0)

The conversion factor from Hartree to wave numbers is 219,474.6, and the wavelength is given (in nm) by 45.56/FREQ.

MAXITA = Maximum number of iterations for an alpha computation. (default=100)

MAXITU = Maximum number of iterations in the second order correction calculation. This applies to iterative beta values and all gammas. (default=100)

ATOL = Tolerance for convergence of first-order results. (default=1.0d-05)

BTOL = Tolerance for convergence of second-order results. (default=1.0d-05)

RETDHF = a flag to choose starting points for iterative calculations from best previous results. (default=.true.)

\* \* \* the following NLO properties are available \* \* \*

BSHG = Calculate beta for second harmonic generation.

BEOPE = Calculate beta for electrooptic Pockels effect.

BOR = Calculate beta for optical rectification.

GTHG = Calculate gamma for third harmonic generation.

GEFISH = Calculate gamma for electric-field induced second harmonic generation.

GIDRI = Calculate gamma for intensity dependent refractive index.

GOKE = Calculate gamma for optical Kerr effect.

These will be computed only if a nonzero energy is requested. The default for each flag is `.TRUE.`, and they may be turned off individually by setting some `.FALSE.` Note however that the program determines the best way to calculate them. For example, if you wish to have the SHG results but no gamma results are needed, the SHG beta will be computed in a non-iterative way from  $\alpha(\omega)$  and  $\alpha(2\omega)$ . However if you request the computation of the THG gamma, the second order  $U(\omega, \omega)$  results are needed and an iterative SHG calculation will be performed whether you request it or not, as it is a required intermediate.

Reference:

S.P.Karna, M.Dupuis *J.Comput.Chem.* 12, 487-504 (1991).  
P.Korambath, H.A.Kurtz, in "Nonlinear Optical Materials", ACS Symposium Series 628, S.P.Karna and A.T.Yeates, Eds. pp 133-144, Washington DC, 1996.

Review: D.P.Shelton, J.E.Rice, *Chem.Rev.* 94, 3-29(1994).

Starting from the PC GAMESS v. 7.0 build # 3448, TDHF and DFT code were extended to allow calculation of static and dynamic (hyper) polarizabilities using `runtyp=tdhf`.

For DFT, alpha values are exact, while beta and gamma are only approximate at present, as second-order (and higher) exchange-correlation kernels are not properly taken into account. This will be fixed when TDDFT gradient code will be incorporated into the future PC GAMESS versions.

The rest of this section is devoted to `CITYP=TDHF` runs for TDHF (a.k.a. RPA) excitation energies.

Current implementation allows the use of only RHF references, but can pick up both singlet and triplet excited states. Nuclear gradients are not yet programmed. Properties are available using "unrelaxed" density. Due to efficiency considerations, TDHF is programmed for SAPS (spin-adapted antisymmetrized product) basis only, so you cannot get both singlet and triplet states at once.

`NCORE = n` Omits the first n occupied alpha and beta orbitals from the calculation. The default for n is the number of chemical core orbitals.

`NSTATE =` Number of states to be found (excluding the ground state).

`ISTATE =` State for which properties and/or gradient will be calculated. Only one state can be chosen.

`MULT =` Multiplicity (1 or 3) of the singly excited SAPS (the reference is necessarily single RHF).

`DIAGZN =` Hamiltonian diagonalization method.  
= DAVID use Davidson diagonalization. (default)  
= FULL construct the full matrix in memory and

diagonalize, thus determining all states  
(not recommended except for small cases).

- NGSVEC = Dimension of the Hamiltonian submatrix that is diagonalized to form the initial CI vectors. The default is the greater of NSTATE\*2 and 10.
- MXVEC = Maximum number of expansion basis vectors in the iterative subspace during Davidson iterations, before the expansion basis is truncated. The default is the larger of 8\*NSTATE and NGSVEC.
- NDAVIT = Maximum number of Davidson iterations. Default=50.
- DAVCVG = Convergence criterion for Davidson eigenvectors. Eigenvector accuracy is proportional to DAVCVG, while the energy accuracy is proportional to its square. The default is 3.0E-05.
- RDTDVC = Flag to read TDHF vectors from a \$TDVEC group in the input file. Default is .FALSE.
- MNMEOP = Flag to force the use of the minimal amount of memory during the Davidson iterations. This is for debug purposes. The default is .FALSE.
- MAXGC = maximum allowed number of trial vectors to be routed through GENCON engine, default is 1. If the number of trial vectors is greater than MAXGC, only FASTINTS will be used. The reason is that for moderately contracted GC basis sets like cc-pVXZ, gencon is faster than fastints only for relatively small number of trial vectors (this is by gencon design). On the other hand, for ANO-like basis sets, it is always better to set MAXGC to be equal the number of initial guess vectors, as fastints will be much slower.
- PRTTOL = threshold for TDHF csf printout and also for states symmetry determination. Default is 0.05.
- ISTSYM = symmetry of states of interest. Default is zero, i.e., does not use any symmetry during calculations. Setting this to the desired index of irrep (according to GAMESS numbering) will solve only for the states of the desired symmetry and exploiting full (including non-abelian) symmetry of molecule, thus significantly reducing computation time.

The state-tracking feature of the PC GAMESS' TDHF code can be activated by selecting negative value of istate in the \$TDHF group. It is intended for geometry optimization of the excited states in the case of root flipping.

Note that oscillator strengths printed in the TDHF summary table are calculated using transition dipoles length form only.

Below is the sample input file:

```

$CONTRL SCFTYP=RHF CITYP=TDHF $END
$SYSTEM TIMLIM=3000 MEMORY=3000000 $END
$BASIS GBASIS=n31 ngauss=6 NDFUNC=1 $END
$TDHF NSTATE=3 ISTDY=0 ISTATE=1 $END
$DATA
H2O
CNV 2

O      8.0  0.0000000000  0.0000000000  0.7205815395
H      1.0  0.0000000000  0.7565140024  0.1397092302
$END

```

```

=====
$TDVEC group      required if RDTDVC in $TDHF is chosen

```

This is formatted data generated by a previous TDHF run, to be read back in as starting vectors.

```

=====

```

### **\$TDDFT group**

required when CITYP=TDDFT  
(note that CITYP=TDDFT requires  
DFTTYP to be set in \$CONTRL)

Current implementation allows the use of only R-DFT references, but can pick up both singlet and triplet excited states. Nuclear gradients are not yet programmed (will be in the PC GAMESS v. 7.2). Properties are available using "unrelaxed" density. Due to efficiency considerations, TDDFT is programmed for SAPS (spin-adapted antisymmetrized product) basis only, so you cannot get both singlet and triplet states at once.

```

NCORE = n  Omits the first n occupied alpha and beta orbitals from
           the calculation.  The default for n is the number
           of chemical core orbitals.

NSTATE =   Number of states to be found (excluding the
           ground state).

ISTATE =   State for which properties and/or gradient will
           be calculated.  Only one state can be chosen.

MULT     =   Multiplicity (1 or 3) of the singly excited
           SAPS (the reference is necessarily single R-DFT).

DIAGZN =   Hamiltonian diagonalization method.
           =   DAVID use Davidson diagonalization.  (default)
           =   FULL  construct the full matrix in memory and
                   diagonalize, thus determining all states

```

(not recommended except for small cases).

- NGSVEC = Dimension of the Hamiltonian submatrix that is diagonalized to form the initial CI vectors. The default is the greater of NSTATE\*2 and 10.
- MXVEC = Maximum number of expansion basis vectors in the iterative subspace during Davidson iterations, before the expansion basis is truncated. The default is the larger of 8\*NSTATE and NGSVEC.
- NDAVIT = Maximum number of Davidson iterations. Default=50.
- DAVCVG = Convergence criterion for Davidson eigenvectors. Eigenvector accuracy is proportional to DAVCVG, while the energy accuracy is proportional to its square. The default is 3.0E-05.
- RDTDVC = Flag to read TDDFT vectors from a \$TDVEC group in the input file. Default is .FALSE.
- MNMEOP = Flag to force the use of the minimal amount of memory during the Davidson iterations. This is for debug purposes. The default is .FALSE.
- MAXGC = maximum allowed number of trial vectors to be routed through GENCON engine, default is 1. If the number of trial vectors is greater than MAXGC, only FASTINTS will be used. The reason is that for moderately contracted GC basis sets like cc-pVXZ, gencon is faster than fastints only for relatively small number of trial vectors (this is by gencon design). On the other hand, for ANO-like basis sets, it is always better to set MAXGC to be equal the number of initial guess vectors, as fastints will be much slower.
- PRTTOL = threshold for TDDFT csf printout and also for states symmetry determination. Default is 0.05.
- ISTSYM = symmetry of states of interest. Default is zero, i.e., does not use any symmetry during calculations. Setting this to the desired index of irrep (according to GAMESS numbering) will solve only for the states of the desired symmetry and exploiting full (including non-abelian) symmetry of molecule, thus significantly reducing computation time.
- ALTER = flag to modify internal logic of Davidson diagonalization code to use dynamic number of trial vectors. Default is .true. Setting it to .false. will slow-down calculations by forcing DAVIDSON diagonalization code to work exactly as CIS code in GAMESS (US).
- TDA = flag to request Tamm-Dancoff approximation to TDDFT (TDDFT/TDA), which is programmed for both pure and hybrid functionals. Default is .false.

The state-tracking feature of the PC GAMESS' TDDFT code can be

activated by selecting negative value of istate in the \$TDDFT group. It is intended for geometry optimization of the excited states in the case of root flipping.

Note that oscillator strengths printed in the TDDFT summary table are calculated using transition dipoles length form only.

Below is the sample input file:

```
$CONTRL SCFTYP=RHF DFTTYP=BLYP CITYP=TDDFT $END
$SYSTEM TIMLIM=3000 MEMORY=3000000 $END
$BASIS GBASIS=n31 ngauss=6 NDFUNC=1 $END
$TDDFT NSTATE=3 ISTDY=0 ISTATE=1 $END
$DATA
H2O
CNV 2

O          8.0   0.0000000000   0.0000000000   0.7205815395
H          1.0   0.0000000000   0.7565140024   0.1397092302
$END
```

The example of TDDFT/TDA calculations:

```
$CONTRL SCFTYP=RHF DFTTYP=B3LYP CITYP=TDDFT $END
$SYSTEM TIMLIM=3000 MEMORY=3000000 $END
$BASIS GBASIS=n31 ngauss=6 NDFUNC=1 $END
$TDDFT NSTATE=3 ISTDY=0 ISTATE=1 TDA=.t. $END
$DATA
H2O
CNV 2

O          8.0   0.0000000000   0.0000000000   0.7205815395
H          1.0   0.0000000000   0.7565140024   0.1397092302
$END
```

=====

**\$TDVEC group** required if RDTDVC in \$TDDFT is chosen

This is formatted data generated by a previous TDDFT run, to be read back in as starting vectors.

=====

=====



## **\$EFRAG group** (optional)

This group gives the name and position of one or more effective fragment potentials. It consists of a series of free format card images, which may not be combined onto a single line! The position of a fragment is defined by giving any three points within the fragment, relative to the ab initio system defined in \$DATA, since the effective fragments have a frozen internal geometry. All other atoms within the fragment are defined by information in the \$FRAGNAME group.

-----

-1- a line containing one or more of these options:

COORD	=CART	selects use of Cartesians coords to define the fragment position at line -3-. (default)
	=INT	selects use of Z-matrix internal coordinates at line -3-.
POLMETHD	=SCF	indicates the induced dipole for each fragment due to the ab initio electric field and other fragment fields is updated only once during each SCF iteration.
	=FRGSCF	requests microiterations during each SCF iteration to make induced dipoles due to ab initio and other fragment fields self consistent among the fragments. (default) Both methods converge to the same dipolar interaction.
POSITION	=OPTIMIZE	Allows full optimization within the ab initio part, and optimization of the rotational and translational motions of each fragment. (default)
	=FIXED	Allows full optimization of the ab initio system, but freezes the position of the fragments. This makes sense only with two or more fragments, as what is frozen is the fragments' relative orientation.
	=EFOPT	the same as OPTIMIZE, but if the fragment gradient is large, up to 5 geometry steps in which only the fragments move may occur, before the geometry of the ab initio piece is relaxed. This may save time by reusing the two electron integrals for the ab initio system.

Input a blank line if all the defaults are acceptable.

-----  
-2- FRAGNAME=XXX

XXX is the name of the fragment whose coordinates are to be given next. All other information defining the fragment is given in a supplemental \$XXX group, which is referred to below as a \$FRAGNAME group.

A RHF/DZP EFP for water is internally stored in GAMESS. Choose FRAGNAME=H2OEF2 to look up this numerical data, and then skip the input of \$H2OEF2 and \$FRGRPL groups.

-----  
-3- NAME, X, Y, Z (COORD=CART)  
NAME, I, DISTANCE, J, BEND, K, TORSION (COORD=INT)

NAME = the name of a fragment point. The name used here must match one of the points in \$FRAGNAME.

X, Y, Z = Cartesian coordinates defining the position of this fragment point RELATIVE TO THE COORDINATE ORIGIN used in \$DATA. The choice of units is controlled by UNITS in \$CONTRL.

I, DISTANCE, J, BEND, K, TORSION = the usual Z-matrix connectivity internal coordinate definition. The atoms I, J, K must be atoms in the ab initio system from in \$DATA, or fragment points already defined in the current fragment or previously defined fragments.

Line -3- must be given a total of three times to define this fragment's position.

-----  
Repeat lines -2- and -3- to enter as many fragments as you desire, and then end the group with a \$END line.

Note that it is quite typical to repeat the same fragment name at line -2-, to use the same fragment system at many different positions.

=====  
\* \* \* \* \*  
For tips on effective fragment potentials  
see the 'further information' section  
\* \* \* \* \*

=====  
(required for each FRAGNAME given in \$EFRAG)

## **\$FRAGNAME group**

This group gives all pertinent information for a given effective fragment potential (EFP). This information falls into three categories:

- electrostatic (distributed multipoles, screening)
- distributed polarizabilities
- exchange repulsion

It is input using several different subgroups, which should be given in the order shown below. Each subgroup is specified by a particular name, and is terminated by the word STOP. You may omit any of the subgroups to omit that term from the EFP. All values are given in atomic units.

To input monopoles,	follow input sequence -EM-
To input dipoles,	follow input sequence -ED-
To input quadrupoles,	follow input sequence -EQ-
To input octupoles,	follow input sequence -EO-
To input screening parameters,	follow input sequence -ES-
To input polarizable points,	follow input sequence -P-
To input repulsive points,	follow input sequence -R-

-----  
-1- a single descriptive title card  
-----

-2- COORDINATES

COORDINATES signals the start of the subgroup containing the multipolar expansion terms (charges, dipoles, ...). Optionally, one can also give the coordinates of the polarizable points, or centers of exchange repulsion.

-3- NAME, X, Y, Z, WEIGHT, ZNUC

NAME is a unique string identifying the point.  
X, Y, Z are the Cartesian coordinates of the point.  
WEIGHT and ZNUC are the atomic mass and nuclear charge, and are given only for the points which are nuclei.

Typically the true nuclei will appear twice, once for defining the positive nuclear charge and its screening, and a second time for defining the electronic distributed multipoles.

Repeat line -3- for each expansion point, and terminate the list with a "STOP".  
-----

-EM1- MONOPOLES

MONOPOLES signals the start of the subgroup containing the electronic and nuclear monopoles.

-EM2- NAME, CHARGE

NAME must match one given in the COORDINATES subgroup.  
CHARGE = nuclear or electronic monopole at this point.

Repeat -EM2- to define all desired charges.  
Terminate this subgroup with a "STOP".

---

-ED1- DIPOLES

DIPOLES signals the start of the subgroup containing the  
dipolar part of the multipolar expansion.

-ED2- NAME, MUX, MUY, MUZ

NAME must match one given in the COORDINATES subgroup.  
MUX, MUY, MUZ are the components of the electronic dipole.

Repeat -ED2- to define all desired dipoles.  
Terminate this subgroup with a "STOP".

---

-EQ1- QUADRUPOLES

QUADRUPOLES signals the start of the subgroup containing  
the quadrupolar part of the multipolar expansion.

-EQ2- NAME, XX, YY, ZZ, XY, XZ, YZ

NAME must match one given in the COORDINATES subgroup.  
XX, YY, ZZ, XY, XZ, and YZ are the components of the  
electronic quadrupole moment.

Repeat -EQ2- to define all desired quadrupoles.  
Terminate this subgroup with a "STOP".

---

-EO1- OCTUPOLES

OCTUPOLES signals the start of the subgroup containing  
the octupolar part of the multipolar expansion.

-EO2- NAME, XXX, YYY, ZZZ, XXY, XXZ,  
          XYY, YYZ, XZZ, YZZ, XYZ

NAME must match one given in the COORDINATES subgroup.  
XXX, ... are the components of the electronic octupole.

Repeat -EO2- to define all desired octupoles.  
Terminate this subgroup with a "STOP".

---

-ES1- SCREEN

SCREEN signals the start of the subgroup containing the screening terms ( $A \cdot \exp[-B \cdot r^2]$ ) for the distributed multipoles, which account for charge penetration effects.

-ES2- NAME, A, B

NAME must match one given in the COORDINATES subgroup. A, B are the parameters of the Gaussian screening term.

Repeat -ES2- to define all desired screening points. Terminate this subgroup with a "STOP".

-----

-P1- POLARIZABLE POINTS

POLARIZABLE POINTS signals the start of the subgroup containing the distributed polarizability tensors, and their coordinates.

-P2- NAME, X, Y, Z

NAME gives a unique identifier to the location of this polarizability tensor. It might match one of the points already defined in the COORDINATES subgroup, but often does not. Typically the distributed polarizability tensors are located at the centroids of localized MOs.

X, Y, Z are the coordinates of the polarizability point. They should be omitted if NAME did appear in COORDINATES. The units are controlled by UNITS= in \$CONTRL.

-P3- XX, YY, ZZ, XY, XZ, YZ, YX, ZX, ZY

XX, ... are components of the distributed polarizability, which is not a symmetric tensor. XY means  $d\mu_x/dF_y$ , where  $\mu_x$  is a dipole component, and  $F_y$  is a component of an applied field.

Repeat -P2- and -P3- to define all desired polarizability tensors, and terminate this subgroup with a "STOP".

-----

-R1- REPULSIVE POTENTIAL

REPULSIVE POTENTIAL signals the start of the subgroup containing the fitted exchange repulsion potential, for the interaction between the fragment and the ab initio part of the system. This term also accounts for charge transfer effects. The term has the form

$$\sum_i^N C_i \exp[-D_i \cdot r^2]$$

-R2- NAME, X, Y, Z, N

NAME may match one given in the COORDINATES subgroup, but need not. If NAME does not match one of the known points, you must give its coordinates X, Y, and Z, otherwise omit these three values. N is the total number of terms in the fitted repulsive potential.

-R3- C, D

These two values define the i-th term in the repulsive potential. Repeat line -R3- for all N terms.

Repeat -R2- and -R3- to define all desired repulsive potentials, and terminate this subgroup with a "STOP".

=====

The entire \$FRAGNAME group is terminated by a " \$END".

=====

## ***\$FRGRPL group***

This group defines the inter-fragment repulsive potential, which consists primarily of exchange repulsions but also includes charge transfer. Note that the functional form used for the fragment-fragment repulsion differs from that used for the ab initio-fragment repulsion, which is defined in the \$FRAGNAME group. The form of the potential is

$$\sum_i^N A_i \exp[-B_i r]$$

-----

-1- PAIR=FRAG1 FRAG2

specifies which two fragment repulsions are being defined. \$FRAGNAME input for the two names FRAG1 and FRAG2 must have been given.

-----

-2- NAME1 NAME2 A B  
      \*or\*  
      NAME1 NAME2 'EQ' NAME3 NAME4

NAME1 must be one of the "NAME" points defined in the \$FRAG1 group's COORDINATE section. Similarly NAME2 must be a point from the \$FRAG2 group. In addition, NAME1 or NAME2 could be the keyword CENTER, indicating the center of mass of the fragment.

A and B are the parameters of the fitted repulsive potential.

The second form of the input allows equal potential fits to be used. The syntax implies that the potential between the points NAME1 and NAME2 should be taken the same as the potential previously given in this group for the pair of points NAME3 and NAME4.

If there are NPT1 points in FRAG1, and NPT2 points in FRAG2, input line -2- should be repeated NPT1\*NPT2 times. Terminate the pairs of potentials with a "STOP" card. Any pairs which you omit will be set to zero interaction.

Typically the number of points on which fitted potentials might be taken to be all the nuclei in a fragment, plus the center of mass.

-----  
Repeat lines -1- and -2- for all pairs of fragments, then terminate the group with a \$END line.

=====  
**\$PCM group** (optional)

This group controls solvent effect computations using the Polarizable Continuum Method. If this group is found in the input file, a PCM computation is performed. The default calculation, chosen by selecting only the SOLVNT keyword, is to compute the electrostatic free energy. Appropriate numerical constants are provided for a wide range of solvents. Additional keywords allow for more sophisticated computations, namely cavitation, repulsion, and dispersion free energies. The methodology for these is general, but only numerical constants for water are provided. There is additional information on PCM in the References chapter of this manual.

**ANALYZ =** Number of passes performed with the PCManalyze in order to remove problematic tesseræ and improve numerical stability. In some instances PCManalyze tends to enter into a quasi-infinite loop. If this happens, set analyz=0 to skip this procedure.

--- the first set of parameters controls the computation:  
ICOMP, ICAV, IDISP, IREP, IDP, and IFIELD.

ICOMP = Renormalization procedure for induced charges.  
Gradient runs require ICOMP be 0 or 2 only.  
= 0 No.  
= 1 Yes, each charge is corrected in proportion to the area of the tesseræ to which it belongs.  
= 2 Yes, using the same factor for all tesseræ.  
(default)

= 3 Yes, with explicit consideration of the portion of solute electronic charge outside the cavity, by the method of Mennucci and Tomasi. See the \$NEWCAV group.

ICAV = At the end of the run, calculate the cavitation energy, by the method of Pierotti and Claverie:  
= 0 skip the computation (default)  
= 1 perform the computation.

If ICAV=1, the following parameter is relevant:

TABS = the absolute temperature, in units K.  
(default=298.0)

There are two procedures for the calculation of the repulsion and dispersion free energy. IDISP is incompatible with IREP and IDP.

IDISP = Calculation of both dispersion and repulsion free energy through the empirical method of Floris and Tomasi.  
= 0 skip the computation (default)  
= 1 perform the computation. See \$DISREP group.

The next two options add repulsive and dispersive terms to the solute hamiltonian, in an ab initio manner, by the method of Amovilli and Mennucci.

IREP = Calculation of repulsion free energy  
= 0 skip the computation (default)  
= 1 perform the computation. See \$NEWCAV group.

IDP = Calculation of dispersion free energy  
= 0 skip the computation (default)  
= 1 perform the computation. See \$DISBS group.

If IDP=1, then three additional parameters must be defined. The two solvent values correspond to water, and therefore these must be input for other solvents.

WA = solute average transition energy. This is computed from the orbital energies for RHF, but must be input for MCSCF runs.  
(default=1.10)

WB = ionization potential of solvent, in Hartrees.  
(default=0.451)

ETA2 = square of the zero frequency refractive index of the solvent. (default=1.75)

IFIELD = At run end, calculate the electric potential and electric field generated by the apparent surface charges.  
= 0 skip the computation (default)  
= 1 on nuclei



= 2 on a planar grid

If IFIELD=2, the following data must be input:

AXYZ,BXYZ,CXYZ = each defines three components of the  
                  vertices of the plane where the reaction  
                  field is to be computed (in Angstroms)  
A ==> higher left corner of the grid  
B ==> lower left corner of the grid  
C ==> higher right corner of the grid  
NAB = vertical subdivision (A--B edge) of the grid  
NAC = horizontal subdivision (A--C edge) of the grid.

IPRINT = 0 normal printing (default)  
          = 1 turns on debugging printout

--- the next group of keywords defines the solvent

SOLVNT = keyword naming the solvent of choice. The eight  
         numerical constants defining the solvent are  
         internally stored for the following:

WATER (or H2O)  
CH3OH  
C2H5OH  
CLFORM (or CHCl3)  
METHYCL (or CH2Cl2)  
12DCLET (or C2H4Cl2)  
CTCL (or CCl4)  
BENZENE (or C6H6)  
TOLUENE (or C6H5CH3)  
CLBENZ (or C6H5Cl)  
NITMET (or CH3NO2)  
NEPTANE (or C7H16)  
CYCHEX (or C6H12)  
ANILINE (or C6H5NH2)  
ACETONE (or CH3COCH3)  
THF  
DMSO (or DMETSOX)

The default solvent name is  
INPUT

which indicates you will specify your solvent by  
giving the following 8 numerical values:

RSOLV = the solvent radius, in units Angstrom  
EPS = the dielectric constant  
EPSINF = the dielectric constant at infinite frequency.  
This value must be given only for RUNTYP=TDHF,  
if the external field frequency is in the optical  
range and the solvent is polar; in this case the  
solvent response is described by the electronic  
part of its polarization. Hence the value of the  
dielectric constant to be used is that evaluated  
at infinite frequency, not the static one (EPS).  
For nonpolar solvents, the difference between  
the two is almost negligible.

TCE = the thermal expansion coefficient, in units 1/K  
VMOL = the molar volume, in units ml/mole  
STEN = the surface tension, in units dyne/cm  
DSTEN = the thermal coefficient of log(STEN)  
CMF = the cavity microscopic coefficient

Values for TCE, VMOL, STEN, DSTEN, CMF need to be given only for the case ICAV=1. Input of any or all of these values will override the internally stored value.

--- the next set of keywords defines the molecular cavity

NESFP = the number of initial spheres.  
(default = number of atoms in solute molecule)

ICENT = option for definition of initial spheres.  
= 0 centers spheres on each nucleus. (default)  
= 1 sphere centers XE, YE, ZE and radii RIN will be specified explicitly in \$PCMCAV.

The cavity generation algorithm may use additional spheres to smooth out sharp grooves, etc. The following parameters control how many extra spheres are generated:

OMEGA and FRO = GEPOL parameters for the creation of the 'added spheres' defining the solvent accessible surface. When an excessive number of spheres is created, which may cause problems of convergence, the value of OMEGA and/or FRO must be increased. For example, OMEGA from 40 to 50 ... up to 90,  
FRO from 0.2 ... up to 0.7.  
(defaults are OMEGA=40.0, FRO=0.7)

RET = minimum radius (in A) of the added spheres. Increasing RET decreases the number of added spheres. A value of 100.0 inhibits the addition of spheres. (default=0.2)

=====

## ***\$PCMCAV group***

(optional)

This group controls generation of the cavity holding the solute during Polarizable Continuum Method runs. The cavity is a union of spheres, according to ICENT and associated input values given in \$PCM. The data given here must be given in Angstrom units.

XE,YE,ZE = arrays giving the coordinates of the spheres.  
if ICENT=0, the atomic positions will be used.  
if ICENT=1, you must supply NESFP values here.

RIN = an array giving the sphere radii.  
if ICENT=0, the program will look up the internally

stored van der Waals radius for: H,He,  
B,C,N,O,F,Ne, Na,Al,Si,P,S,Cl,Ar,  
K,As,Se,Br,Kr, Rb,Sb,Te,I, Cs,Bi  
Data for other elements is not tabulated.  
if ICENT=1, give NESFP values.

ALPHA = an array of scaling factors, for the definition of  
the solvent accessible surface. If only the first  
value is given, all radii are scaled by the same  
factor. (default is ALPHA(1)=1.2)

Example: Suppose the 4th atom in your molecule is Fe, but  
all other atoms have van der Waals radii. You  
decide a good guess for Fe is twice the covalent  
radius: \$PCMCAV RIN(4)=2.33 \$END

The source for the van der Waals radii is "The Elements",  
2nd Ed., John Emsley, Clarendon Press, Oxford, 1991,  
except that for C,N,O, the U.Pisa's experience with the  
best radii for PCM treatment of singly bonded C,N,O atoms  
is used instead. The radii for a few transition metals  
are given by A.Bondi, J.Phys.Chem. 68, 441-451(1968).

=====

### **\$NEWCAV group**

(optional)

This group controls generation of the "escaped charge"  
cavity, used when ICOMP=3 or IREP=1 in \$PCM. This cavity  
is used only to calculate the fraction of the solute  
electronic charge escapes from the original cavity.

IPTYPE = choice for tessellation of the cavity's spheres.  
= 1 uses a tetrahedron  
= 2 uses a pentakis dodecahedron (default)

ITSNUM = m, the number of tessera to use on each sphere.

if IPTYPE=1, input  $m=30*(n**2)$ , with n=1,2,3 or 4  
if IPTYPE=2, input  $m=60*(n**2)$ , with n=1,2,3 or 4  
(default is 60)

\*\*\* the next three parameters pertain to IREP=1 \*\*\*

RHOW = density, relative to liquid water (default = 1.0)

PM = molecular weight (default = 18.0)

NEVAL = number of valence electrons on solute (default=8)

The defaults for RHOW, PM, and NEVAL correspond to water,  
and therefore must be correctly input for other solvents.

=====

## **\$DISBS group**

(optional)

This group defines auxiliary basis functions used to evaluate the dispersion free energy by the method of Amovilli and Mennucci. These functions are used only for the dispersion calculation, and thus have nothing to do with the normal basis given in \$BASIS or \$DATA. If the input group is omitted, only the normal basis is used for the IDP=1 dispersion energy.

NADD = the number of added shells

XYZE = an array giving the x,y,z coordinates (in bohr) of the center, and exponent of the added shell, for each of the NADD shells.

NKTYPE = an array giving the angular momenta of the shells

An example placing 2s,2p,2d,1f on one particular atom,

```
$DISBS  NADD=7  NKTYP(1)= 0 0 1 1 2 2 3
        XYZE(1)=2.9281086  0.0  .0001726  0.2
                2.9281086  0.0  .0001726  0.05
                2.9281086  0.0  .0001726  0.2
                2.9281086  0.0  .0001726  0.05
                2.9281086  0.0  .0001726  0.75
                2.9281086  0.0  .0001726  0.2
                2.9281086  0.0  .0001726  0.2  $END
```

---

## **\$DISREP group**

(optional)

This group controls evaluation of the dispersion and repulsion energies by the empirical method of Floris and Tomasi. The group must be given with IDISP=1 in \$PCM. The two options are controlled by ICLAV and ILJ, only one of which should be selected.

ICLAV = selects Claverie's disp-rep formalism.  
= 0 skip computation.  
= 1 Compute the solute-solvent disp-rep interaction as a sum over atom-atom interactions through a Buckingham-type formula ( $R^{-6}$  for dispersion, exp for repulsion). (default)  
Ref: Pertsin-Kitaigorodsky "The atom-atom potential method", page 146.

ILJ = selects a Lennard-Jones formalism.  
= 0 skip computation. (default)  
= 1 solute atom's-solvent molecule interaction is modeled by Lennard-Jones type potentials,  $R^{-6}$  for dispersion,  $R^{-12}$  for repulsion).

---- the following data must given for ICLAV=1:

RHO = solvent numeral density  
N = number of atom types in the solvent molecule  
NT = an array of the number of atoms of each type in a solvent molecule  
RDIFF = distances between the first atoms of each type and the cavity  
DK,RW = parameters of atom-atom interactions

The defaults are chosen for water,

RHO=3.348D-02

N=2

NT(1)=2,1

RDIFF(1)=1.20,1.50

DKT(1)=1.0,1.36

RWT(1)=1.2,1.5

---- the following data must given for ILJ=1:

RHO = solvent numeral density  
EPSI = an array of energy constants referred to each atom of the solute molecule.  
SIGMA = an array of typical distances, relative to each solute atom

=====

## ***\$SCRF group***

(optional)

The presence of this group in the input turns on the use of the Kirkwood-Onsager spherical cavity model for the study of solvent effects. The method is implemented for RHF, UHF, ROHF, GVB and MCSCF wavefunctions and gradients, and so can be used with any RUNTYP involving the gradient. The method is not implemented for MP2, CI, any of the semiempirical models, or for analytic hessians.

DIELEC = the dielectric constant, 80 is often used for H2O

RADIUS = the spherical cavity radius, in Angstroms

G = the proportionality constant relating the solute molecule's dipole to the strength of the reaction field. Since G can be calculated from DIELEC and RADIUS, do not give G if they were given.

Additional information on the \$SCRF model can be found in the References chapter of this manual.

=====

## ***\$ECP group***

(required if ECP=READ in \$CONTRL)

This group lets you read in effective core potentials, for some or all of the atoms in the molecule. You can use built in potentials for some of the atoms if you like.

This is a free format (positional) input group.

\*\*\* Give a card set -1-, -2-, and -3- for each atom \*\*\*

-card 1- PNAME, PTYPE, IZCORE, LMAX+1

PNAME is a 8 character descriptive tag for this potential.  
If it is repeated for a subsequent atom, no other information need be given on this card, and cards -2- and -3- may also be skipped. The information will be copied from the first atom by this PNAME.  
Do not use the option to repeat the previously read ECP for an atom with PTYPE=NONE, instead type "none".  
PTYPE = GEN a general potential should be read.  
= SBKJC look up the Stevens/Basch/Krauss/Jasien/Cundari potential for this type of atom.  
= HW look up the Hay/Wadt built in potential for this type of atom.  
= NONE treat all electrons on this atom.  
IZCORE is the number of core electrons to be removed.  
LMAX is the maximum angular momentum occupied in the core orbitals being removed (usually). Give IZCORE and LMAX only if PTYPE is GEN.

\*\*\* For the first occurrence of PNAME, if PTYPE is GEN, \*\*\*  
\*\*\* then give cards -2- and -3-. Otherwise go to -1-. \*\*\*

\*\*\* Card sets -2- and -3- are repeated LMAX+1 times \*\*\*

The potential U(LMAX+1) is given first,  
followed by U(L)-U(LMAX+1), for L=1,LMAX.

-card 2- NGPOT

NGPOT is the number of Gaussians in this part of the local effective potential.

-card 3- CLP,NLP,ZLP (repeat this card NGPOT times)

CLP is the coefficient of this Gaussian in the potential.  
NLP is the power of r for this Gaussian.  
ZLP is the exponent of this Gaussian.

\* \* \*

By far the easiest way to use the SBKJC potential for all atoms in the formic acid molecule is to request ECP=SBKJC in \$CONTRL. But the next page shows two alternatives.

The first way is to look up the program's internally stored SBKJC potentials one atom at a time:

```
$ECP
C-ECP SBKJC
H-ECP NONE
```

```
O-ECP SBKJC
O-ECP
H-ECP NONE
$END
```

The second oxygen duplicates the first, no core electrons are removed for hydrogen. The order of the atoms must follow that generated by \$DATA. Note PTYPE allows you to type in one or more atoms explicitly, while using built in data for some other atoms.

The second example reads all SBKJC potentials explicitly:

```
$ECP
C-ECP GEN 2 1
1 ----- CARBON U(P) -----
-0.89371 1 8.56468
2 ----- CARBON U(S)-U(P) -----
1.92926 0 2.81497
14.88199 2 8.11296
H-ECP NONE
O-ECP GEN 2 1
1 ----- OXYGEN U(P) -----
-0.92550 1 16.11718
2 ----- OXYGEN U(S)-U(P) -----
1.96069 0 5.05348
29.13442 2 15.95333
O-ECP
H-ECP NONE
$END
```

Again, the 2nd oxygen copies from the first. It is handy to use the rest of card -2- as a descriptive comment.

As a final example, for antimony we have LMAX+1=3 (there are core d's). One must first enter U(f), followed by U(s)-U(f), U(p)-U(f), U(d)-U(f).

Caution:  
-----

At the present time, there are some numerical problems in the ECP code, which has been programed to use spdffg basis sets, and core potentials up to g. If one is using a g basis function, or a g potential (bottom row elements beyond the lanthanide series), there are small errors in the ECP integrals. A tight optimization (OPTTOL=1D-05) will usually result in the energy rising slightly during the last few geometry steps. The error seems to be about 0.000002 Hartree, so be cautious about using a g potential or basis function. When both are used in the same run, the error in the energy is about 0.0002, which means the use of both is too inaccurate to be trusted. The use of f functions or f potentials (or lower) seems to be free of any errors.

=====

**\$EFIELD group** (not required)

This group permits the study of the influence of an external electric field on the molecule. The method is general, and so works for all ab initio SCFTYPs.

EVEC = an array of the three x,y,z components of the applied electric field.

SYM = a flag to specify when the field to be applied breaks the molecular symmetry. Since most fields break symmetry, the default is .FALSE.

=====

Restrictions: analytic hessians are not available, but numerical hessians are. Because an external field causes a molecule with a dipole to experience a torque, geometry optimizations must be done in Cartesian coordinates only. Internal coordinates eliminate the rotational degrees of freedom, which are no longer free.

Notes: a hessian calculation will have two rotational modes with non-zero "frequency", caused by the torque. A gas phase molecule will rotate so that the dipole moment is anti-parallel to the applied field. To carry out this rotation during geometry optimization will take many steps, and you can help save much time by inputting a field opposite the molecular dipole. There is also a stationary point at higher energy with the dipole parallel to the field, which will have two imaginary frequencies in the hessian. Careful, these will appear as the first two modes in a hessian run, but will not have the i for imaginary included on the printout since they are rotational modes.

=====

**\$INTGRL group** (optional)

This group controls AO integral formats. It should probably never be given, as the program always picks sensible values.

SCHWRZ = a flag to activate use of the Schwarz inequality to predetermine small integrals. There is no loss of accuracy when choosing this option, and there are appreciable time savings for bigger molecules. Default=.TRUE. for over 5 atoms, or for direct SCF, and is .FALSE. otherwise.



NOPK = 0 PK integral option on, which is permissible  
for RHF, UHF, ROHF, GVB energy/gradient runs.  
= 1 PK option off (default for all jobs).  
Must be off for anything with a transformation.

NORDER = 0 (default)  
= 1 Sort integrals into canonical order. There  
is little point in selecting this option, as  
no part of GAMESS requires ordered integrals.  
See also NSQUAR.

NINTMX = Maximum no. of integrals in a record block.  
(default=15000 for J or P file, =10000 for PK)

The following parameters control the integral sort.  
(values given are defaults)

NSQUAR = 0 Sorted integrals will be in triangular  
canonical order (default)  
= 1 instead sort to square canonical order.  
NDAR = Number of direct access logical records to be  
used for the integral sort (default=2000)  
LDAR = Length of direct access records (site dependent)  
NBOXMX = 200 Maximum number of bins.  
NWORD = 0 Memory to be used (default=all of it).  
NOMEM = 0 If non-zero, force external sort.

IREST = 0 Normal operation (default)  
= 1 Only 1-e integrals and integrals needed for Schwarz  
screening are calculated, 2-e integrals are read from the  
= 2 Only the integrals needed for Schwarz screening are  
calculated, 2-e integrals are read from the existing  
AOINTS file.  
= -1 Similar to 1, but 2-e integrals are never  
recalculated, even if the molecular geometry is changed.  
= -2 Similar to 2, but 2-e integrals are never  
recalculated, even if the molecular geometry is changed.

PACKAO = .False. Normal operation  
= .True. Turns on the additional packing of the AO integral  
file. Both indices and integrals are packed. This  
option reduces the size of AOINTS file by a factor  
of up to 2.5-3, thus saving the disk space and the  
CPU time. See also a more detailed discussion on the  
packing-related problems. Relevant only for NOPK=1  
case (conventional integral storage mode, not in a  
supermatrix form)

PKTHR A threshold to activate packing of 2-e integrals.  
Default is  $10^{-7}$

The following parameters control integral restarts  
(values given are defaults)

IST= 1 JST= 1 KST= 1 LST= 1  
NREC= 1 INTLOC= 1

=====

## **\$FMM group** (relevant if QFMM selected in \$INTGRL)

QFMM is turned on by the logical variable QFMM in the \$INTGRL group (its default value is `.false.`, i.e., no QFMM calculations). You must select `DIRSCF=.TRUE.` in `$SCF` and `SCHWRZ=.TRUE.` (default) in `$INTGRL` if you use this option. Most of the QFMM-related options are controlled by the corresponding \$FMM group. Some keywords in the \$CONTRL group affect QFMM as well, namely `ICUT`, `ITOL`, `FSTINT` and `REORDR`.

This group controls the quantum fast multipole method evaluation of Fock matrices. The defaults are reasonable, so there is little need to give this input.

`ITGERR` = Target error in final energy, to  $10^{**-(ITGERR)}$  Hartree. The accuracy is usually better than the setting of `ITGERR`, in fact QFMM runs should suffer no loss of accuracy or be more accurate than a conventional integral run (default=7).

`QOPS` = a flag to use the Quantum Optimum Parameter Searching technique, which finds an optimum FMM parameter set. (Default=`.TRUE.`)

If `QOPS=.FALSE.`, the `ITGERR` value is not used. In this case the user should specify the following parameters:

`NP` = the highest multipole order for FMM (Default=15).

`NS` = the highest subdivision level (Default=2).

`IWS` = the minimum well-separateness (Default=2).

`IDPGD` = point charge approximation error ( $10^{**(-IDPGD)}$ ) of the Gaussian products (Default=9).

`IEPS` = very fast multipole method (vFMM) error, ( $10^{**(-IEPS)}$ ) (Default=9)

These are additional useful options which are either PC GAMESS specific or not documented in the GAMESS (US) manual due to bugs in their implementation:

`METHOD` = one of `DISK`, `SEMIDRCT`, or `FULLDRCT`. Controls disk vs CPU usage during first (FMM) part of the calculations. At present, `FULLDRCT` (fully direct) is equivalent to `SEMIDRCT` (semidirect). Semidirect uses less disk space and is usually faster than disk-based (`DISK`), especially for very large systems, and is the default.

`NUMRD` = (positive integer). Controls disk read caching during FMM, as well as the granularity of the static/dynamic load balancing during parallel QFMM runs. The default value is 10 and is reasonable in most cases.

`MODIFY` = a flag to allow QOPS code to modify `ICUT` and `ITOL` variables of \$CONTRL group. The default is `.false.`, i.e., not to modify them.

Although it can be set it to `.true.` for better compatibility with GAMESS (US) it is generally not recommended to activate this option.

`MQOPS` = 0 or 1. If set to zero (default) and when `QOPS=.TRUE.`, `SCLF` AND `NS` are determined automatically. If `MQOPS=1`, user provided values of `SCLF` and `NS` override those found by `QOPS`.

`SCLF` = FMM cube scaling factor, must be greather or equal to 1.00

`STATIC` = a flag to use static load balancing (SLB) during FMM part of calculations even if `DLB` is activated. Default is `.true.` because in the case of homogeneous environment the static load balancing is implemented much more efficient.

`NEARJ` = 0, 1, or 2. Selects the routine to calculate near field Coulomb terms.

`NEARJ=0` means to select an optimal default based on `FSTINT` & `REORDR` settings in `$CONTRL`.

`NEARJ=1` means use of the bugfixed/improved GAMESS (US)-based routine using `HONDO` integral package.

`NEARJ=2` means use of the PC GAMESS specific routine based on the `fastints` code which is generally much faster. It requires `FSTINT` and `REORDR` to be set in `$CONTRL`.

`LEX` = 0, 1, 2, or 3. Selects the routine to calculate HF exchange terms.

`LEX=0` means to select an optimal default based on the `FSTINT` & `REORDR` settings in `$CONTRL`, as well as on the molecular symmetry.

`LEX=1` means use of the bugfixed/improved GAMESS (US)-based routine using `HONDO` integral package. You should take into account that this implementation of linear exchange is by design to some degree approximate and is not fully equivalent to direct SCF, although in most cases one can safely neglect this fact.

`LEX=2` means use of the `fastints` based routine which evaluates some extra integrals but is the only part of the `QFMM` which can take into account the molecular symmetry. It is strictly equivalent to the direct SCF. For highly-symmetrical systems it is faster than any other available method. It requires `FSTINT` to be set in `$CONTRL`.

`LEX=3` means use of the `fastints` based routine which evaluates the minimal number of all the necessary 2-e integrals and is also strictly equivalent to direct SCF. It does not exploit molecular symmetry, though. For low-symmetry systems it is the fastest method available. It requires `FSTINT` and `REORDR` to be set in `$CONTRL`.

Default is `LEX=0`.

`SKIP1` = a flag to modify the behavior of the inner loop of the 2-e integral

selection code of the LEX=1 exchange routine. SKIP1=.true. means GAMESS (US)-style behavior. SKIP1=.false. makes it more precise by the cost of some CPU overhead. The default is .true. (see note at the end of the STRICT flag input description).

SKIP2 = a flag to modify the behavior of the outer loop of the 2-e integral selection code of the LEX=1 exchange routine. SKIP2=.true. means GAMESS (US)-style behavior. SKIP2=.false. makes it more precise by the cost of some CPU overhead. The default is .true. (see note at the end of the STRICT flag input description).

STRICT = a flag to modify the behavior of the density matrix sorting and nonzero elements selection for the LEX=1 exchange routine. STRICT=.false. means GAMESS (US)-style behavior. STRICT=.true. makes it more precise by the cost of some CPU overhead. The default is .false. because even if SKIP1=.false. SKIP2=.false. STRICT=.true. the LEX=1 routine is not exactly equivalent to direct SCF, while the CPU overhead is very significant.

=====

## ***\$TRANS group***

(optional for -CI- or -MCSCF-)  
(relevant to analytic Hessians)  
(relevant to energy localization)

This group controls the integral transformation. MP2 integral transformations are controlled instead by the \$MP2 input group. There is little reason to give any but the first variable.

DIRTRF = a flag to recompute AO integrals rather than storing them on disk. The default is .FALSE. for MCSCF and CI runs. If your job reads \$SCF, and you select DIRSCF=.TRUE. in that group, a direct transformation will be done, no matter how DIRTRF is set.

Note that the transformation may do many passes over the AO integrals for large basis sets, and thus the direct recomputation of AO integrals can be very time consuming.

MPTRAN = method to use for the integral transformation. the default is try 0, then 1, then 2.  
0 means use the incore method  
1 means use the segmented method. This is the only method that works in parallel.  
2 means use the alternate method, which uses less memory than 2, but requires an extra large disk file.

NWORD = Number of words of fast memory to allow. Zero uses all available memory. (default=0)

CUTTRF = Threshold cutoff for keeping transformed two electron integrals. (default= 10\*\*(-9))

AOINTS = defines AO integral storage during conventional integral transformations, during parallel runs. DUP stores duplicated AO lists on each node, and is the default for parallel computers with slow interprocessor communication, e.g. ethernet. DIST distributes the AO integral file across all nodes, and it is the default for parallel computers with high speed communications. DELETE means that AOINTS file is deleted when the integral transformation is done

IREST = 0 Normal operation  
 = 1 Only 1-e integrals are transformed, while 2-e integrals are read from the existing MOINTS file Default is 0  
 = 2 Both 1-e and 2-e integrals are read from the existing MOINTS file  
 =-1,-2 Similar to 1 and 2, but the effect is permanent for the whole calculation.

=====

## **\$nbo**

From the very beginning of its history, PC GAMESS included fully-functional NBO module. However, NBO part of the PC GAMESS requires activation as it is a commercial code. You should purchase PC GAMESS NBO license codes and NBO manual from TCI/NBO. The PC GAMESS' NBO license is very inexpensive (\$30).

To activate NBO module, you should add the following strings to your PC GAMESS input:  
 \$license nbolid=lid nbokey=key \$end  
 \$nbo < nbo options > \$end

Normally, you should receive both NBO activation key (nbokey, 8-digit hexadecimal code), and NBO license ID (nbolid, some decimal number) from TCI. Nevertheless, in many cases they forgot to provide users by nbolid value. If it is the case, please contact us as we can recover your nbolid using NBO activation key value by just looking in our NBO database.

Note that we have NBO v. 5.0 code at hand, nevertheless, the PC GAMESS at moment has older version of NBO (4.M) incorporated in it. We plan to upgrade NBO code in the PC GAMESS to NBO v. 5.0 soon enough. The purchased PC GAMESS NBO license will be valid for both versions 4.M and 5.x of NBO code.

=====

=====  
The remaining groups apply only to -CI- and -MCSCF- runs.

\* \* \* \* \*  
For hints on how to do -CI- and -MCSCF-  
see the 'further information' section  
\* \* \* \* \*

=====  
**\$CIINP group** (optional, relevant for CITYP=GUGA or ALDET)

This group is the control box for Graphical Unitary Group Approach (GUGA) CI calculations, or Ames Laboratory determinant (ALDET) full CI. Each step which is executed potentially requires a further input group described later.

NRNFG = An array of 10 switches controlling which steps of a CI computation are performed.

1 means execute the module, 0 means don't.

- NRNFG(1) = Generate the configurations. See either \$CIDRT or \$CIDET input. (default=1)
- NRNFG(2) = Transform the integrals. See \$TRANS. (default=1)
- NRNFG(3) = Sort integrals and calculate the Hamiltonian matrix. See \$CISORT and \$GUGEM. (default=1)  
This does not apply to ALDET.
- NRNFG(4) = Diagonalize the Hamiltonian matrix. See \$GUGDIA or \$CIDET. (default=1)
- NRNFG(5) = Construct the one electron density matrix, and generate NO's. See \$GUGDM or \$CIDET. (default=1)
- NRNFG(6) = Construct the two electron density matrix. See \$GUGDM2 or \$CIDET. (default=0 normally, but 1 for CI gradients)
- NRNFG(7) = Construct the Lagrangian of the CI function. Requires DM2 matrix exists. See \$LAGRAN. (default=0 normally, but 1 for CI gradients)  
This does not apply to ALDET.
- NRNFG(8-10) are not used.

Users are not encouraged to change these values, as the defaults are quite reasonable ones.

NPFLG = An array of 10 switches to produce debug printout. There is a one to one correspondance to NRNFG, set to 1 for output. (default = 0,0,0,0,0,0,0,0,0,0)  
The most interesting is NPFLG(2)=1 to see the transformed 1e- integrals, NPFLG(2)=2 adds the very numerous transformed 2e- integrals to this.

IREST = n Restart the -CI- at stage NRNFG(n).

CASTRF= .t. Selects fast MCSCF-like integral transformation  
for standalone CI runs.

=====  
**\$CIS group**

required when CITYP=CIS

The CIS method (singly excited CI) is the simplest way to treat excited states. By Brillouin's Theorem, a single determinant reference such as RHF will have zero matrix elements with singly substituted determinants. The ground state reference therefore has no mixing with the excited states treated with singles only. Reading the references given in Section 4 of this manual will show the CIS method can be thought of as a non-correlated method, rigorously so for the ground state, and effectively so for the various excited states. Some issues making CIS rather less than a black box method are:

- a) any states characterized by important doubles are simply missing from the calculation.
- b) excited states commonly possess Rydberg (diffuse) character, so the AO basis used must allow this.
- c) excited states often have different point group symmetry than the ground state, so the starting geometries for these states must reflect their actual symmetry.
- d) excited state surfaces frequently cross, and thus root flipping may very well occur.

The CIS code in the PC GAMESS is based on heavily modified sources of the original GAMESS (US) AO-basis CIS code written by Simon P. Webb

The implementation allows the use of only RHF references, but can pick up both singlet and triplet excited states. Nuclear gradients are available, as are properties.

NCORE = n Omits the first n occupied alpha and beta orbitals from the calculation. The default for n is the number of chemical core orbitals.

NSTATE = Number of states to be found (excluding the ground state).

ISTATE = State for which properties and/or gradient will be calculated. Only one state can be chosen.

HAMTYP = Type of CI Hamiltonian to use.  
= SAPS spin-adapted antisymmetrized product of the desired MULT will be used (default)  
= DETS determinant based, so both singlets and triplets will be obtained. Note that this option will disable use of FASTINTS/GENCON code during direct CIS runs.

MULT = Multiplicity (1 or 3) of the singly excited SAPS (the reference is necessarily single RHF). Only relevant for SAPS based run.

DIAGZN = Hamiltonian diagonalization method.  
 = DAVID use Davidson diagonalization. (default)  
 = FULL construct the full matrix in memory and diagonalize, thus determining all states (not recommended except for small cases).

DGAPRX = Flag to control whether approximate diagonal elements of the CIS Hamiltonian (based only on the orbital energies) are used in the Davidson algorithm. Note, this only affects the rate of convergence, not the resulting final energies. If set .FALSE., the exact diagonal elements are determined and used. Default=.TRUE.

NGSVEC = Dimension of the Hamiltonian submatrix that is diagonalized to form the initial CI vectors. The default is the greater of NSTATE\*2 and 10.

MXVEC = Maximum number of expansion basis vectors in the iterative subspace during Davidson iterations, before the expansion basis is truncated. The default is the larger of 8\*NSTATE and NGSVEC.

NDAVIT = Maximum number of Davidson iterations. Default=50.

DAVCVG = Convergence criterion for Davidson eigenvectors. Eigenvector accuracy is proportional to DAVCVG, while the energy accuracy is proportional to its square. The default is 1.0E-05.

CISPRP = Flag to request the determination of CIS level properties, using the relaxed density. Relevant to RUNTYP=ENERGY jobs, although the default is .FALSE. because additional CPHF calculation will be required. Properties are computed as a normal byproduct of runs involving the CIS gradient.

CHFSLV = Chooses type of CPHF solver to use.  
 = CONJG selects an ordinary preconditioned conjugate gradient solver. This is the default.  
 = DIIS selects a diis-like iterative solver.

RDCISV = Flag to read CIS vectors from a \$CISVEC group in the input file. Default is .FALSE.

MNMEDG = Flag to force the use of the minimal amount of memory in construction of the CIS Hamiltonian diagonal elements. This is only relevant when DGAPRX=.FALSE., and is meant for debug purposes. The default is .FALSE.



MNMEOP = Flag to force the use of the minimal amount of memory during the Davidson iterations. This is for debug purposes. The default is .FALSE.

-----  
The additional PC GAMESS (Firefly) specific keywords are:

CPTOL - convergence criterion for CPHF equations, default is 1.0d-5 for GAMESS (US) compatibility.

THRDII - threshold to turn on DIIS if it was selected to solve CPHF. Default is 0.05

MAXGC - maximum allowed number of trial vectors to be routed through GENCON engine, default is 1. If the number of trial vectors is greater than MAXGC, only FASTINTS will be used. The reason is that for moderately contracted GC basis sets like cc-pVXZ, gencon is faster than fastints only for relatively small number of trial vectors (this is by gencon design). On the other hand, for ANO-like basis sets, it is always better to set MAXGC to be equal the number of initial guess vectors, as fastints will be much slower.

ONEDIM = .true./.false. Default is true for abelian groups, false otherwise. Option to select fast algorithm for excited state properties/gradients for systems with symmetry. Any state transforming by any pure one-dimension representation on the point group can be handled by this manner. It is safe to always turn this option on as the check is performed and this option disables itself if the symmetry is not appropriate.

PRTTOL - threshold for CIS determinants/csf printout and also for states symmetry determination. Default is 0.05.

ISTSYM - symmetry of states of interest. Default is zero, i.e., does not use any symmetry during calculations. Setting this to the desired index of irrep(according to GAMESS numbering) will solve only for the states of the desired symmetry and exploiting full (including non-abelian) symmetry of molecule, thus significantly reducing computation time.

ALTER - flag to modify internal logic of Davidson diagonalization code to use dynamic number of trial vectors. Default is .true. Setting it to .false. will slow-down calculations by forcing DAVIDSON diagonalization code to work exactly as in GAMESS (US).

The state-tracking feature of the PC GAMESS' CIS code can be activated by selecting negative value of istate in the \$cis group. It is intended for geometry optimization of the excited states in the case of root flipping.  
Note that oscillator strengths printed in the CIS summary table are calculated using transition dipoles length form only.

=====  
**\$CISVEC group** required if RDCISV in \$CIS is chosen

Below is the sample input file.

```
$CONTRL SCFTYP=RHF CITYP=CIS $END
$SYSTEM TIMLIM=3000 MEMORY=3000000 $END
$BASIS GBASIS=n31 ngauss=6 NDFUNC=1 $END
$CIS NSTATE=3 ISTSYM=0 ISTATE=1 $END
$DATA
H2O
CNV 2

O          8.0    0.0000000000    0.0000000000    0.7205815395
H          1.0    0.0000000000    0.7565140024    0.1397092302
$END
```

```
=====
$DET group          (required for SCFTYP=MCSCF if CISTEP=ALDET)
$CIDET group       (required if CITYP=ALDET)
```

This group describes the determinants to be used in a full MCSCF active space, or full CI wavefunction.

Determinants contain several spin states, in contrast to configuration state functions. The Sz quantum number of each determinant is the same, but the Hamiltonian eigenvectors will have various spins  $S=S_z, S_z+1, S_z+2,$

In addition, the determinant CI code does not exploit the spatial symmetry of the orbitals, so the CI states will include not only various spin symmetries, but also all space symmetries. You may need to think about NSTATE!

There is no default for NCORE, NACT, and NELS:

NCORE = total number of orbitals doubly occupied in all determinants.

NACT = total number of active orbitals.

NELS = total number of active electrons.

SZ = azimuthal spin quantum number for each of the determinants, two times SZ is therefore the number of excess alpha spins in each determinant. The default is  $SZ=S$ , extracted from the  $MULT=2S+1$  given in \$CONTRL.

\* \* \* the following control the diagonalization \* \* \*

NSTATE = Number of CI states to be found, the default is 1. The maximum number of states is 100.

PRTTOL = Printout tolerance for CI coefficients, the default is to print any larger than 0.05.

ITERMX = Maximum number of Davidson iterations per root. The default is 100.

CVGTOL = Convergence criterion for Davidson eigenvector routine. This value is proportional to the accuracy of the coefficients of the eigenvectors found. The energy accuracy is proportional to its square. The default is 1.0E-5.

NHGSS = dimension of the Hamiltonian submatrix which is diagonalized to obtain the initial guess eigenvectors. The determinants forming the submatrix are chosen on the basis of a low diagonal energy, or if needed to complete a spin eigenfunction. The default is 300.

NSTGSS = Number of eigenvectors from the initial guess Hamiltonian to be included in the Davidson's iterative scheme. It is seldom necessary to include extra states to obtain convergence to the desired states. The default is the value for NSTATE.

MXXPAN = Maximum number of expansion basis vectors in the iterative subspace during the Davidson iterations before the expansion basis is truncated. The default is the larger of 10 or NSTGSS+1. If this value is decreased, convergence is hurt but the memory required is decreased.

\* \* \* the following control the 1st order density \* \* \*  
 These are ignored during MCSCF, but are used during a CI.

IROOT = the root whose density is saved on the disk file for subsequent property analysis. Only one root can be saved, and the default value of 1 means the ground state. Be sure to set NFLGDM to form the density of the state you are interested in!

NFLGDM = Controls each state's density formation.  
 0 -> do not form density for this state.  
 1 -> form density and natural orbitals for this state, print and punch occ.num. and NOs.  
 2 -> same as 1, plus print density over MOs.  
 The default is NFLGDM(1)=1,0,0,...,0 meaning only ground state NOs are generated.

\* \* \* the following control the state averaged  
 \* \* \* 1st and 2nd order density matrix computation  
 Usually ignored by CI runs, these are relevant to MCSCF.

PURES = a flag controlling the spin purity of the state averaging. If true, the WSTATE array pertains

to the lowest states of the same S value as is given by the MULT keyword in \$CONTRL. In this case the value of NSTATE will need to be bigger than the total number of weights given by WSTATE if there are other spin states present at low energies. If false, it is possible to state average over more than one S value, which might be of interest in spin-orbit coupling jobs. The default is .TRUE.

WSTATE = An array of up to 100 weights to be given to the densities of each state in forming the average. The default is to optimize a pure ground state, WSTATE(1)=1.0,0.0,...,0.0  
A small amount of the ground state can help the convergence of excited states greatly.  
Gradient runs are possible only with pure states. Be sure to set NSTATE above appropriately!

=====

**\$DRT group** (required for SCFTYP=MCSCF if CISTEP=GUGA)  
**\$CIDRT group** (required if CITYP=GUGA)

This group describes the -MCSCF- or -CI- wavefunction. The distinct row table is the means by which the Graphical Unitary Group Approach (GUGA) names the configurations.

The group is spelled DRT for MCSCF runs, and CIDRT for CI runs. The main difference in these is NMCC vs. NFZC.

There is no default for GROUP, and you must choose one of FORS, FO CI, SOCI, or IEXCIT.

GROUP = the name of the point group to be used. This is usually the same as that in \$DATA, except for RUNTYP=HESSIAN, when it must be C1. Choose from the following: C1, C2, CI, CS, C2V, C2H, D2, D2H, C4V, D4, D4H. If your \$DATA group is not listed, choose only C1 here.

FORS = flag specifying the Full Optimized Reaction Space set of configuration should be generated. This is usually set true for MCSCF runs, but if it is not, see FORS in \$MCSCF. (Default=.FALSE.)

FOCI = flag specifying first order CI. In addition to the FORS configurations, all singly excited CSFs from the FORS reference are included. Default=.FALSE.

SOCI = flag specifying second order CI. In addition to the FORS configurations, all singly and doubly excited configurations from the FORS reference

are included. (Default=.FALSE.)

IEXCIT= electron excitation level, for example 2 will lead to a singles and doubles CI. This variable is computed by the program if FORS, FOCI, or SOCI is chosen, otherwise it must be entered.

\* \* the next variables define the single reference \* \*

The single configuration reference is defined by filling in the orbitals by each type, in the order shown. The default for each type is 0.

Core orbitals, which are always doubly occupied:  
NMCC = number of MCSCF core MOs (in \$DRT only).  
NFZC = number of CI frozen core MOs (in \$CIDRT only).

Internal orbitals, which are partially occupied:  
NDOC = number of doubly occupied MOs in the reference.  
NAOS = number of alpha occupied MOs in the reference, which are singlet coupled with a corresponding number of NBOS orbitals.  
NBOS = number of beta spin singly occupied MOs.  
NALP = number of alpha spin singly occupied MOs in the reference, which are coupled high spin.  
NVAL = number of empty MOs in the reference.

External orbitals, occupied only in FOCI or SOCI:  
NEXT = number of external MOs. If given as -1, this will be set to all remaining orbitals (apart from any frozen virtual orbitals).  
NFZV = number of frozen virtual MOs, never occupied.

\* \* \* the final choices are seldom used \* \* \*

INTACT= flag to select the interacting space option. The CI will include only those spin couplings which have a nonvanishing matrix element with the reference configuration.

MXNINT = Buffer size for sorted integrals. (default=20000)

MXNEME = Buffer size for energy matrix. (default=10000)

NPRT = Configuration printout control switch.  
This can consume a HUMUNGUS amount of paper!  
0 = no print (default)  
1 = print electron occupancies, one per line.  
2 = print determinants in each CSF.

=====

**\$MCSCF group**

(optional for -MCSCF-)

This group controls the MCSCF orbital optimization step. The difference between the four convergence methods is outlined in Chapter Four of this manual, which you must carefully study before attempting MCSCF computations.

OPTACT = .False. Default  
= .True. Activates minor changes in the MCSCF FULLNR converger. This may somehow help in the case of a slow convergence of FULLNR MCSCF for the non FORS-type wavefunction.

--- the next choose the configuration basis ---

CISTEP = ALDET chooses the Ames Lab. determinant CI, and requires \$DET input. (default)  
= GUGA chooses the graphical unitary group CSFs, and requires \$DRT input. This is the only value usable with the QUAD converger.

--- the next four choose the orbital optimizer ---

FOCAS = a flag to select a method with a first order convergence rate. (default=.FALSE.)

SOSCF = a flag selecting an approximately second order convergence method. (default=.TRUE.)

FULLNR = a flag selecting a second order method, with an exact orbital hessian. (default=.FALSE.)

QUAD = a flag to pick a fully quadratic (orbital and CI coefficient) optimization method, which is applicable to FORS or non-FORS wavefunctions. QUAD may not be used with state-averaging. (default = .FALSE.)

Note that FOCAS must be used only with FORS=.TRUE. in \$DRT.

The other convergers are usable for either FORS or non-FORS wavefunctions, although convergence is always harder in the latter case, when FORS below must be set .FALSE.

--- the next apply to all convergence methods ---

FORS = a flag to specify that the MCSCF function is of the Full Optimized Reaction Space type, which is sometimes known as CAS-SCF. .TRUE. means omit act-act rotations from the optimization. Since convergence is usually better for FULLNR with these rotations included, the default is sensible for the case FORS=.TRUE. in \$DRT. (default is .TRUE. for FOCAS/SOSCF, .FALSE. for FULLNR/QUAD)

ACURCY = the major convergence criterion, the maximum permissible asymmetry in the Lagrangian matrix. (default=1.0E-05)

ENGTOL = a secondary convergence criterion, the run is considered converged when the energy change is smaller than this value. (default=1.0E-10)

MAXIT = Maximum number of iterations (default=100 for FOCAS, 60 for SOSCF, 30 for FULLNR or QUAD)

MICIT = Maximum number of microiterations within a single MCSCF iteration. (default=5 for FOCAS or SOSCF, or 1 for FULLNR or QUAD)

NWORD = The maximum memory to be used, the default is to use all available memory. (default=0)

CANONC = a flag to cause formation of the closed shell Fock operator, and generation of canonical core orbitals. This will order the MCC core by their orbital energies. (default=.TRUE.)

EKT = a flag to cause generation of extended Koopmans' theorem orbitals and energies. (Default=.FALSE.)  
For this option, see R.C.Morrison and G.Liu, J.Comput.Chem., 13, 1004-1010 (1992). Note that the process generates non-orthogonal orbitals, as well as physically unrealistic energies for the weakly occupied MCSCF orbitals. The method is meant to produce a good value for the first I.P.

NPUNCH = MCSCF punch option (analogous to \$SCF NPUNCH)  
0 do not punch out the final orbitals  
1 punch out the occupied orbitals  
2 punch out occupied and virtual orbitals  
The default is NPUNCH = 2.

NPFLG = an array of debug print control. This is analagous to the same variable in \$CIINP. Elements 1,2,3,4,6,8 make sense, the latter controls debugging the orbital optimization.

--- the next refers to SOSCF optimizations ---

NOFO = set to 1 to skip use of FOCAS for one iteration during SOSCF. This is a testing parameter, at present NOFO defaults to 0 to do one FOCAS iter.

--- the next three refer to FOCAS optimizations ---

CASDII = threshold to start DIIS (default=0.05)

CASHFT = level shift value (default=1.0)

NRMCAS = renormalization flag, 1 means do Fock matrix

renormalization, 0 skips (default=1)

--- the next applies to the QUAD method ---  
(note that FULLNR input is also relevant)

QUATHR = threshold on the orbital rotation parameter,  
SQCDF, to switch from the initial FULLNR  
iterations to the fully quadratic method.  
(default = 0.05)

--- all remaining input applies only to FULLNR ---

DAMP = damping factor, this is adjusted by the program  
as necessary. (default=0.0)

METHOD = DM2 selects a density driven construction of the  
Newton-Raphson matrices. (default).  
= TEI selects 2e- integral driven NR construction.  
See the 'further information' section for more  
details concerning these methods. TEI is slow!

LINSER = a flag to activate a method similar to direct  
minimization of SCF. The method is used if  
the energy rises between iterations. It may in  
some circumstances increase the chance of  
converging excited states. (default=.FALSE.)

FCORE = a flag to freeze optimization of the MCC core  
orbitals, which is useful in preparation for  
RUNTYP=TRANSITN jobs. Setting this flag will  
automatically force CANONC false. This option  
is incompatible with gradients, so can only be  
used with RUNTYP=ENERGY. (default=.FALSE.)

--- the next four are seldom used ---

DROPC = a flag to include MCC core orbitals during the  
CI computation. The default is to drop them  
during the CI, instead forming Fock operators  
which are used to build the correct terms in  
the orbital hessian. (default = .TRUE.)

NORB = the number of orbitals to be included in the  
optimization, the default is to optimize with  
respect to the entire basis. This option is  
incompatible with gradients, so can only be used  
with RUNTYP=ENERGY. (default=number of AOs  
given in \$DATA).

MOFRZ = an array of orbitals to be frozen out of the  
orbital optimization step (default=none frozen).

NOROT = an array of up to 250 pairs of orbital rotations



to be omitted from the NR optimization process. The program automatically deletes all core-core rotations, all act-act rotations if FORS=.T., and all core-act and core-virt rotations if FCORE=.T. Additional rotations are input as I1,J1,I2,J2... to exclude rotations between orbital I running from 1 to NORB, and J running up to the smaller of I or NVAL in \$TRANS.

=====

**\$MCQDPT group** (relevant to SCFTYP=MCSCF if MLEVEL=2)

Controls 2nd order MCQDPT (multiconfiguration quasi-degenerate perturbation theory) runs, if requested by MLEVEL=2 in \$CONTRL. MCQDPT2 is implemented only for FORS (aka CASSCF) wavefunctions. The MCQDPT method is a multistate, as well as multireference perturbation theory. The implementation is a separate program, interfaced to GAMESS, with its own procedures for determination of the canonical MOs, CSF generation, integral transformation, CI in the reference CAS, etc. Therefore some of the input in this group repeats data given elsewhere, particularly the \$DET/\$DRT. A more complete discussion may be found in the 'Further Information' chapter. Analytic gradients are not available.

\*\*\* MCSCF reference wavefunction \*\*\*

NEL = total number of electrons, including core.  
 (default from \$DATA and ICHARG in \$CONTRL)

MULT = spin multiplicity (default from \$CONTRL)

NMOACT = Number of orbitals in FORS active space  
 (default is the active space in \$DET or \$DRT)

NMOFZC = number of frozen core orbitals, NOT correlated  
 in the perturbation calculation. (default is  
 number of chemical cores)

NMODOC = number of orbitals which are doubly occupied in  
 every MCSCF configuration, that is, not active  
 orbitals, which are to be included in the  
 perturbation calculation. (The default is all  
 valence orbitals between the chemical core and  
 the active space)

NMOFZV = number of frozen virtuals, NOT occupied during  
 the perturbation calculation. The default is  
 to use all virtuals in the MP2. (default=0)

ISTSYM = the state symmetry of the target state(s).  
 This is given as an integer, note that only  
 Abelian groups are supported in \$DATA:

ISTSYM=	1	2	3	4	5	6	7	8
	C1	A						
	Ci	Ag	Au					
	Cs	A'	A''					
	C2	A	B					

C2v A1 A2 B1 B2  
C2h Ag Bg Au Bu  
D2 A B1 B2 B3  
D2h Ag B1g B2g B3g Au Blu B2u B3u  
(The default is 1, the totally symmetric state)

\*\*\* perturbation specification \*\*\*

KSTATE= state is used (1) or not (0) in the MCQDPT2.  
Maximum of 20 elements, including zeros.  
For example, if you want the perturbation  
correction to the second and the fourth roots,  
KSTATE(1)=0,1,0,1  
(default=1,0,0,0,0,0,0,0,...)

\*\*\* MO input and flow control \*\*\*

INORB = 0 optimize the MCSCF wavefunction in this run.  
= 1 read the converged orbitals from a \$VEC group,  
and skip immediately to the MCQDPT computation.  
A complete \$VEC including virtuals must be given.  
(default=0)

\*\*\* Canonical Fock orbitals \*\*\*

IFORB = 1 determine the canonical Fock orbitals  
= 0 omit this step. (default=1)  
AVECOE = weight of each CAS-CI state in computing the  
closed shell Fock matrix.  
(default is AVECOE(1)=1.0,1.0,1.0,...)

\*\*\* Intruder State Removal \*\*\*

EDSHFT = energy denominator shift. (default=0.0)  
Intruder State Avoidance (ISA) calculations  
can be made by changing the energy denominators  
around poles (where the denominator is zero).  
Each denominator  $x$  is replaced by  $x + \text{EDSHFT}/x$ ,  
so that far from the poles (when  $x$  is large) the  
effect of such change is small. A suggested  
value is 0.02, but experimentation with your system  
is recommended. Setting this value to zero is ordinary  
MCQDPT, and infinite collapses to the MCSCF reference.  
Note that the energy denominators (which are ket-dependent  
in MCQDPT) are changed in a different way for each  
ket-vector, that is, for each row in MCQDPT  
Hamiltonian matrix. In other words, the zeroth  
order energies are not "universal", but state  
specific. This is strictly speaking some weak  
inconsistency in defining zeroth order energies  
that are usually chosen "universally".

In order to maintain continuity when studying

a PES, one usually uses the same EDSHFT value for all points on PES. In order to study the potential surface for any extended range of geometries, it is recommended to use ISA, as it is quite likely that one or more regions of the PES will be unphysical due to intruder states.

For an example of how intruder states can appear at some points on the PES, see Figures 1,2,7 of K.R.Glaesemann, M.S.Gordon, H.Nakano Phys.Chem.Chem.Phys. 1, 967-975(1999)

and also

H.A.Witek, D.G.Fedorov, K.Hirao, A.Viel, P.-O.Widmark J.Chem.Phys. 116, 8396-406(2002)

For a discussion of intruder state removal from MCQDPT, see

H.A.Witek, Y.-K.Cho, J.P.Finley, K.Hirao J.Comput.Chem. 23, 957-965(2002)

\*\*\* Miscellaneous options \*\*\*

THRGEN = threshold for one-, two-, and three-body density matrix elements in the perturbation calculation. If you want to obtain energies, for instance, to 6 figures after point, choose THRGEN=1.0D-08 or 1.0D-09. (default=1.0D-08)

THRENE = threshold for the energy convergence in the Davidson's method CAS-CI. (default=-1.0D+00)

THRCON = threshold for the vector convergence in the Davidson's method CAS-CI. (default=1.0D-06)

LPOUT = print option, 0 gives normal printout, while <0 gives debug print (e.g. -1,-5,-10,-100) (default=0)

Finally, there are additional very specialized input options, described in the source code routine MQREAD:

IROT, ISELCT, LENGTH, MAXCSF, MAXERI, MAXROW, MDI, MXBASE, MXTRFR, NSOLUT, NSTOP, GENZRO, THRERI, THRWGT, MAINCS, NSTATE, NSTCI

=====

## ***\$MCQFIT group***

The presence of this group in the input file triggers the new MCQDPT2 code. The main purpose of the new MCQDPT2 code is to speed up calculations in the cases of medium to large basis sets and medium to large active spaces. It is not efficient for small active spaces (e.g., less than 500 CSF) and/or for small number of basis functions (e.g., less than 50). In the latter cases it is recommended to use the default PC GAMESS' MCQDPT2 code. It is also not recommended to use new code for calculations which do not use ISA energy denominators shift, as in the cases of

intruder states new code can become numerically unstable. On the other hand, we recommend to use ISA technique/nonzero edshft parameter for **any** MCQDPT2 run.

In most cases, it is sufficient to just specify \$mcqfit \$end without altering any parameters of the \$mcqfit group - the default parameters were selected to introduce errors in the computed energies which is typically less than  $10^{-8}$  Hartree.

For better efficiency of the new MCQDPT2 code running in SMP mode, we recommend to manually increase the default value of the *length* parameter of \$mcqdpt group to be equal to the number of CSF generated by your run.

DELTAE - double precision, default is zero. If nonzero, this value is used to define parameters of the interpolation grid.

NPOINT - integer, default is 400. Together with deltae this value is used to define parameters of the interpolation grid. Actual grid will be the smaller of npoint value and the number of points computed based on the deltae value. Increasing npoint will result in increase of required CPU time but will reduce errors in computed energy.

IORDER - integer, default value is 7. Available values are 3, 5, and 7. Defines the order of polynomes used for interpolation.

=====

**\$CISORT group** (optional, relevant for -CI- and -MCSCF-)

This group provides further control over the sorting of the transformed integrals.

NDAR = Number of direct access records.  
(default = 2000)

LDAR = Length of direct access record (site dependent)

NBOXMX = Maximum number of boxes in the sort.  
(default = 200)

NWORD = Number of words of fast memory to use in this step. A value of 0 results in automatic use of all available memory. (default = 0)

NOMEM = 0 (set to one to force out of memory algorithm)

MOINTS = DELETE Delete MOINTS file after sorting

=====

## **\$GUGDRT group**

NWORD = Controls the memory usage during generation of DRT. It may be necessary to increase the default value of 180018 DP words when the active space used in CI/MCSCF calculation is too large.

=====

## **\$GUGEM group** (optional, relevant for -CI- or -MCSCF-)

This group provides further control over the calculation of the energy (Hamiltonian) matrix.

CUTOFF = Cutoff criterion for the energy matrix.  
(default=1.0E-8)

NWORD = not used.

FASTCI =  
= .False. Normal operation  
= .True. Turns on the packing of GUGA CI Hamiltonian file. Both indices and matrix elements are packed. This option reduces the size of WORK16 file by a factor of up to 2.7-3, thus saving the disk space and the CPU time. See also a more detailed discussion on the packing-related problems. (default)

PKTHR = A threshold to activate packing of the matrix elements of GUGA CI Hamiltonians. Default is  $10^7 * \text{CUTOFF}$

PACK2 =.False. Normal operation (default)  
=.True. Turns on the additional (secondary) packing of GUGA CI Hamiltonian file, which works in conjunction with the FASTCI method of compression. This is a lossless method, which generally reduces the size of WORK16 file by a factor of up to 3-6 additionally to the FASTCI squeezing. Thus, the total degree of compression can reach the factor of 10-20 and even more, especially for the non-FORS type CI. Hence, this option is very useful in the case of relatively large CI calculations (i.e. 50000 CSFs or more, up to several millions), saving the disk space and reducing the real execution time. **Currently, the fastest way to perform any large CI/MCSCF calculation is to use PACK2 method of packing. The only exception is the case of GUGA CI Hamiltonians of medium size, that can entirely reside in the system file cache (in the case of Win95, WinNT, and Linux, the size of the latter is limited only by the amount of the physical memory currently unused). For such jobs, the default packing settings are still faster.**  
It is often faster to perform a CI energy

calculation as the first iteration of the *approximate second-order MCSCF program*. This is due to the fact that the SOSCF MCSCF integral transformation can be significantly faster as compared to the generic one. Relevant only when FASTCI=.True.

LCHAIN = A parameter that controls the degree of compression achieved by the PACK2 packing method. The greater the LCHAIN value, the better the compression and the higher the CPU overhead. Note also that the most efficient packing is always achieved when LCHAIN=0, because this special value is simply an alias for the infinitely large LCHAIN. Relevant only when PACK2=.True. Default is 7, which seems to be quite reasonable.

DIRCI = .True. Performs the direct GUGA CI calculation, in which the Hamiltonian matrix is not stored on disk, and the matrix elements are recalculated during each Davidson diagonalization step. This may consume considerably more time than normal GUGA CI calculation, but allows one to perform it in the case when the Hamiltonian matrix is too large to fit on disk.

= .False. Performs conventional GUGA CI calculation (default)

=====

## **\$GUGDIA group** (optional, relevant for -CI- or -MCSCF-)

This group provides control over the Davidson method diagonalization step.

NSTATE = Number of CI states to be found. (default=1)  
You can solve for any number of states, but only 100 can be saved for subsequent sections, such as state averaging.

PRTTOL = Printout tolerance for CI coefficients  
(default = 0.05)

MXXPAN = Maximum no. of expansion basis vectors used before the expansion basis is truncated.  
(default=30)

ITERMX = Maximum number of iterations (default=50)

CVGTOL = Convergence criterion for Davidson eigenvector routine. This value is proportional to the accuracy of the coefficients of the eigenvector(s) found. The energy accuracy is proportional to its square. (default = 1.0E-5)

NWORD = Number of words of fast memory to use in this step. A value of zero results in the use of all available memory. (default = 0)

MEMMAX = 1000\*limgiv + limh where limgiv is the largest matrix diagonalized via Givens-Householder (default=50) and limh is the dimension of the largest Hamiltonian that may be memory resident (default=100)

NIMPRV = Maximum no. of eigenvectors to be improved every iteration. (default = nstate)

NSELCT = Determines initial guess to eigenvectors.  
 = 0 -> Unit vectors corresponding to the NSTATE lowest diagonal elements and any diagonal elements within SELTHR of them. (default)  
 < 0 -> First abs(NSELCT) unit vectors.  
 > 0 -> use NSELCT unit vectors corresponding to the NSELCT lowest diagonal elements.

SELTHR = Guess selection threshold when NSELCT=0. (default=0.01)

NEXTRA = Number of extra expansion basis vectors to be included on the first iteration. NEXTRA is decremented by one each iteration. This may be useful in "capturing" vectors for higher states. (default=5)

KPRINT = Print flag bit vector used when NPFLG(4)=1 in the \$CIINP group (default=8)  
 value 1 bit 0 print final eigenvalues  
 value 2 bit 1 print final tolerances  
 value 4 bit 2 print eigenvalues and tolerances at each truncation  
 value 8 bit 3 print eigenvalues every iteration  
 value 16 bit 4 print tolerances every iteration

=====

## ***\$GUGDM group***

(optional, relevant for -CI-)

This group provides further control over formation of the 1-particle density matrix.

NFLGDM = Controls each state's density formation.  
 0 -> do not form density for this state.  
 1 -> form density and natural orbitals for this state, print and punch occ.num. and NOs.  
 2 -> same as 1, plus print density over MOs. (default=1,99\*0, meaning G.S. NOs only)  
 See also NSTATE in \$GUGDIA. Note that forming

the 1-particle density for a state is negligible against the diagonalization time for that state.

IROOT = The -CI- root whose density matrix is saved on the direct access dictionary file for later computation of properties. (default=1)

IBLOCK = Density blocking switch. If nonzero, the off diagonal block of the density below row IBLOCK will be set to zero before the (approximate) natural orbitals are found. One use for this is to keep the internal and external orbitals in a FOCI or SOCI calculation from mixing. (default=0)

NWORD = Number of words of fast memory to use in this step. A value of zero uses all available memory (default=0).

=====

### ***\$GUGDM2 group*** (optional, relevant for -CI- or -MCSCF-)

This group provides control over formation of the 2-particle density matrix.

WSTATE = An array of up to 100 weights to be given to the 2 body density of each state in forming the DM2. The default is to optimize a pure ground state. (Default=1.0,99\*0.0)  
A small amount of the ground state can help the convergence of excited states greatly.  
Gradient runs are possible only with pure states.

Be sure to set NSTATE in \$GUGDIA appropriately!

CUTOFF = Cutoff criterion for the 2nd-order density. (default = 1.0E-9)

NWORD = Number of words of fast memory to use in sorting the DM2. The default uses all available memory. (default=0).

NOMEM = 0 uses in memory sort, if possible.  
= 1 forces out of memory sort.

NDAR = Number of direct access records. (default=4000)

LDAR = Length of direct access record (site dependent)

NBOXMX = Maximum no. of boxes in the sort. (default=200)

=====



**\$LAGRAN group** (optional, relevant for -CI- gradient)

This group provides further control over formation of the CI Lagrangian, a quantity which is necessary for the computation of CI gradients.

NOMEM = 0 form in core, if possible  
          = 1 forces out of core formation  
NWORD = 0 (0=use all available memory)  
NDAR = 4000  
LDAR = Length of each direct access record  
          (default is NINTMX from \$INTGRL)

=====  
**\$TRFDM2 group** (optional, relevant for -CI- gradient)

This group provides further control over the back transformation of the 2 body density to the AO basis.

NOMEM = 0 transform and sort in core, if possible  
          = 1 transform in core, sort out of core, if poss.  
          = 2 transform out of core, sort out of core  
NWORD = 0 (0=use all available memory)  
CUTOFF= 1.0D-9, threshold for saving DM2 values  
NDAR = 2000  
LDAR = Length of each direct access record  
          (default is system dependent)  
NBOXMX= 200

=====  
Usually neither of these two groups is given. Since these groups are normally used only for CI gradient runs, we list here some of the restrictions on the CI gradients:

- a) SCFTYP=RHF, only
- b) no FZV orbitals in \$CIDRT, all MOs must be used.
- c) the derivative integrals are computed in the 2nd derivative code, which is limited to spd basis sets.
- d) the code does not run in parallel.
- e) Use WSTATE in \$GUGDM2 to specify the state whose gradient is to be found. Use IROOT in \$GUGDM to specify the state whose other properties will be found. These must be the same state!
- f) excited states often have different symmetry than the ground state, so think about GROUP in \$CIDRT.
- g) the gradient can probably be found for any CI for which you have sufficient disk to do the CI itself. Time is probably about 2/3 additional.

=====

## **\$TRANST group**

(only for CITYP=GUGA)

relevant for RUNTYP=TRANSITN)

(relevant for RUNTYP=SPINORBT)

This group controls the evaluation of the radiative transition moment, or spin orbit coupling. The defaults assume that there is one common set of orbitals, all of which are occupied. The program can use two separately optimized MO sets, provided certain conditions are met.

NUMVEC = the number of different MO sets. This can be either 1 or 2. (default=1)

NUMCI = the number of different CI calculations to do. This should equal NUMVEC for TRANSITN, and is usually 2 or greater for SPINORBT (1 if the multiplicities of the states are the same). (default=1)

NOCC = the number of occupied orbitals. The default is the number of AOs given in \$DATA, which is not usually correct.

NFZC = When NUMVEC=2, this is the number of identical core orbitals in the two vector sets, and most likely this is NFZC in the \$CDIRT groups. When NUMVEC=1, and for RUNTYP=TRANSITN, NFZC should equal NOCC, whereas for RUNTYP=SPINORBT, it should equal the NFZC value in the \$CIDRTs. The default is the number of AOs given in \$DATA, again this is not very reasonable.

IROOTS = array containing the number of CI states for which the transition moments are to be found. The default is 1 for each CI, which is probably not a correct choice for TRANSITN runs, but is quite reasonable for SPINORBT.

NSTATE = array of CI states to be found when diagonalising the CI Hamiltonian. Of those, the first IROOTS(i) will be used to find transition moments. NSTATE in \$GUGDIA overrides NSTATE(i), if it is larger.

Reasonable values for TRANSITN runs where you want to know the moments for the first 4 states are IROOTS(1)=4  
NSTATE(1)=8 for the case of a common orbital set, or  
IROOTS(1)=1,4 NSTATE(1)=8,8 for two sets of orbitals.

\* \* \* the next seven pertain only to spin-orbit runs \* \* \*

METHOD = ZEFF regards the nuclear charge appearing in the one electron operator as a scaling parameter to compensate for omitting the two electron term altogether. (default)  
= BREIT includes the full 1e- and 2e- operator. Only singlet and triplet states can be

coupled. If you ask for more than one triplet state ( $i_{\text{roots}}(2) > 1$ ), then no Hso diagonalisation will be performed. This code is presently limited to a maximum of 10 orbitals (32 bit machines) or about 16 orbitals (64 bit machines) in the active space.

ZEFTYP specifies what Zeff charges to use  
= 3-21G selects a set obtained by 3-21G, but probably appropriate for any all electron basis set  
= SBK selects a set obtained for the SBKJC ECP basis set, specifically.

ZEFF = an array of effective nuclear charges overriding the charges chosen in ZEFTYP. This pertains only to METHOD=ZEFF SPINORBT runs.  
(default: true charges with any effective core potential protons removed if ECPs are used).

TMOMNT = flag to control spin-orbit transition moment calculation (default is .FALSE.) (only with ZEFF)

SAVDSK = flag to repeat the form factor calculation twice. This saves disk space if limited storage is available. (default=.FALSE.) (only with BREIT)

ACTION control of disk file DAFL30 reuse.  
= NORMAL calculate the form factors in this run.  
= SAVE calculate, and store the form factors on disk for future runs with the same active space characteristics.  
= READ read the form factors from disk from an earlier run which used SAVE.  
(default=NORMAL) (only with BREIT)

MS = limit spin projection of the triplet state  
0 do matrix elements for ms=0 only  
1 do matrix elements for ms=1 only  
-2 do matrix elements for all ms (0,1 and -1)  
-3 do not calculate any matrix element  
(ie useful for saving form factors)  
(default=-2) (only BREIT)

\* \* \* the remaining parameters are not so important \* \* \*

PRTCMO = flag to control printout of the corresponding orbitals. (default is .FALSE.)

TOLZ = MO coefficient zero tolerance (as for \$GUESS).  
(default=1.0E-8)

TOLE = MO coefficient equating tolerance (as for \$GUESS). (default=1.0E-5)

PRTPRM = flag to provide detailed information about the

composition of the spin-mixed states in terms of  
adiabatic states. (default is .FALSE.) (only ZEFF)

PRMTOL = threshold for Slater determinant output.  
(default is 0.0) (only ZEFF)

=====

• \* \* \* \* \*

•

For information on TRANSITN and SPINORBT,  
see the 'further information' section.

• \* \* \* \* \*