APPENDIX A

ORTEP-III SUBPROGRAMS

`FUNCTION ARCCOS(X)`	Computes θ, the arc cosine of X in degrees; $0 \le \theta \le 180°$.
`ATOM(QA,Z)`	Finds the triclinic coordinates **Z** for the atom described by the atom designator code QA.
`AXEQB(A1,X,B1,JJJ)`	Solves the matrix equation **A1 X** = **B1** for **X**. The matrices **B1** and **X** are (3,JJJ) and **A1** is always (3,3). To invert **A1**, make **B1** an identity matrix.
`AXES(U,V,X,ITYPE)`	Provides three orthogonal column vectors in **X**, each 1 Å long, from the two vectors **U** and **V**. ITYPE > 0: Cartesian system ITYPE < 0: triclinic system |ITYPE| = 1: $\mathbf{X}_1 = \mathbf{U}$; $\mathbf{X}_2 = (\mathbf{U} \times \mathbf{V})$; $\mathbf{X}_3 = \mathbf{U} \times (\mathbf{U} \times \mathbf{V})$ |ITYPE| = 2: $\mathbf{X}_1 = \mathbf{U}$; $\mathbf{X}_2 = (\mathbf{U} \times \mathbf{V}) \times \mathbf{U}$; $\mathbf{X}_3 = \mathbf{U} \times \mathbf{V}$ ITYPE = 0: same as type 2 except **U** = *a* crystal axis, **V** = *b* crystal axis.
`BOND(Z1,Z2,NB,NA1,NA2)`	Draws a bond, described by Format No. 2 trailer card number NB, between two atoms. Z*n* is atom designator code of atom *n*, and NA*n* is number of atom *n* in ATOMS array.
`COLRxx(ICOLOR)`	Sets plot color on "device" *xx* to ICOLOR.
`CURSSC`	Identifies atoms selected on screen display.
`DFLTS`	Sets default values for items requested from user.
`DIFV(X,Y,Z)`	Performs the vector subtraction **X** – **Y** = **Z**. **Z** may have the same location as **X** or **Y**.
`DRAW(W,DX,DY,NPEN)`	Interconnects ORTEP and the plot package. It also prevents the pen from crossing the boundaries. If the indicator ITILT in COMMON is zero, the array W contains *x* and *y* in plotter coordinates. While perspective lettering is being plotted, ITILT ≠ 0; and W contains *x*, *y*, *z* in Cartesian coordinates, which will be rotated and projected by DRAW to form plotter x,y coordinates. DX and DY are added to the plotter *x* and *y*, respectively, before the plot package is called. NPEN = 2 for pen down and 3 for pen up.

`EDITR`	Controls ORTEP line editor.
`EIGEN(W,VALU,VECT)`	Determines the three eigenvalues **VALU** and the three column eigenvectors **VECT** of the matrix **W**. Indeterminate eigenvectors are replaced by zeros and the fault indicator NG set to a negative value (eigenvectors are assigned for the indeterminate cases by PRELIM).
`END`*xx*	Terminates plotting on "device" *xx*.
`ERPNT(TD,N)`	Prints error message when a fault is found. The arguments identify the atom designator code TD and the instruction N involved in the fault. The fault indicator, NG, is in COMMON.
`EXITNG(ING)`	Prints fault indicator ING if abnormal termination and stops program execution.
`F200`	Executes the 200 series instructions.
`F400`	Executes the 400 series instructions.
`F500`	Executes all 500 series instructions.
`F600`	Executes all 600 series instructions.
`F700`	Executes all 700 series instructions.
`F800`	Executes all 800 series instructions. Bonds to be drawn are found by F800, then drawn by subroutine BOND.
`F900`	Executes all 900 series instructions.
`F1000`	Executes the 1001 instruction.
`FUNCTION IEND(STRING)`	Returns the position of the last non-space character in a character STRING.
`INIT`*xx*	Initializes plotting on "device" *xx*.
`LAP500(NTYPE)`	Sorts the ATOMS array, then calculates the projected outline ellipses for all atoms in the ATOMS array. The ellipses are stored in the CONIC array along with the minima and maxima in x and y for a rectangle enclosing each ellipse. If NTYPE < 0, previous overlap information is cleared.

`LAP700(NA,ICQ)`	Finds the atoms that overlap a given atom to be drawn. The routine first checks the bounding rectangles for intersections, then forms the cubic discriminant from the quadratic descriptions of the two projection ellipses. The discriminant provides a specification for complete overlap, partial overlap, or no overlap. A list of up to 20 interfering ellipses is compiled. NA is the atom to be drawn. ICQ is set > 0 if overlap exists and = -1 if not.
`LAP800(NA1,NA2,ICQ)`	Used in the "Projected Outline Storage Step" to store the projected quadrangles for the bonds specified by the trailer cards of the 1001, 821, and 822 instructions. The routine also is used in the "Area-Overlap Search Step" to find the projected bond quadrangles that overlap a given bond to be drawn. A list containing up to 30 interfering quadrangles is compiled. NA1 and NA2 are the two atoms of the bond. ICQ is set > 0 if overlap exists and = -1 if not.
`LAPAB(IQ,IA,ICQ,ITY)`	Finds the bonds that overlap an atom to be drawn and the atoms that overlap a bond to be drawn. It is used in the "Area-Overlap Search Step." ITY > 0 checks for atom, IA, over bond, IQ, and ITY < 0 checks for bond over atom. ICQ is set > 0 if overlap exists, = 0 for no overlap, and < 0 for hidden atom or bond.
`LAPCON(CON1,CON,Y,OVMR)`	Transforms conic, CON1, to plotter homogeneous coordinate system, CON, with center at Y. OVMR denotes overlap margin.
`LAPDRW(Y,NPEN,NCQ)`	Checks each line segment to be drawn for intersection with the interfering ellipses and quadrangles and compiles a list of intersections. The intersection list is sorted according to distance along the line segment, and the intersection pattern is analyzed to determine which subsegments are visible and which are hidden. The line subsegments are passed to the SCRIBE routine. Y is pen position, NPEN denotes if pen is up or down, and NCQ is set to NCOVER+NQOVER.
`CHARACTER*(*) FUNCTION MAKSYM(GP)`	Returns a character string representation (xyz notation) of a symmetry operator stored in ORTEP's internal representation in array GP.
`MM(X,Y,Z)`	Performs the matrix multiplication **XY** = **Z**. The location of **Z** must be different from **X** and **Y**.

`MV(X,Y,Z)`	Performs the matrix-vector multiplication **XY** = **Z**. The location of **Z** must be different from **X** and **Y**.
`NORM(X,Y,Z,ITYPE)`	Stores at **Z** a vector (not necessarily a unit vector) perpendicular to both **X** and **Y**. The sense of **Z** is that of the vector product **X** × **Y**. ITYPE > 0: Cartesian system ITYPE ≤ 0: triclinic system
`NUMBUR(W,W2,HGT,DIST,THT,ND)`	Converts number to character string for placement on the drawing. W contains coordinates of the lower left edge of the first character, W2 is unused, HGT is the height of the characters, DIST is the number to be drawn, THT is the angle by which the base line of the characters is to be rotated counterclockwise from the positive *x* axis, and ND is the number of digits to the right of the decimal point.
`ORTEP`	ORTEP is the MAIN program and controlling routine that decodes the ORTEP instructions. It either executes the command directly or calls the appropriate subroutine to execute the instruction.
`PAXES(DCODE,ITYPE)`	Stores the covariance (dispersion) matrix for the thermal ellipsoid or its inverse matrix, which is the matrix of coefficients in the quadratic form describing the ellipsoid, in COMMON at Q for the atom with atom designator code DCODE. ITYPE > 0 for covariance matrix ITYPE < 0 for ellipsoid quadratic form matrix |ITYPE| = 1 based on triclinic system |ITYPE| = 2 based on working Cartesian system |ITYPE| = 3 based on reference Cartesian system
PEN*xx*(X,Y,IPEN)	Controls pen movement on "device" *xx*. X is the abscissa and Y is the ordinate expressed in inches. IPEN=2: pen draws line as it moves IPEN=3: pen moves without drawing line
PENW*xx*(PENW)	Sets pen thickness on "device" *xx* to PENW. PENW is provided in thousandths of an inch. The default is 5.
`PLOT(X,Y,IPEN)`	Calls the appropriate PEN*xx* routine for drawing lines on "device" *xx*. Parameters are sent to PEN*xx*.
`PLTXY(X,Y)`	Calculates the plotter coordinates Y from the unscaled Cartesian coordinates X. The distance to the closest boundary of the plot is stored in the variable EDGE in COMMON.

PRELIM	Performs all calculations to process (e.g., principal axis transformations) and store the input crystallographic parameters.
PRIME	"Primes the program" by initializing all the "primer parameters".
PROJ(D,DP,X,XO,VIEW,I1,I2,I3)	Used to obtain an array, DP, of plotter coordinates from a scaled array, D, of points described in Cartesian coordinates. X, XO, and VIEW are parameters involved in the projection, and I1, I2, I3 are DO loop parameters for indexing through the array.
RADIAL(ND)	Generates a "radial" array (D in COMMON) of points lying on an ellipse, given two conjugate radius vectors of the ellipse in the array DA in COMMON. From 8 to 128 points are generated depending on the value of ND ($1 \leq ND \leq 5$).
READIN(IU,CHEM,ID1,ID2,X1, X2,X3,IT,IS,B1,B2,B3,B4,B5, B6,BTYPE)	Reads atom parameters in any format from a file. This subroutine may be modified by the user. See Section 4.5 for a description of the parameters.
RECYCLE	Returns instruction pointer to 201 instruction and zeroes ATOMS array.
SCRIBE(Y,NPEN)	Filters out the hidden line segments and passes the visible line segments to the DRAW routine.
SEARC	Conducts an exhaustive (but educated) search to find all points within a sphere or rectangular box. Interatomic distances and angles are also calculated for the 100 series.
SIMBOL(W,W2,HGT,ITXT,THT,N)	Processes character strings for placement on the drawing. W contains coordinates of the lower left edge of the first character, W2 is unused, HGT is the height of the characters, ITXT is the string to be drawn, THT is the angle by which the base line of the characters is to be rotated counterclockwise from the positive *x* axis, and N is the number of characters to be drawn.
SPARE(INST)	Expands the user supplied instruction set by responding to any INST $\geq$ 12. INST = instruction /100.
STOR(TD1)	Stores atom with atom designator code TD1 in (or removes atom from) the ATOMS array. Coordinates in whichever system is in use are communicated to STOR via array V1 of COMMON.

`TEPSYM(TXT,NUM,KK)`	Parses symmetry operator in character string representation (xyz notation), TXT, and stores the information in ORTEP's internal representation. NUM is symmetry operator number, and KK is the component number.
`TMM(X,Y,Z)`	Performs the matrix multiplication $(\mathbf{X}^T \mathbf{Y})^T = \mathbf{Z}$. The location of **Z** must be different from **X** and **Y**.
`UINPUT(IN,NOUT)`	Controls user input. IN is input file device number, and NOUT is output file device number.
`UNITY(X,Z,ITYPE)`	Makes the vector **Z** 1 Å long and parallel to **X**. The vectors **X** and **Z** may have the same location. ITYPE > 0: Cartesian system ITYPE < 0: triclinic system
`VM(Y,X,Z)`	Performs the vector-matrix multiplication $\mathbf{Y}^T\mathbf{X} = \mathbf{Z}^T$. The location of **Z** must be different from **Y** and **X**.
`FUNCTION VMV(X1,Q,X2)`	Performs the vector-matrix-vector multiplication $\mathbf{X1}^T \mathbf{Q}\ \mathbf{X2}$ = scalar.
`FUNCTION VV(X,Y)`	Performs the vector-vector multiplication $\mathbf{X}^T\mathbf{Y}$ = scalar.
`XYZ(DQA,X,ITYPE)`	Returns in X coordinates for atom with atom designator code DQA. ITYPE = 0: triclinic coordinates ITYPE = 1 or 2: working Cartesian system coordinates ITYPE = 3: reference Cartesian system coordinates