Table 1.4.4.1. *Correspondence between types of centring in direct and reciprocal lattices*

Direct lattice		Reciprocal lattice		
Lattice type(s)	Centring translations	Lattice type(s)	Restriction on *hkl*	Multiple unit cell
P, *R*		*P*, *R*		$\mathbf{a}^*$, $\mathbf{b}^*$, $\mathbf{c}^*$
A	$0, \frac{1}{2}, \frac{1}{2}$	*A*	$k+l=2n$	$\mathbf{a}^*$, $2\mathbf{b}^*$, $2\mathbf{c}^*$
B	$\frac{1}{2}, 0, \frac{1}{2}$	*B*	$h+l=2n$	$2\mathbf{a}^*$, $\mathbf{b}^*$, $2\mathbf{c}^*$
C	$\frac{1}{2}, \frac{1}{2}, 0$	*C*	$h+k=2n$	$2\mathbf{a}^*$, $2\mathbf{b}^*$, $\mathbf{c}^*$
I	$\frac{1}{2}, \frac{1}{2}, \frac{1}{2}$	*F*	$h+k+l=2n$	$2\mathbf{a}^*$, $2\mathbf{b}^*$, $2\mathbf{c}^*$
F	$0, \frac{1}{2}, \frac{1}{2}$ $\frac{1}{2}, 0, \frac{1}{2}$ $\frac{1}{2}, \frac{1}{2}, 0$	*I*	$k+l=2n$ $h+l=2n$ $h+k=2n$	$2\mathbf{a}^*$, $2\mathbf{b}^*$, $2\mathbf{c}^*$
$R_{\rm hex}$	$\frac{2}{3}, \frac{1}{3}, \frac{1}{3}$ $\frac{1}{3}, \frac{2}{3}, \frac{2}{3}$	$R_{\rm hex}$	$-h+k+l=3n$	$3\mathbf{a}^*$, $3\mathbf{b}^*$, $3\mathbf{c}^*$

the removal of lattice points with zero weights (absences). All the restrictions on *hkl* pertain to indexing on $\mathbf{a}^*$, $\mathbf{b}^*$ and $\mathbf{c}^*$.

(ii) The centring type of the reciprocal lattice refers to the multiple unit cell given in the table.

(iii) The centring type denoted by $R_{\rm hex}$ is a representation of the rhombohedral lattice *R* by a triple hexagonal unit cell, in the obverse setting (*IT* I, 1952), *i.e.* according to the transformation

$$\begin{aligned}\mathbf{a} &= \mathbf{a}_R - \mathbf{b}_R \\ \mathbf{b} &= \mathbf{b}_R - \mathbf{c}_R \\ \mathbf{c} &= \mathbf{a}_R + \mathbf{b}_R + \mathbf{c}_R,\end{aligned} \qquad (1.4.4.14)$$

where $\mathbf{a}_R$, $\mathbf{b}_R$ and $\mathbf{c}_R$ pertain to a primitive unit cell in the rhombohedral lattice *R*.

The corresponding multiple reciprocal cell, with centring denoted by $R_{\rm hex}$, contains nine lattice points with coordinates 000, 021, 012, 101, 202, 110, 220, 211 and 122 – indexed on the usual reciprocal to the triple hexagonal unit cell defined by (1.4.4.14). Detailed derivations of these correspondences are given by Buerger (1942), and an elementary proof of the reciprocity of *I* and *F* lattices can be found, *e.g.*, in pamphlet No. 4 of the Commission on Crystallographic Teaching (Authier, 1981). Intuitive proofs follow directly from the restrictions on *hkl*, given in Table 1.4.4.1.

Appendix 1.4.1.

Comments on the preparation and usage of the tables

(U. SHMUELI)

The straightforward but rather extensive calculations and text processing related to Tables A1.4.3.1 through A1.4.3.7 and Table A1.4.4.1 in Appendices 1.4.3 and 1.4.4, respectively, were performed with the aid of a combination of FORTRAN and REDUCE (Hearn, 1973) programs, designed so as to enable the author to produce the table entries directly from a space-group symbol and with a minimum amount of intermediate manual intervention. The first stage of the calculation, the generation of a space group (coordinates of the equivalent positions), was accomplished with the program *SPGRGEN*, the algorithm of which was described in some detail elsewhere (Shmueli, 1984). A complete list of computer-adapted space-group symbols, processed by *SPGRGEN* and not given in the latter reference, is presented in Table A1.4.2.1 of Appendix 1.4.2.

The generation of the space group is followed by a construction of symbolic expressions for the scalar products $\mathbf{h}^T(\boldsymbol{P}\mathbf{r}+\mathbf{t})$; *e.g.* for position No. (13) in the space group $P4_132$ (No. 213, *IT* I, 1952, *IT* A, 1983), this scalar product is given by $h(\frac{3}{4}+y)+k(\frac{1}{4}+x)+l(\frac{1}{4}-z)$. The construction of the various table entries consists of expanding the sines and cosines of these scalar products, performing the required summations, and simplifying the result where possible. The construction of the scalar products in a FORTRAN program is fairly easy and the extremely tedious trigonometric calculations required by equations (1.4.2.19) and (1.4.2.20) can be readily performed with the aid of one of several available computer-algebraic languages (for a review, see *Computers in the New Laboratory – a Nature Survey*, 1981); the REDUCE language was employed for the above purpose.

Since the REDUCE programs required for the summations in (1.4.2.19) and (1.4.2.20) for the various space groups were seen to have much in common, it was decided to construct a FORTRAN interface which would process the space-group input and prepare automatically REDUCE programs for the algebraic work. The least straightforward problem encountered during this work was the need to 'convince' the interface to generate *hkl* parity assignments which are appropriate to the space-group information input. This was solved for all the crystal families except the hexagonal by setting up a 'basis' of the form: $h/2, k/2, l/2, (k+l)/2, \ldots, (h+k+l)/4$ and representing the translation parts of the scalar products, $\mathbf{h}^T\mathbf{t}$, as sums of such 'basis functions'. A subsequent construction of an automatic parity routine proved to be easy and the interface could thus produce any number of REDUCE programs for the summations in (1.4.2.19) and (1.4.2.20) using a list of space-group symbols as the sole input. These included trigonal and hexagonal space groups with translation components of $\frac{1}{2}$. This approach seemed to be too awkward for some space groups containing threefold and sixfold screw axes, and these were treated individually.

There is little to say about the REDUCE programs, except that the output they generate is at the same level of trigonometric complexity as the expressions for *A* and *B* appearing in Volume I (*IT* I, 1952). This could have been improved by making use of the pattern-matching capabilities that are incorporated in REDUCE, but

it was found more convenient to construct a FORTRAN interpreter which would detect in the REDUCE output the basic building blocks of the trigonometric structure factors (see Section 1.4.3.3) and perform the required transformations.

Tables A1.4.3.1–A1.4.3.7 were thus constructed with the aid of a chain composed of (i) a space-group generating routine, (ii) a FORTRAN interface, which processes the space-group input and 'writes' a complete REDUCE program, (iii) execution of the REDUCE program and (iv) a FORTRAN interpreter of the REDUCE output in terms of the abbreviated symbols to be used in the tables. The computation was at a 'one-group-at-a-time' basis and the automation of its repetition was performed by means of procedural constructs at the operating-system level. The construction of Table A1.4.4.1 involved only the preliminary stage of the processing of the space-group information by the FORTRAN interface. All the computations were carried out on a Cyber 170-855 at the Tel Aviv University Computation Center.

It is of some importance to comment on the recommended usage of the tables included in this chapter in automatic computations. If, for example, we wish to compute the expression: $A = -8(\mathrm{Escs} + \mathrm{Ossc})$, use can be made of the facility provided by most versions of FORTRAN of transferring subprogram names as parameters of a FUNCTION. We thus need only two FUNCTIONs for any calculation of A and B for a cubic space group, one FUNCTION for the block of even permutations of x, y and z:

```
      FUNCTION E(P,Q,R)
      EXTERNAL SIN,COS
      COMMON/TSF/TPH,TPK,TPL,X,Y,Z
      E=P(TPH*X)*Q(TPK*Y)*R(TPL*Z)
     1+P(TPH*Z)*Q(TPK*X)*R(TPL*Y)
     2+P(TPH*Y)*Q(TPK*Z)*R(TPL*X)
      RETURN
      END
```

where TPH, TPK and TPL denote $2\pi h$, $2\pi k$ and $2\pi l$, respectively, and a similar FUNCTION, say O(P,Q,R), for the block of odd permutations of x, y and z. The calling statement in the calling (sub)program can thus be:

```
A=-8*(E(SIN,COS,SIN)+O(SIN,SIN,COS)).
```

A small number of such FUNCTIONs suffices for all the space-group-specific computations that involve trigonometric structure factors.

Appendix 1.4.2.

Space-group symbols for numeric and symbolic computations

A1.4.2.1. *Introduction* (U. Shmueli, S. R. Hall and R. W. Grosse-Kunstleve)

This appendix lists two sets of computer-adapted space-group symbols which are implemented in existing crystallographic software and can be employed in the automated generation of space-group representations. The computer generation of space-group symmetry information is of well known importance in many crystallographic calculations, numeric as well as symbolic. A prerequisite for a computer program that generates this information is a set of computer-adapted space-group symbols which are based on the generating elements of the space group to be derived. The sets of symbols to be presented are:

(i) *Explicit symbols*. These symbols are based on the classification of crystallographic point groups and space groups by Zachariasen (1945). These symbols are termed *explicit* because they contain in an explicit manner the rotation and translation parts of the space-group generators of the space group to be derived and used. These computer-adapted explicit symbols were proposed by Shmueli (1984), who also describes in detail their implementation in the program *SPGRGEN*. This program was used for the automatic preparation of the structure-factor tables for the 17 plane groups and 230 space groups, listed in Appendix 1.4.3, and the 230 space groups in reciprocal space, listed in Appendix 1.4.4. The explicit symbols presented in this appendix are adapted to the 306 representations of the 230 space groups as presented in *IT* A (1983) with regard to the standard settings and choice of space-group origins.

The symmetry-generating algorithm underlying the explicit symbols, and their definition, are given in Section A1.4.2.2 of this appendix and the explicit symbols are listed in Table A1.4.2.1.

(ii) *Hall symbols*. These symbols are based on the implied-origin notation of Hall (1981*a*,*b*), who also describes in detail the algorithm implemented in the program *SGNAME* (Hall, 1981*a*). In the first edition of *IT* B (1993), the term 'concise space-group symbols' was used for this notation. In recent years, however, the term 'Hall symbols' has come into use in symmetry papers (Altermatt & Brown, 1987; Grosse-Kunstleve, 1999), software applications (Hovmöller, 1992; Grosse-Kunstleve, 1995; Larine *et al.*, 1995; Dowty, 1997) and data-handling approaches (Bourne *et al.*, 1998). This term has therefore been adopted for the second edition.

The main difference in the definition of the Hall symbols between this edition and the first edition of *IT* B is the generalization of the origin-shift vector to a full change-of-basis matrix. The examples have been expanded to show how this matrix is applied. The notation has also been made more consistent, and a typographical error in a default axis direction has been corrected.* The lattice centring symbol 'H' has been added to Table A1.4.2.2. In addition, Hall symbols are now provided for 530 settings to include all settings from Table 4.3.1 of *IT* A (1983). Namely, all non-standard symbols for the monoclinic and orthorhombic space groups are included.

Some of the space-group symbols listed in Table A1.4.2.7 differ from those listed in Table B.6 (p. 119) of the first edition of *IT* B. This is because the symmetry of many space groups can be represented by more than one subset of 'generator' elements and these lead to different Hall symbols. The symbols listed in this edition have been selected after first sorting the symmetry elements into a strictly prescribed order based on the shape of their Seitz matrices, whereas those in Table B.6 were selected from symmetry elements in the order of *IT* I (1965). Software for selecting the Hall symbols listed in Table A1.4.2.7 is freely available (Hall, 1997). These symbols and their equivalents in the first edition of *IT* B will generate identical symmetry elements, but the former may be used as a reference table in a strict mapping procedure between different symmetry representations (Hall *et al.*, 2000).

The Hall symbols are defined in Section A1.4.2.3 of this appendix and are listed in Table A1.4.2.7.

* The correct default axis direction $\mathbf{a} - \mathbf{b}$ of an N preceded by 3 or 6 replaces $\mathbf{a} + \mathbf{b}$ on p. 117, right-hand column, line 4, in the first edition of *IT* B.

references