JOURNAL OF COMPUTATIONAL PHYSICS 16, 167–185 (1974)

Computerized Group Theory for Lattice Dynamical Problems*

L. L. BOYER†

Solid State and Materials Laboratory, Princeton University, Princeton, New Jersey 08540

Received April 10, 1974

The calculation of the normal modes of vibration of a crystal in general requires the diagonalization of the dynamical matrix for wave vector **q**, which has rank $3n$, where n is the number of atoms per unit cell. The symmetry of the crystal can be used to block diagonalize the dynamical matrix using group representation theory. Here we describe a subroutine, named BLOCDI that computes the unitary matrix which group theoretically block diagonalizes the dynamical matrix for any crystal structure and wave vector.

The input to the subroutine is first put into a form that is generally acceptable for all crystal structures. This is made possible by specifying the structure in terms of integer variables which relate to the primitive axes. The projection operator is used to generate and print out possible symmetry coordinates, labeled according to their irreducible representation and site symmetry. The set of coordinates to be used in the construction of the transformation matrix is then easily selected by inspection from the printed sets. In a second run the selected coordinates are computed and the nonzero values of the matrix and their indices are stored in singly dimensioned arrays.

Possible reductions due to time reversal symmetry are not considered. There is essentially no restriction on the size of the unit cell. The actual transformation of the dynamical matrix (using the output of BLOCDI) becomes a time and storage problem long before the computation and indexing of the nonzero elements of the transformation matrix becomes a problem.

I. INTRODUCTION

The symmetry of a crystal greatly simplifies the computation of its normal vibrational modes. In the harmonic approximation lattice translations and the assumption of periodic boundary conditions reduce the task from that of diagonalizing a matrix of rank $3Nn$ (where N is the number of primitive, or Bravais, unit cells in the chosen volume of periodicity and n is the number of atoms in the primitive cell) to that of diagonalizing N matrices of rank $3n$ [1]. Each of these N matrices corresponds to a wave vector in the "allowed" set of N-wave vectors in a single Brillouin zone of the reciprocal lattice. This is an enormous

* Work Supported by Grant No. DE 02492 of the National Institutes of Health.

† Present Address: Code 6682, Naval Research Laboratory, Washington, D.C. 20375.

simplification of the general problem, especially since it is formally accomplished for any crystal structure.

For certain wave vectors $\mathbf{q}$, additional simplifications are often possible owing to symmetry operations other than lattice translations. These additional symmetry operations form a group $F_{\mathbf{q}}$, known as the group of the wave vector [2]. $F_{\mathbf{q}}$ depends not only on the $\mathbf{q}$ but on the space group of the crystal as well. Furthermore, the specific unitary matrix which block diagonalizes the dynamical matrix depends not only on $F_{\mathbf{q}}$ but also on the sites of the various atoms in the primitive cell. The point is that, whereas the block diagonalization arising from lattice translations has general applicability, any additional reductions are only obtained by considering a particular wave vector and a particular crystal structure.

The general approach to this problem (group representation theory [3]) is well understood but its application (if done by hand) is very tedious and time consuming if n is large, and often, human errors result. Also, because of the large number of specific cases that must be accounted for, the problem is one which appears difficult to program for the computer in a general way.

Recently, however, Worlton and Warren [4] have presented a program which is reported to be rather generally applicable to this problem, although they do restrict the size of the unit cell to $n \leqslant 20$. Also, their program is quite large (1600 cards) since in endeavoring to reduce the input to a minimum, they construct the space group from atomic positions. Also, their program constructs a dynamical matrix symmetry, reduces it, and carries out the block diagonalization.

We have found that for large N, a more satisfactory approach is possible whereby one first puts the information defining the crystal structure and wave vector into a form which is generally applicable for all crystal structures and wave vectors. As we shall see, this is not a difficult task, and as a result, the program required to carry out the computations is greatly simplified.

In the sections that follow we present and discuss a computer subroutine named BLOCDI, consisting of 135 cards in FORTRAN, which is designed to compute the unitary matrix which group theoretically block diagonalizes the dynamical matrix for any wave vector and any crystal structure. In Section 2 we review the general theory that goes into the computations. In Section 3 we describe the input and output for BLOCDI in general, and illustrate its use by considering a specific example having space group $I4_1/a$ and 18 atoms per primitive cell. In Section 4 we discuss how the subroutine works by relating specific computations in the routine to the general theory. A listing of BLOCDI is given in the appendix.

II. Summary of the General Theory

For a general review of the subject matter summarized in this section see Boyer [5]; [3, 6–8] may also be helpful in this regard.

Let $\omega^2(j, \mathbf{q})$ be the jth eigenvalue and $e_\alpha(kj, \mathbf{q})$ the (αk) component of the eigenvector of the dynamical matrix, $D_{\alpha\beta}(kk', \mathbf{q})$, for wave vector $\mathbf{q}$. That is,

$$\sum_{k'\beta} D_{\alpha\beta}(kk', \mathbf{q})\, e_\beta(k'j, \mathbf{q}) = \omega^2(j, \mathbf{q})\, e_\alpha(kj, \mathbf{q}). \tag{1}$$

The indices, α and β, refer to the axes of a Cartesian coordinate system while k and k' take on values from 1 to n and label the various atoms within the primitive (or Bravais) unit cell.

Let F be the factor group of the space group with respect to the lattice translations. The elements of F are actually sets of symmetry operations where each set is made up of those operations of the space group which differ only by lattice translations. Thus, the ρth element of F is given by a rotation and (or) reflection part, $\theta^\rho_{\alpha\beta}$, plus a translation, $\mathbf{t}^\rho$, where the translation is not a nonzero lattice vector. Let $\mathbf{x}(l, k)$ be the position of the k atom in the lth cell with respect to a chosen cartesian coordinate system. A new coordinate system ρ, may be obtained from the chosen one by applying the ρth operation in F. With respect to the ρth frame the position of the (l, k) atom is given by

$$x_\alpha{}^\rho(l, k) = \sum_\beta \theta^\rho_{\alpha\beta} x_\beta(l, k) + t_\alpha{}^\rho. \tag{2}$$

Since ρ is a symmetry operation there exists an atom, (l_ρ, k_ρ) such that

$$\mathbf{x}^\rho(l, k) = \mathbf{x}(l_\rho, k_\rho) \tag{3}$$

and k_ρ is the same kind of atom as k.

It can be shown that under the operations ρ, the eigenvectors of the dynamical matrix for wave vector $\mathbf{q}$, satisfy the relation

$$e_\alpha(kj, \mathbf{q}^\rho) = \sum_\beta \theta^\rho_{\beta\alpha} e_\beta(k_\rho j, \mathbf{q}), \tag{4}$$

where $\mathbf{q}^\rho$ is defined by

$$q_\alpha{}^\rho = \sum_\beta \theta^\rho_{\beta\alpha} q_\beta\,. \tag{5}$$

Equation (4) defines a $3n \times 3n$ representation of $F_\mathbf{q}$ where $F_\mathbf{q}$ is that subgroup of F for which $\mathbf{q}^\rho$ and $\mathbf{q}$ differ only by a reciprocal lattice vector.

It is convenient to define a single index i, from the two indices α and k by

$$i = (\alpha - 1)n + k, \tag{6}$$

so that the elements of the dynamical matrix and the representation matrices are written $D_{ij}(\mathbf{q})$ and $R_{ij}(\rho)$. One constructs the unitary matrix $U_{ij}(\mathbf{q})$, which block

diagonalizes the dynamical matrix, from symmetry coordinates obtained from the projection operator

$$P^{\nu}_{ij}(\mathbf{q}) \propto \sum_{\rho} \chi_{\nu}^{*}(\rho, \mathbf{q})\, R_{ij}(\rho), \tag{7}$$

where $\chi_{\nu}(\rho, \mathbf{q})$ is the character of the ρth operation in the νth irreducible representation of $F_{\mathbf{q}}$ and the summation is over those ρ in $F_{\mathbf{q}}$. The number of linearly independent coordinates, obtained by applying $\mathbf{P}^{\nu}(\mathbf{q})$ to arbitrary $3n$-dimensional column vectors, is the dimension of the νth irreducible representation $d_{\nu}(\mathbf{q})$, times its multiplicity,

$$m_{\nu}(\mathbf{q}) = \frac{1}{g_{\mathbf{q}}} \sum_{\rho} \chi_{\nu}^{*}(\rho, \mathbf{q})\, \chi(\rho), \tag{8}$$

where $\chi(\rho)$ is the character of $\mathbf{R}(\rho)$ and $g_{\mathbf{q}}$ is the number of elements in $F_{\mathbf{q}}$. When orthonormalized and arranged in groups with the same ν, these coordinates form the matrix $\mathbf{U}(\mathbf{q})$. The block diagonalized dynamical matrix,

$$\mathbf{B}(\mathbf{q}) = \mathbf{U}^{\dagger}(\mathbf{q})\, \mathbf{D}(\mathbf{q})\, \mathbf{U}(\mathbf{q}), \tag{9}$$

consists of $p(\mathbf{q})$ matrices of rank $d_{\nu}(\mathbf{q})\, m_{\nu}(\mathbf{q})$ where $p(\mathbf{q})$ is the number of irreducible representations of $F_{\mathbf{q}}$.

III. Description of Subroutine BLOCDI

A subroutine named BLOCDI is listed in the appendix which consists of approximately 130 FORTRAN statements. Its function is to determine $m_{\nu}(\mathbf{q})$ and $\mathbf{U}(\mathbf{q})$ as described in the previous section given the crystal structure, the wave vector, and the characters of the irreducible representations of $F_{\mathbf{q}}$. We first consider the zero wave-vector problem and illustrate the use of BLOCDI by treating a specific crystal structure with space group symmetry $I4_1/a - C^6_{4h}$.

The computation of $\mathbf{U}$ requires two steps. In the first, a run is made with the logical variable CALCU assigned the value .FALSE.. When CALCU is so assigned, the multiplicities are computed and sets of allowed symmetry coordinates, obtained by applying the projection operator to appropriate subspaces of the representation given by Eq. (4), are printed. The symmetry coordinates to be used in the construction of $\mathbf{U}$ are then selected (by hand) from the printed sets and indexed by assigning values to the arrays MULT and IVEC. In practice this is not difficult as we shall demonstrate for our example. In the second run we set CALCU = .TRUE. and $\mathbf{U}$ is constructed from the selected symmetry coordinates. All input variables and all output variables are listed in a COMMON block named MB. We first define all input necessary for the use of BLOCDI when CALCU = .FALSE..

The number of irreducible representations of $F_0 = F$ is given by NIREP and the number of elements (symmetry operations) in F_0 by NING. CHAR(I, J) is the character of the Jth element of F_0 in the Ith irreducible representation. For our example, we need the point group $4/m - C_{4h}$ which has NING = 8 symmetry operations and whose character table is that listed in Table 1. The order in which the indices I and J are assigned to the irreducible representations and group elements is arbitrary. The two E_g and the two E_u representations are each considered separately and thus we have NIREP = 8.

TABLE I

Character Table for the Point Group $4/m - C_{4h}$ and the Corresponding Values For CHAR(I, J).

I \ J	1	2	3	4	5	6	7	8	
1	1	1	1	1	1	1	1	1	A_g
2	1	1	−1	−1	1	1	−1	−1	B_g
3	1	−1	i	$-i$	1	−1	i	$-i$	E_g
4	1	−1	$-i$	i	1	−1	$-i$	i	
5	1	1	1	1	−1	−1	−1	−1	Au
6	1	1	−1	−1	−1	−1	1	1	Bu
7	1	−1	i	$-i$	−1	1	$-i$	i	E_u
8	1	−1	$-i$	i	−1	1	i	$-i$	
	E	C_2	C_4	C_4^3	i	iC_2	iC_4	iC_4^3	

For our example we assume a crystal structure having two A atoms occupying b sites (in Wyckoff notation), four B atoms and four C atoms occupying e sites, and eight D atoms occupying f sites. For convenience in the subsequent discussion we say that for $\mathbf{q} = 0$, two atoms have the same *site symmetry* if they have the same Wyckoff symbol. Later on in the discussion we will use the term "site symmetry" in connection with F_q for $\mathbf{q} \neq 0$.

To determine the crystal structure input to BLOCDI for $\mathbf{q} = 0$, we begin by constructing a table corresponding to the appropriate space group which is very similar to that given in the International Tables for X-ray Crystallography [9]. This new table is obtained by modifying the corresponding table in [9] in the following ways.

1. The coordinates of all equivalent positions must be taken with respect to the primitive axes of the lattice. For many space groups the coordinates in [9] are already with respect to the primitive axes. In our example, however, they are listed with respect to the tetragonal axes. The

primitive lattice vectors in this body centered system are $\mathbf{a}_1 = (a, a, c)$, $\mathbf{a}_2 = (-a, -a, c)$, and $\mathbf{a}_3 = (a, -a, -c)$ where a and c are the two lattice constants.

2. The coordinates of the equivalent positions are scaled by some appropriate integer value L, so that the coordinates of a general position $(y_1/L)\,\mathbf{a}_1 + (y_2/L)\,\mathbf{a}_2 + (y_3/L)\,\mathbf{a}_3$ are (y_1, y_2, y_3). The value of L is chosen so that the coordinates of the equivalent positions for all site symmetries can be represented by integer values.
3. The order of the equivalent positions for the most general site symmetry are listed to correspond to the order in which the characters (CHAR(I, J), $J = 1, 2, \ldots,$ NING) are listed.

TABLE II

The Number, Site Symmetry and Coordinates of Equivalent Positions in a Primitive Unit Cell with Space Group $I4_1/a - C_{4h}^6$. Coordinates are taken with Respect to the Primitive Axes and in Units where the Length of the Primitive Lattice Vectors (a, a, c), $(-a, -a, c)$, + $(a, -a, -c)$ is 8. Also listed are the Symbols for the Nontranslational Parts of the Elements in F_0 as they Correspond to the Coordinates of Equivalent Positions for the Most General Site Symmetry.

	Site Symmetry			
Number of Positions	Wyckoff Notation	Index for Occupied Sites I	Coordinates of Equivalent Positions	Symbol for nontranslational part of the elements in F_0
8	F	Occupied by *D atoms* $I = 3$	(y_1, y_2, y_2) $(y_2 - y_3, y_1 - y_3, -y_3)$ $(2 + y_2, 6 + y_2 - y_3, 4 - y_1 + y_2)$ $(2 + y_1 - y_3, 6 + y_1, 4 + y_1 - y_2)$ $(2 - y_1, 6 - y_2, 4 - y_3)$ $(2 - y_2 + y_3, 6 - y_1 + y_3, 4 + y_3)$ $(-y_2, -y_2 + y_3, y_1 - y_2)$ $(-y_1 + y_3, -y_1, -y_1 + y_2)$	E C_2 C_4 C_4^3 i iC_2 iC_4 iC_4^3
4	e	Occupied by B and C atoms $I = 2$	$(x, x, 0); (2 + x, 6 + x, 4)$ $(-x, -x, 0); (2 - x, 6 - x, 4)$	
4	d	Not occupied	(5, 3, 6); (5, 7, 2); (5, 3, 2); (7, 3, 6)	
4	c	Not occupied	(1, 7, 6); (1, 3, 2); (1, 7, 2); (5, 7, 6)	
2	b	Occupied by A atoms; $I = 1$	(4, 4, 0); (6, 2, 4)	
2	a	Not occupied	(0.0.0); (2, 6, 4)	

The crystal structure input can now be taken directly from Table 2. NOFSG is the number of occupied site symmetries; for our example NOFSG = 3. NINSG(I) is the number of equivalent positions with site symmetry I. The order in which I is associated with the occupied site symmetries is not important. However, once the order is chosen the index k, of the dynamical matrix must be defined in a consistent manner (see Eq. (10)). For our example we let $I = 1$ correspond to b sites, $I = 2$ to e sites and $I = 3$ to f sites. Thus, we have NINSG(1) = 2, NINSG(2) = 4, and NINSG(3) = 8. NIONSG(I) is the number of nonequivalent atoms with site symmetry I. For our example we have NIONSG(1) = 1, NIONSG(2) = 2, and NIONSG(3) = 1. LATCON is the value of L; in our case LATCON = 8.

The arrays ITRAN AND IFE, which, respectively, denote the translational and nontranslational components of the elements of F_0, are taken directly from the coordinates of equivalent positions for the most general site symmetry regardless of whether it is occupied or not. Specifically, ITRAN(I, J) is the Jth component of the translation associated with the Ith equivalent position while IFE(I, J, K) is the coefficient of y_K in the Jth component of the Ith position. Here, $I = 1, 2,...,$ NING refers to the order in which these equivalent positions are written, which, as was mentioned above, is the same order used in writing the character table. The values of ITRAN and IFE for our example are shown in Table 3.

The array LX(I, J, K) denotes the Kth component of a pertinent set of $J = 1, 2,...,$ NINSG(I), equivalent positions with site symmetry I with the requirement that $0 \leqslant \mathrm{LX}(I, J, K) < L$. For our example we determine LX by choosing $y_1 = 1$, $y_2 = 2$, $y_3 = 3$, and $x = 1$ (see Table 2) with the resultant array given

TABLE III

Values for the Arrays IFE and ITRAN Used As Input To Subroutine BLOCDI which Define the Space Group Operations of $I4_1/a - C_{4h}^6$

I	IFE(I, J, K) (J, K)									ITRAN(I, J) J		
	(1, 1)	(1, 2)	(1, 3)	(2, 1)	(2, 2)	(2, 3)	(3, 1)	(3, 2)	(3, 3)	1	2	3
1	1	0	0	0	1	0	0	0	1	0	0	0
2	0	1	−1	1	0	−1	0	0	−1	0	0	0
3	0	1	0	0	1	−1	−1	1	0	2	6	4
4	1	0	−1	1	0	0	1	−1	0	2	6	4
5	−1	0	0	0	−1	0	0	0	−1	2	6	4
6	0	−1	1	−1	0	1	0	0	1	2	6	4
7	0	−1	0	0	−1	1	1	−1	0	0	0	0
8	−1	0	1	−1	0	0	−1	1	0	0	0	0

TABLE IV

Values of the Array LX(I, J, K) Representing the Equivalent Positions For Site Symmetries $I = 1, 2$, and 3, or, in Wyckoff notation, respectively, sites b, e, and f, for space group $I4_1a - C_{4h}^6$. The Values were obtained from Table 2 Using $y_1 = 1$, $y_2 = 2$, $y_3 = 3$ and $x = 1$.

	LX(3, J, K)			LX(2, J, K)			LX(1, J, K)		
J \ K	1	2	3	1	2	3	1	2	3
1	1	2	3	1	1	0	4	4	0
2	7	6	5	7	7	0	6	2	4
3	4	5	5	3	7	4			
4	0	7	3	1	5	4			
5	1	4	1						
6	3	0	7						
7	6	1	7						
8	2	7	1						

in Table 4. Other values for y_1, y_2, y_3 and x would work equally well. For example, we could have chosen $x = 2$. However, $x = 4$ is not allowed since the associated equivalent positions are not e sites, but rather, b sites. The order in which the equivalent positions are assigned to values for I and J in LX(I, J, K) is arbitrary, but once an assignment is made, the labels of the atoms in the primitive unit cell ($k = 1, 2,..., n$) in the definition of the dynamical matrix must be assigned as follows. In general, we require

$$k = N_I + (N - 1) \cdot \text{NINSG}(I) + J, \tag{10}$$

where N_I is the total number of atoms with site symmetries $1, 2,..., I - 1$; N is the Nth nonequivalent atom with site symmetry I ($N = 1, 2,...,$ NIONSG(I)), and J is the second index in LX. For our example, suppose the $B(C)$ atoms are located at positions given by $x = x_B (x = x_C)$ and the D atoms at positions given by $y_1 = y_1{}^D$, $y_2 = y_2{}^D$ and $y_3 = y_3{}^D$. $k = 1$ and $k = 2$ refer, respectively, to the A atoms at (4, 4, 0) and (6, 4, 2). $k = 3$, 4, 5 and 6 refer, respectively, to the B atoms at $(x_B, x_B, 0)$, $(-x_B, -x_B, 0)$, $(2 + x_B, 6 + x_B, 4)$, and $(2 - x_B, 6 - x_B, 4)$ while $k = 7$, 8, 9 and 10 refer, respectively, to the C atoms at $(x_C, x_C, 0)$, $(-x_C, -x_C, 0)$, $(2 + x_C, 6 + x_C, 4)$, and $(2 - x_C, 6 - x_C, 4)$. Alternatively the k assignment for the B and C atoms could be that obtained by interchanging letter B and letter C in the previous statement. $k = 11, 12,..., 18$ refers to the sequence of D atoms located at positions given by the order in which they are listed in Table 2, since this was the order chosen for the representative positions LX(3, J, K), $J = 1, 2,..., 8$.

As we remarked earlier, the α and β indices in Eqs. (1), (2), (4) and (5) refer to components along some chosen cartesian coordinate system. Let A be the matrix given by

$$A_{\alpha\beta} = a_\alpha{}^\beta,$$

where $a_\alpha{}^\beta$ is the βth component (with respect to the Cartesian system) of the primitive lattice vector $\mathbf{a}_\alpha$. In BLOCDI, T(I, J) is the (I, J) component of **A** while TI(I, J) is the corresponding element of $\mathbf{A}^{-1}$. For our example, we choose the usual tetragonal axes for our cartesian system, which gives

$$A = \tfrac{1}{2}\begin{pmatrix} a & -a & a \\ a & -a & -a \\ c & c & -c \end{pmatrix}.$$

For convenience we may take $a = c = 1$ with the resulting values of T and TI given in Table 5.

TABLE V

Values for the Arrays T and TI of BLOCDI Relating the Transformation, and its Inverse, from the Tetragonal Axes to the Primitive Axes of a Body Centered Tetragonal System

	T(I, J)			TI(I, J)		
I \ J	1	2	3	1	2	3
1	0.5	−.5	0.5	1.0	0.0	1.0
2	0.5	−.5	−.5	0.0	−1.0	1.0
3	0.5	0.5	−.5	1.0	−1.0	0.0

Finally, INV(I) where $I = 1, 2, ...,$ NING is the index for the inverse of the Ith element of F_0. In our example INV(I) = 1, 2, 4, 3, 5, 6, 8, 7, respectively, for $I = 1, 2, ..., 8$.

We have now defined all the input variables necessary for the use of BLOCDI with CALCU = .FALSE. ; namely, NING, NIREP, CHAR, NOFSG, NINSG, NIONSG, LATCON, ITRAN, IFE, LX, T, TI and INV. When the program is run with CALCU = .FALSE. the variables MULT(I, J) and MNU(I) are computed and symmetry coordinates obtained using the projection operator, Eq. (7), are printed. MULT(I, J) is the multiplicity of the Jth irreducible representation arising from the Ith site symmetry. MNU(I) is the total multiplicity of the Ith irreducible representation which is simply the product of MULT(J, I) and NIONSG(J) summed over all site symmetries J. The values of MULT and MNU for our example are given in Table 6.

TABLE VI

Multiplicities for the Three Site Symmetries MULT, and the Total Multiplicities MNU, for our Example

J	1	2	3	4	5	6	7	8
MULT(1, J)	0	1	1	1	1	0	1	1
MULT(2, J)	1	1	2	2	1	1	2	2
MULT(3, J)	3	3	3	3	3	3	3	3
MNU(J)	5	6	8	8	6	5	8	8

Let $\delta_j(i)$ be the jth component of a $3n$-dimensional column vector with a zero for each component except $j = i$ which has the value 1. Let $x_j^{\nu}(i)$ be the jth component of the vector obtained by applying $\mathbf{P}^{\nu}$ (Eq. (7)) to $\delta(i)$. Necessarily, the nonzero components of $\mathbf{x}^{\nu}(i)$ correspond to atoms which are the same or equivalent to the atom given by i (see Eq. (6)). For our example $n = 18$ and the B atoms are $k = 3$, 4, 5 and 6. Thus, for $i = 3, 4, 5, 6, 21, 22, 23, 24, 39, 40, 41$ and 42, $x_j^{\nu}(i) = 0$ for $j = 1, 2, 7, 8,..., 19, 20, 25, 26,..., 37, 38, 43, 44,..., 53$ and 54. Furthermore, since the B and C atoms have the same site symmetry and since the sequence of k values given to the B atoms is the same as that given to the C atoms,

$$x_{j+4}^{\nu}(i + 4) = x_j^{\nu}(i), \tag{11}$$

where i and j take on values corresponding to the B atoms. The information printed by BLOCDI is a convenient listing of those portions of the column vectors $\mathbf{x}^{\nu}(i)$, for all irreducible representations ν, and site symmetries i, that are not necessarily zero.

In Table 7 we show that portion of the printout corresponding to the third irreducible representation and the first and second site symmetries. For site symmetry No. 1 there are 12 columns with six numbers in a column. The first two columns are the real and imaginary parts of the components of $\mathbf{x}^3(1)$ which are not necessarily zero. The second two columns similarly correspond to $\mathbf{x}^3(2)$, the third two columns to $\mathbf{x}^3(19)$, etc. Reading down a column, the first through sixth numbers correspond, respectively, to the components $j = 1, 2, 19, 20, 37$ and 38. The 12 double columns under the heading site symmetry No. 2 have similar meanings. For example, the sixth number in the eighth column (−2.00) is the imaginary part of $x_{22}^3(6)$ which is also the imaginary part of $x_{26}^3(10)$ by Eq. (11).

The number of linearly independent and nonzero vectors in the (I, J) set, where I and J refer to the site symmetry and irreducible representation indices, is the dimension of the Ith irreducible representation times MULT(I, J). One must choose this number of vectors from the (I, J) set which are mutually orthogonal

TABLE VII
That Portion of the Printout Corresponding to the Third Irreducible Representation and the First and Second Site Symmetries

IRREP NO. 3 SITE SYMMETRY NO. 1

2.00	0.00	−2.00	0.00	0.00	−2.00	0.00	2.00	0.00	0.00	0.00	0.00
−2.00	0.00	2.00	0.00	0.00	2.00	0.00	−2.00	0.00	0.00	0.00	0.00
0.00	2.00	0.00	−2.00	2.00	0.00	−2.00	0.00	0.00	0.00	0.00	0.00
0.00	−2.00	0.00	2.00	−2.00	0.00	2.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

IRREP NO. 3 SITE SYMMETRY NO. 2

2.00	0.00	0.00	0.00	0.00	0.00	−2.00	0.00	0.00	0.00	0.00	−2.00	0.00	2.00	0.00	0.00
0.00	0.00	2.00	0.00	−2.00	0.00	0.00	0.00	0.00	−2.00	0.00	0.00	0.00	0.00	0.00	2.00
0.00	0.00	−2.00	0.00	2.00	0.00	0.00	0.00	0.00	2.00	0.00	0.00	0.00	0.00	0.00	−2.00
−2.00	0.00	0.00	0.00	0.00	0.00	2.00	0.00	0.00	0.00	0.00	2.00	0.00	−2.00	0.00	0.00
0.00	0.00	0.00	2.00	0.00	−2.00	0.00	0.00	2.00	0.00	0.00	0.00	0.00	0.00	−2.00	0.00
0.00	2.00	0.00	0.00	0.00	0.00	0.00	−2.00	0.00	0.00	2.00	0.00	−2.00	0.00	0.00	0.00
0.00	−2.00	0.00	0.00	0.00	0.00	0.00	2.00	0.00	0.00	−2.00	0.00	2.00	0.00	0.00	0.00
0.00	0.00	0.00	−2.00	0.00	2.00	0.00	0.00	−2.00	0.00	0.00	0.00	0.00	0.00	2.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

to be used in the construction of **U**. In principle, this may not always be possible, and for such cases one would need to include an orthogonalization routine to be used when needed. In practice, however, such cases are certainly rare (if indeed they exist at all) when the cartesian axes are chosen to correspond to symmetry directions in the crystal. This is discussed in greater detail in Section 4.

For the second run in which CALCU = .TRUE. the arrays MULT and IVEC must be assigned the appropriate values and used as input to BLOCDI. MULT(I, J) is the number of mutually orthogonal vectors in the (I, J) set where I refers to the site symmetry and J to the irreducible representation. In our example, since the dimension of all irreducible representations is 1, the values for MULT are those shown in Table 6. IVEC(I, J, K) denotes which vectors are selected from the (I, J) set to be used in the construction of **U**. For example, from the (2, 3) set in Table 7 we may choose the first and second vectors and thus require IVEC(2, 3, 1) = 1 and IVEC(2, 3, 2) = 2 or we could have selected the fourth and fifth vectors which give IVEC(2, 3, 1) = 4 and IVEC(2, 3, 2) = 5.

The second run of BLOCDI, with CALCU = .TRUE. and the values of the arrays MULT and IVEC appropriately assigned, determines the matrix **U** which block diagonalizes the dynamical matrix. The *nonzero* values of **U** are stored in the singly dimensional array U(I), and the i, j indices of U_{ij} corresponding to U(I) are given by II(I) and JJ(I). The number of nonzero values is given by KOUNT.

We now discuss the use of BLOCDI for constructing **U**(**q**) for nonzero wave vectors. For nonzero wave vectors additional degeneracies not predicted from the group $F_{\mathbf{q}}$ are possible owing to time reversal symmetry. BLOCDI gives the group theoretical reduction due to $F_{\mathbf{q}}$.

We illustrate the use of BLOCDI for nonzero wave vectors by considering phonons with wave vectors along the [001] symmetry axis in the example crystal given above. It is customary to denote wave vectors in this direction by Λ if it is within the Brillouin zone and by Z if it is on the zone boundary [10]. If $\mathbf{q} = \mathbf{q}_Z$ then the problem is no different from that discussed above, for $F_{\mathbf{q}}$ is the same whether $\mathbf{q} = \mathbf{q}_Z$ or $\mathbf{q} = 0$. Recall that $F_{\mathbf{q}}$ is that subgroup of F for which $\mathbf{q}^o = \mathbf{q} + \mathbf{Q}$ where **Q** is a reciprocal lattice vector and $\mathbf{q}^o$ is defined by Eq. (5). For those elements involving the inversion operation (see Table 2), $\mathbf{q}_Z{}^o = -\mathbf{q}_Z$, but since $2\mathbf{q}_Z$ is a reciprocal lattice vector, $\mathbf{q}_Z{}^o = \mathbf{q}_Z + \mathbf{Q}$ is satisfied. $\mathbf{q}_Z$ is unchanged (differs by $\mathbf{Q} = 0$) by those elements not involving the inversion operation. Thus $F_{\mathbf{q}_Z} = F$ just as $F_0 = F$ and hence $\mathbf{U}(\mathbf{q}_Z) = \mathbf{U}(0)$.

For $\mathbf{q} = \mathbf{q}_\Lambda$, $F_{\mathbf{q}_\Lambda}$ contains those elements of F that do not involve the inversion operation. The input to BLOCDI is determined in a manner completely analogous to that described above for $\mathbf{q} = 0$. The only differences are (1) the character table is that for the point group C_4, and (2) a new table similar to Table 2 must be constructed in which the equivalent positions and site symmetries are based on the elements in $F_{\mathbf{q}_\Lambda}$. The new character table is that portion of Table 1 for

TABLE VIII

The Number, Site Symmetry and Coordinates for Occupied Equivalent positions, in analogy with Table 2, but for Wave Vector $\mathbf{q} = \mathbf{q}_A$. Also listed are the Symbols for the nontranslational Parts of $F_{\mathbf{q}_A}$ as they Correspond to the Coordinates of Equivalent Positions for the Most General Site Symmetry.

Number of Positions	Site Symmetry Index I, for Occupied Sites	Coordinates of Equivalent Positions	Symbol for the nontranslational part of the elements in $F_{\mathbf{q}_A}$
		(y_1, y_2, y_3)	E
	$I = 3$	$(y_2 - y_3, y_1 - y_3, -y_3$	C_2
4	Occupied by D_I and D_{II}	$(2 + y_2, 6 + y_2 - y_3, 4 - y_1 + y_2)$	C_4
	atoms	$(2 + y_1 - y_3, 6 + y_1, 4 + y_1 - y_2)$	C_4^3
	$I = 2$		
2	Occupied by B_I, B_{II}, C_I and C_{II} atoms.	$(x, x, 0); (2 + x, 6 + x, 4)$	
	$I = 1$		
2	Occupied by A atoms	$(4, 4, 0); (6, 2, 4)$	

which I and J take on values from 1–4. Of course, we now have NIREP = 4 and NING = 4. The table analogous to Table 2 is shown in Table 8.

The values for the arrays IFE and ITRAN are obtained from the equivalent positions for the most general site symmetry ($I = 3$ in Table 8) exactly as they were obtained for zero wave vector problem using Table 2. As before we have three occupied symmetries, so NOFSG = 3. However, now the eight D atoms must be treated as two sets of four atoms labeled D_I and D_{II}. Similarly the four B (four C) atoms become two B_I and two B_{II} (two C_I and two C_{II}) atoms. Thus we now have NINSG(3) = 4, NINSG(2) = 2, NIONSG(3) = 2, and NIONSG(2) = 4. On the other hand, the $\mathbf{q} = \mathbf{q}_A$ vibrations do not destroy the equivalence of the A atoms so we have NINSG(1) = 2 and NIONSG(1) = 1 as we did for $\mathbf{q} = 0$. The representative positions, LX(I, J, K), for the three site symmetries may be determined by again selecting $y_1 = 1$, $y_2 = 2$, $y_3 = 3$ and $x = 1$. In the discussion for $\mathbf{q} = 0$ we described in detail how the values for k, used in the definition of the dynamical matrix, must be chosen in a manner consistent with (1) the order of the site symmetries, and (2) the order of the representative positions within a given site symmetry. We emphasize this point again here. In fact, as a consequence of the order chosen for the representative positions of site symmetry, $I = 2$, for $\mathbf{q} = 0$; the resulting values assigned to k are not allowed for $\mathbf{q} = \mathbf{q}_A$. Specifically, the k assignment given above has $k = 3$, 4, 5 and 6 corresponding, respectively, to B_I, B_{II}, B_I and B_{II} atoms whereas the allowed k assignment for $\mathbf{q} = \mathbf{q}_A$ must

be such that the B_I atoms and the B_{II} atoms each have consecutive values, and similarly for the C_I and C_{II} atoms.

Clearly, the values for LATCON, T, and TI for $\mathbf{q} = \mathbf{q}_A$ are the same as for $\mathbf{q} = 0$. Also, the value given by INV(I) for $I = 1, 2, 3$ and 4 are the same as given above.

This completes the definition of all the input variables for running BLOCDI with CALCU = .FALSE. for $\mathbf{q} = \mathbf{q}_A$. The arrays IVEC and MULT required for the second run, with CALCU = .TRUE., are determined in the same way as we described for $\mathbf{q} = 0$.

IV. How Subroutine BLOCDI Works

The purpose of this section is not to give an extremely detailed account of how BLOCDI works, but rather, to relate in a general way the computations to the theory. Knowing the general theory and the definitions of the input and output variables (Sections 2 and 3), a reader with a basic understanding of FORTRAN programming may readily determine the details.

Subroutine BLOCDI is listed in the appendix. The cards are numbered sequentially from 1–135 in the right-hand columns. All dimensioned variables except NX, FEE, KFE, PJ and QJ are in the common block and discussed in detail in Section 3. NX is always dimensioned NX(3), and FEE is given the same dimensions as IFE. The remaining variables are dimensioned KFE(NOFSG, NING, K), PJ(K3, K, 3), and QJ(K, 3) where K is the maximum value of NINSG(I) and K3 is 3 times K. Notice that CHAR, U, PJ and QJ are complex variables. However, if the character table is real, these variables may all be specified as reals. In the following discussion we use the card numbers to refer to different sections of the subroutine; for example, (9–40) designates that section of BLOCDI beginning with card number 9 and ending with card number 40.

Section (9–40) determines the representation matrices defined by Eq. (4). There is no need to assign values to all elements of the matrices $\mathbf{R}(\rho)$. All that is required are the θ^p matrices and κ_ρ. Furthermore, κ_ρ does not need to be specified in its entirety since k and k_ρ must refer to dynamically equivalent atoms. Section (9–32) computes an array KFE(I, J, K) which contains the information in k_ρ. The indices I, J and K refer, respectively, to the site symmetry, the elements in $F_\mathbf{q}$ and k. K is not the exact value of k (Eq. (10)) but takes on values from 1 to NINSG(I) corresponding to the equivalent positions in the representative sets, LX. KFE is computed by applying the operations of $F_\mathbf{q}$ (IFE and ITRAN) to the positions LX, to determine new positions NX. For each new position the LX positions are searched to see which is the same as the new position, and its index is the value assigned to KFE.

Section (33–40) computes the matrices θ^o (FEE in BLOCDI) from the corresponding integer matrices IFE, using T and TI. Recall that IFE gives the nontranslational parts of the elements in F_q with respect to primitive axes, while the θ^o are with respect to the chosen cartesian system.

If CALCU = .TRUE., card number 41 transfers control to card 85, and U, II, JJ and KOUNT are computed in section (85–131). If CALCU = .FALSE., section (42–83) is executed where the multiplicities are computed and possible symmetry coordinates printed. The multiplicities (MULT and MNU) and the coordinates (PJ) are determined using Eqs. (8) and (7) as described in Section 3. Here PJ(I, J, K) is the (J, K) component of the Ith coordinate where J labels the atoms in the representative sets LX, and K labels the cartesian components.

Clearly, the format for printing these coordinates, or column vectors, may be changed if desired. For example, if one wants six column vectors across the printed page, then the 8 in cards 65, 70 and 79 must be replaced by 6. Also, if PJ is real, then these cards may be appropriately changed to accomodate more columns per page. The term 0.001 in card 82 is to insure that MULT will not be rounded down to the wrong integer value as a result of round of error in updating CA. Although CA is specified as complex, its imaginary component will be zero and its real part very near integer values when card 82 is executed, provided, of course, the input is correct. In fact, CA may be printed at this point to isolate input errors according to site symmetry and irreducible representations. Another check on the validity of the input is to check to be sure that the number of independent vectors in each set of PJ's is given by the multiplicity times the dimension of the irreducible representation for that set.

Now we focus our attention on the section (85–133) which is executed when CALCU = .TRUE.. Here, the coordinates QJ, to be used in the construction of the matrix $\mathbf{U}(\mathbf{q})$ are computed. Each nonzero component of QJ is a nonzero element in U which is stored in the singly dimensioned array U. The corresponding i and j indices of U_{ij} are computed and assigned to the variables II and JJ. The number of these nonzero elements is given by KOUNT.

These coordinates are computed in the same way as the PJ's in (49–64) except that only the selected ones are computed. The selection is made using the additional input, MULT and IVEC (see (97–102)). Section (114–118) computes the magnitude of QJ and section (119–131) assigns the appropriate values to U, II and JJ and updates KOUNT. Due to card number 126, those values with magnitude less than 0.00001 are not included in U.

In Section 3 we mentioned that if it is not possible to select, by inspection, the required number of *orthogonal* coordinates from the printed sets then some modification of BLOCDI would be necessary. If the selected coordinates are linearly independent but not necessarily orthogonal, then each QJ must be orthogonalized with respect to the previously computed QJ's in one execution of

DO loop 190. Alternatively, one could rewrite BLOCDI following card 32 to first compute the multiplicities and then compute, select and orthogonalize the required number (based on the multiplicities) of symmetry coordinates, and finally, determine U, II, JJ and KOUNT from these coordinates. While we do not rule out the possible necessity for such modifications in future studies, we believe that such cases will be rare and thus such a generalization would be unwarranted. This contention is based on previous lattice dynamical studies of $Gd_4(MoO_4)_6$ [11] (space group P42/*m*) and of two phases of $Ca_{10}(PO_4)_6F_2$ [12, 13] having the hexagonal space groups $P6_3/m$ and $P6_3/mcm$ as well as on the example reported here.

APPENDIX: A Listing of Subroutine BLOCDI

```
      SUBROUTINE BLOCDI (CALCU)                                          0010
      LOGICAL CALCU                                                      0020
      COMPLEX CHAR, CMPLX, CONJG, COM, U, PJ, CA, QJ                     0030
      DIMENSION CHAR(8, 8), U(500), II(500), JJ(500), T(3, 3), TI(3, 3), LX(3, 8, 3  0040
     1), NX(3), NINSG(3), NIONSG(3), ITRAN(8, 3), IFE(8, 3, 3), KFE(3, 8, 8), PJ(24,  0050
     28, 3), MNU(8), INV(8), FEE(8, 3, 3), MULT(3, 8), IVEC(3, 8, 3), QJ(8, 3)  0060
      COMMON/MB/CHAR, U, T, TI, II, JJ, MNU, NINSG, NIONSG, NING, NIREP,  0070
     1IFE, ITRAN, INV, LX, LATCON, MULT, KOUNT, IVEC, NOFSG              0080
      DO 30 INS = 1, NOFSG                                               0090
      NIN = NINSG(INS)                                                   0100
      DO 30 I = 1, NING                                                  0110
      DO 30 K = 1, NIN                                                   0120
      IA = INV(I)                                                        0130
      DO 40 J = 1, 3                                                     0140
      NX(J) = ITRAN(IA, J)                                               0150
      DO 41 L = 1, 3                                                     0160
   41 NX(J) = NX(J) + IFE(IA, J, L) *LX(INS, K, L)                       0170
   43 IF(NX(J). LT. LATCON) GO TO 42                                     0180
      NX(J) = NX(J) - LATCON                                             0190
      GO TO 43                                                           0200
   42 IF(NX(J). GE. 0) GO TO 40                                          0210
      NX(J) = NX(J) + LATCON                                             0220
      GO TO 42                                                           0230
   40 CONTINUE                                                           0240
      DO 50 L = 1, NIN                                                   0250
      IF(NX(1). NE. LX(INS, L, 1)) GO TO 50                              0260
      IF(NX(2). NE. LX(INS, L, 2)) GO TO 50                              0270
      IF(NX(3). NE. LX(INS, L, 3)) GO TO 50                              0280
      GO TO 60                                                           0290
   50 CONTINUE                                                           0300
   60 KFE(INS, I, K) = L                                                 0310
   30 CONTINUE                                                           0320
      DO 70 I = 1, NING                                                  0330
```

```
      IA = INV(I)                                                        0340
      DO 70 J = 1, 3                                                     0350
      DO 70 K = 1, 3                                                     0360
      FEE(I, J, K) = 0.0                                                 0370
      DO 70 L = 1, 3                                                     0380
      DO 70 M = 1, 3                                                     0390
   70 FEE(I, J, K) = FEE(I, J, K) + T(J, L)*IFE(IA, L, M)*TI(M, K)       0400
      IF(CALCU) GO TO 110                                                0410
      DO 100 IR = 1, NIREP                                               0420
      MNU(IR) = 0                                                        0430
      DO 100 IOS = 1, NOFSG                                              0440
      CA = CMPLX(0.0, 0.0)                                               0450
      NIN = NINSG(IOS)                                                   0460
      NIN 3 = NIN*3                                                      0470
      NION = NIONSG(IOS)                                                 0480
      DO 80 K = 1, NIN                                                   0490
      DO 81 J = 1, NIN3                                                  0500
      DO 81 L = 1, 3                                                     0510
   81 PJ(J, K, L) = CMPLX(0.0, 0.0)                                      0520
      DO 80 IG = 1, NING                                                 0530
      COM = CHAR(IR, IG)                                                 0540
      COM = CONJG(COM)                                                   0550
      KK = KFE(IOS, IG, K)                                               0560
      DO 82 J = 1, 3                                                     0570
      JX = (J - 1)*NIN + KK                                              0580
      DO 82 L = 1, 3                                                     0590
   82 PJ(JX, K, L) = PJ(JX, K, L) + COM*FEE(IG, J, L)                    0600
      IF(KK. NE. K) GO TO 80                                             0610
      DO 84 I = 1, 3                                                     0620
   84 CA = CA + FEE(IG, I, I) * COM                                      0630
   80 CONTINUE                                                           0640
    1 FORMAT(1X, 8(2F6.2, 1 X))                                          0650
    2 FORMAT(1H )                                                        0660
    3 FORMAT('- IRREP NO. ', I2,'  SITE SYMMETRY NO. ', I2)              0670
      WRITE(6, 3) IR, IOS                                                0680
      IMIN = 1                                                           0690
      IMAX = 8                                                           0700
   91 IF(IMAX. GE. NIN3) IMAX = NIN3                                     0710
      DO 90 J = 1, 3                                                     0720
      DO 90 K = 1, NIN                                                   0730
      WRITE(6, 1)  (PJ(I, K, J), I = IMIN,  IMAX)                        0740
   90 CONTINUE                                                           0750
      WRITE(6, 2)                                                        0760
      IF(IMAX. EQ. NIN3) GO TO 92                                        0770
      IMIN = IMAX + 1                                                    0780
      IMAX = IMAX + 8                                                    0790
      GO TO 91                                                           0800
```

```
 92 CONTINUE                                                    0810
    MULT(IOS, IR) = CA/NING + 0.001                             0820
100 MNU(IR) = MNU(IR) + MULT(IOS, IR) *NION                     0830
    RETURN                                                      0840
110 NK = 0                                                      0850
    DO 175 I = 1, NOFSG                                         0860
175 NK = NK + NINSG(I) *NIONSG(I)                               0870
    JK = 0                                                      0880
    KOUNT = 0                                                   0890
    DO 200 IR = 1, NIREP                                        0900
    NKP = 0                                                     0910
    DO 200 IOS = 1, NOFSG                                       0920
    NIN = NINSG(IOS)                                            0930
    NION = NIONSG(IOS)                                          0940
    I = IOS - 1                                                 0950
    IF(I. GT. 0) NKP = NKP + NINSG(I) *NIONSG(I)                0960
    JMAX = MULT(IOS, IR)                                        0970
    IF(JMAX. EQ. 0) GO TO 200                                   0980
    DO 190 J = 1, JMAX                                          0990
    IV = IVEC(IOS, IR, J)                                       1000
    JV = (IV + NIN - 1)/NIN                                     1010
    KV = IV - (JV - 1) *NIN                                     1020
    DO 180 K = 1, NIN                                           1030
    DO 181 L = 1, 3                                             1040
181 QJ(K, L) = CMPLX(0.0, 0.0)                                  1050
    DO 180 IG = 1, NING                                         1060
    KK = KFE(IOS, IG, K)                                        1070
    IF(KK · NE · KV) GO TO 180                                  1080
    COM = CHAR(IR, IG)                                          1090
    COM = CONJG(COM)                                            1100
    DO 182 L = 1, 3                                             1110
182 QJ(K, L) = QJ(K, L) + COM*FEE(IG, JV, L)                    1120
180 CONTINUE                                                    1130
    R = 0.0                                                     1140
    DO 187 K = 1, NIN                                           1150
    DO 187 L = 1, 3                                             1160
    COM = QJ(K, L)                                              1170
187 R = R + COM*CONJG(COM)                                      1180
    R = 1.0/SQRT(R)                                             1190
    DO 186 M = 1, NION                                          1200
    JK = JK + 1                                                 1210
    DO 186 I = 1, NIN                                           1220
    DO 186 L = 1, 3                                             1230
    COM = QJ(I, L)*R                                            1240
    RR = COM*CONJG(COM)                                         1250
    IF(RR.LT.1.0E-10) GO TO 186                                 1260
    KOUNT = KOUNT + 1                                           1270
    JJ(KOUNT) = JK                                              1280
```

```
      II(KOUNT) = (L - 1)*NK + NKP + (M - 1)*NIN + I                    1290
      U(KOUNT) = COM                                                    1300
  186 CONTINUE                                                          1310
  190 CONTINUE                                                          1320
  200 CONTINUE                                                          1330
      RETURN                                                            1340
      END                                                               1350
```

REFERENCES

1. M. BORN AND K. HUANG, "Dynamical Theory of Crystal Lattices," Oxford Univ. Press, London, 1954.
2. Time reversal symmetry, which we do not consider here, can sometime aid in the reduction of the dynamical matrix. For a general discussion see [6].
3. J. W. LEECH AND D. J. NEWMAN, "How to Use Groups" Mathuen, London, 1969.
4. T. G. WORLTON AND J. L. WARREN, *Comput. Phys. Comm.* **3** (1972), 88.
5. L. L. BOYER, *Phys. Rev. B* **9** (1974), 2684.
6. A. A. MARADUDIN AND S. H. VOSKO, *Rev. Mod. Phys.* **40** (1968), 1.
7. J. L. WARREN, *Rev. Mod. Phys.* **40** (1968), 38.
8. J. L. BIRMAN, Symmetry changes, phase transitions and ferroelectricity, *in* "Ferroelectricity" (E. F. Weller, Ed.), p. 20, Elsevier, Amsterdam/London/New York.
9. "International Tables for X-ray Crystallography" (N. F. M. Henry and K. Lonsdale, Eds.), Kynoch Press, Bermingham, England.
10. G. F. KOSTER, Space groups and their representations, *in* "Solid State Physics" (F. Seitz and D. Turnbull, Eds.), Vol. 5, p. 173, Academic Press, New York, 1957.
11. H. HEINISH AND J. R. HARDY, Dept. of Physics, University of Nebraska, Lincoln, private communication.
12. L. L. BOYER, *Phys. Stat. Solidi (a)* **20** (1973), 555.
13. L. L. BOYER AND P. A. FLEURY, *Phys. Rev. B.* **9** (1974), 2693.