Chapter 6.1.7.1

SHELXL-97

Crystal Data & General Instructions

George W. Sheldrick
Dept. of Structural Chemistry
Tammannstraße 4
37077 Göttingen, Germany
email: gsheldr@shelx.uni-ac.gwdg.de

This chapter lists the instructions that may be used in the *.ins* file for SHELXL-97. Defaults are given in square brackets; '#' indicates that the program will generate a suitable default value based on the rest of the available information. Continuation lines are flagged by '=' at the end of a line, the instruction being continued on the next line which must start with one or more spaces. Other lines beginning with spaces are treated as comments, so blank lines may be added to improve readability. All characters following '!' or '=' in an instruction line are ignored.

The *.ins* file may include an instruction of the form: +filename (the '+' character MUST be in column 1). This causes further input to be taken from the named file until an END instruction is encountered in that file, whereupon the file is closed and instructions are taken from the next line of the *.ins* file. The input instructions from such an 'include' file are not echoed to the *.lst* and *.res* file, and may NOT contain FVAR, BASF, EXTI or SWAT instructions or atoms (except inside a FRAG...FEND section) since this would prevent the *.res* file from being used unchanged for the next refinement job (after renaming as *.ins*).

The '+filename' facility enables standard fragment coordinates or long lists of restraints etc. to be read from the same files for each refinement job, and for different structures to access the same fragment or restraint files. One could also for example store the LATT and SYMM instructions for different space groups, or neutron scattering factors for particular elements, or LAUE instructions followed by wavelength-dependent scattering factors, in suitably named files. Since these 'include' files are not echoed, it is a good idea to test them as part of an *.ins* file first, to check for possible syntax errors. Such 'include' files may be nested; the maximum allowed depth depends upon the operating system and compiler used.

6.1.7.1 Crystal data and general instructions

TITL [ ]
Title of up to 76 characters, to appear at suitable places in the output. The characters '!' and '=', if present, are part of the title and are not specially interpreted.

CELL λ a b c α β γ
Wavelength and unit-cell dimensions in Å and degrees.

ZERR Z esd(a) esd(b) esd(c) esd(α) esd(β) esd(γ)
Z value (number of formula units per cell) followed by the estimated standard deviations in the unit-cell dimensions. Z is only required for the CIF output; the cell esds contribute to the estimated esds in bond lengths etc. after full-matrix refinement.

LATT N[1]
Lattice type: 1=P, 2=I, 3=rhombohedral obverse on hexagonal axes, 4=F, 5=A, 6=B, 7=C. N must be made negative if the structure is non-centrosymmetric.

SYMM symmetry operation
Symmetry operators, i.e. coordinates of the general positions as given in International Tables. The operator x, y, z is always assumed, so MUST NOT be input. If the structure is centrosymmetric, the origin MUST lie on a center of symmetry. Lattice centering and the presence of an inversion center should be indicated by LATT, not SYMM. The symmetry operators may be specified using decimal or fractional numbers, e.g. 0.5-x, 0.5+y, -z or Y-X, -X, Z+1/6; the three components are separated by commas.

SFAC elements
Element symbols which define the order of scattering factors to be employed by the program. The first 94 elements of the periodic system are recognized. The element name may be preceded by '$' but this is not obligatory (the '$' character is allowed for logical consistency but is ignored). The program uses the neutral atom scattering factors, f', f" and absorption coefficients from International Tables for Crystallography, Volume C (1992), Ed. A.J.C. Wilson, Kluwer Academic Publishers, Dordrecht: Tables 6.1.1.4(pp. 500-502), 4.2.6.8 (pp. 219-222) and 4.2.4.2 (pp. 193-199) respectively. The covalent radii stored in the program are based on experience rather than taken from a specific source, and are deliberately overestimated for elements which tend to have variable coordination numbers so that 'bonds' are not missed, at the cost of generating the occasional 'non-bond'. The default radii (not those set for individual atoms by CONN) are printed before the connectivity table.

SFAC label a1 b1 a2 b2 a3 b3 a4 b4 c f' f" mu r wt
Scattering factor in the form of an exponential series, followed by real and imaginary dispersion terms, linear absorption coefficient, covalent radius and atomic weight. Except for the 'label' and atomic weight the format is the same as that used in SHELX-76. label consists of up to 4 characters beginning with a letter (e.g. Ca2+) and should be included before a1; for consistency the first label character may be a '$', but this is ignored (note however that the '$', if used, counts as one of the four characters, leaving only three for the rest of the label). The two SFAC formats may be used in the same *.ins* file; the order of the SFAC instructions (and the order of element names in the first type of SFAC instruction) define the scattering factor numbers which are referenced by atom instructions. The units of mu should be barns/atom, as in Table 4.2.4.2 of International Tables, Volume C (see above). For neutrons this format should be used, with a1...b4 set to zero.

Hydrogen atoms are treated specially by SHELXL; they are recognized by having the scattering factor number that corresponds to 'H' on the SFAC instruction. For X-ray

structures that contain both D and H, e.g. because the crystals were grown from a deuterated solvent in an n.m.r tube (a common source of good crystals!), both H and D should be included on the SFAC and UNIT instructions, but all the H and D atoms should employ the 'H' scattering factor number. In this way the density will be calculated correctly, but the D atoms may be idealized using HFIX etc.

DISP E f' f" [#] mu [#]
The DISP instruction allows the dispersion and (optionally) the absorption coefficient of a particular element (the name may be optionally prefaced by '$') to be read in without having to use the full form of the SFAC instruction. It will typically be used for synchrotron data where the wavelength does not correspond to the values (for Cu, Mo and Ag radiation) for which these terms are stored in the program. All other terms on the SFAC instruction are independent of the wavelength, so its short form may then be used. DISP instructions, if present, MUST come between the last SFAC and the UNIT instruction.

UNIT n1 n2 ...
Number of atoms of each type in the unit-cell, in SFAC order.

LAUE E
Wavelength-dependent values of f' and f" may be defined for an element E by means of the LAUE instruction, which is used in conjunction with the HKLF 2 reflection data format (in which the wavelength is given separately for each reflection). This is primarily intended for refinement of structures against Laue data collected using synchrotron radiation, but could also be used for refinement of a structure using data collected at different wavelengths for which some of the dispersion terms are significant (e.g. MAD data for macromolecules). There is no provision for handling overlapping reflection orders, and scaling for the source intensity distribution and Lp, absorption corrections etc. must have been performed before using SHELXL. A dummy wavelength of say 0.7 Å should be given on the CELL instruction, and the absorption coefficient estimated by the program should be ignored.

The element symbol may be preceded by '$' but this is optional; it must be followed by at least one blank or the end of the line. Any remaining information on the LAUE instruction line is ignored. The line immediately following the LAUE instruction is always ignored, and so may be used for headings. The following lines contain values of wavelength (in Å), f' and f" in FORMAT(F7.3,2F8.3); further information (e.g. the absorption coefficient μ) may follow on the same line but will be ignored. The wavelength values must be in ascending order and will be linearly interpolated; the wavelength intervals do not need to be equal (but it is more efficient if most of them are) and should indeed be smaller in the region of an absorption edge. This list is terminated by a record in which all three values are given as zero. There should only be one LAUE instruction for each element type; if

a reflection wavelength is outside the range specified, the constant f' and f" values defined by the corresponding SFAC instruction are used instead.

A LAUE instruction must be preceded by (normal) SFAC and UNIT instructions referencing the elements in question, and by all atoms. Thus the LAUE instruction(s) are usually the last instructions before HKLF 2 (or -2) at the end of the *.ins* file (which facilitates editing). The +filename construction may conveniently be used to read long LAUE tables from 'include' files without echoing them.

REM
Followed by a comment on the same line. This comment is copied to the results file (*.res*). A line beginning with at least one blank may also be used as a comment, but such comments are only copied to the *.res* file if the line is completely blank; REM comments are always copied. Comments may also be included on the same line as any instruction following the character '!', and are copied to the *.res* file (except in the case of atoms and FVAR, EXTI, SWAT and BASF instructions).

MORE m [1]
MORE sets the amount of (printer) output; m takes a value in the range 0 (least) to 3 (most verbose). MORE 0 also suppresses the echoing to the *.lst* file of any instructions or atoms which follow it (until the next MORE instruction).

TIME t [#]
If the time t (measured in seconds from the start of the job) is exceeded, SHELXL performs no further least-squares cycles, but goes on to the final structure factor calculation followed by bond lengths, Fourier calculations etc. The default value of t is installation dependent, and is either set to 'infinity' or to a little less than the maximum time allocation for a particular class of job. Usually t is 'CPU time', but some some operating systems (e.g. MSDOS) the elapsed time may have to be used instead.

END
END is used to terminate an 'include' file, and may also be included after HKLF in the *.ins* file (for compatibility with SHELX-76).