

# Cameron Manual

Chemical Crystallography Laboratory, Oxford

Tue Jul 5 2005

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	HOW TO CONTROL THE PROGRAM . . . . .	2
1.2	Command Groups . . . . .	2
1.3	ERROR HANDLING . . . . .	3
1.4	The HELP facility . . . . .	4
1.5	Mouse Activation . . . . .	4
1.6	General Input Syntax . . . . .	4
<b>2</b>	<b>How To Get Started</b>	<b>6</b>
<b>3</b>	<b>Data Input</b>	<b>7</b>
3.1	Input of CRYSTALS List 5 files . . . . .	7
<b>4</b>	<b>Outputting Data</b>	<b>8</b>
<b>5</b>	<b>Editing The Atom List</b>	<b>9</b>
<b>6</b>	<b>Obeying Files</b>	<b>10</b>
<b>7</b>	<b>Graphical Output Devices</b>	<b>11</b>
<b>8</b>	<b>View Direction Control</b>	<b>12</b>
8.1	BASIC CONTROL . . . . .	12
8.2	ORIENTATION COMMANDS . . . . .	13
8.3	Other related commands . . . . .	14
8.4	PHOTOGRAPHS . . . . .	15
<b>9</b>	<b>Include And Exclude</b>	<b>16</b>
9.1	Further related commands . . . . .	17
9.2	Generation of Dummy atoms . . . . .	17
9.3	Example . . . . .	18
<b>10</b>	<b>Drawing Style Control</b>	<b>19</b>
10.1	LINE commands. . . . .	19
10.2	BALL commands. . . . .	20
10.3	ELLIPSE commands . . . . .	20

<b>11 Connectivity Control</b>	<b>22</b>
11.1 CONNECT . . . . .	22
11.2 Miscellaneous CONNECT commands . . . . .	25
<b>12 Control Of Colour</b>	<b>26</b>
12.1 Example . . . . .	27
<b>13 Atom Labelling</b>	<b>29</b>
<b>14 Other Picture Controlling Commands</b>	<b>31</b>
<b>15 Symmetry Input</b>	<b>32</b>
15.1 Input of symmetry operators . . . . .	32
15.2 Example of Space Group Input . . . . .	33
15.3 Other symmetry related operations . . . . .	33
<b>16 Crystal Packing Commands</b>	<b>35</b>
16.1 PACK . . . . .	35
16.2 Dealing with connectivity . . . . .	36
16.3 EXAMPLES . . . . .	36
16.4 ENCLOSURE . . . . .	36
16.5 Choice of enclosure "origin" . . . . .	37
16.6 Type of volume to be used . . . . .	37
16.7 EXAMPLE . . . . .	38
16.8 PACKING MORE COMPLICATED STRUCTURES . . . . .	38
16.9 UNPACK . . . . .	39
<b>17 Add And Move - Further Symmetry Related Commands</b>	<b>40</b>
17.1 ADD . . . . .	40
17.2 EXAMPLES . . . . .	41
17.3 MOVE . . . . .	42
<b>18 Distance And Angle Calculations</b>	<b>43</b>
18.1 Method of Calculation . . . . .	43
<b>19 Information On Data Held Within The Program</b>	<b>45</b>
<b>20 Group Definitions</b>	<b>46</b>
20.1 Example . . . . .	46
<b>21 Miscellaneous Commands</b>	<b>48</b>
<b>22 How To Stop The Program</b>	<b>49</b>
<b>23 Menu Definition File</b>	<b>50</b>
<b>24 Some Useful Ideas</b>	<b>51</b>

# Chapter 1

## Introduction

CAMERON was designed and built in the Chemical Crystallography Laboratory, Oxford, by Lisa Pearce, Keith Prout and David Watkin. Many people have contributed ideas for improvements, and Louis Ferrugia pioneered a completely WINDOWS based version.

### 1.1 HOW TO CONTROL THE PROGRAM

The program is controlled by typing commands at the keyboard, by use of a mouse to pick items from a menu, or by using the mouse to manipulate the diagram.

### 1.2 Command Groups

The commands have been collected into groups (twenty-three at present). Each group contains commands and sub-commands which perform related functions or which will act as qualifiers to each other. The command processor will spot invalid combinations of commands and sub-commands and enter error mode. Once in this mode the error can be corrected.

Command groups are used to decide when the user has finished entering all the qualifiers for a particular operation. The qualifiers are assumed to be finished when a command of another group is entered. For example :-

```
XROT 20 YROT 30 VIEW LINE ALL
```

**XROT**, **YROT** and **VIEW** are all in the same group but **LINE** (a command that sets the drawing style) is not. When the command processor reaches the **LINE** command it will then execute the previous commands ie **XROT** to **VIEW**. This is useful because it means that the user does not need to input all the qualifiers on one line, and also because the user does not need to tell the computer when it has finished with an operation - the computer is able to work this out for itself. The input lines :-

```
XROT 20 YROT 30 VIEW LINE
```

and

```
XROT 20  
YROT  
20 VIEW  
LINE
```

will produce exactly the same results.

If the user wishes to execute a group of commands without entering a command of another group this can be done by sending a blank line to the command processor (ie press the RETURN key twice).

```
XROT 20 YROT 30 VIEW <RETURN> <RETURN>
```

will execute the three commands without the need for **LINE**.

## 1.3 ERROR HANDLING

The command processor is able to check for some errors in the commands input by the user. The computer checks for two types of error:

1. Whether a command / sub-command combination is valid.
2. Whether the arguments supplied after a command are valid and also if there is the correct number of arguments.

The action taken after an error depends upon the setting of the **EDIT** flag. If EDIT is OFF, only an error message is displayed. If EDIT is ON, the faulty input can be edited. Once an error has been detected the user can take action and modify the input line so that as many commands as possible can be processed. (If error checking is not carried out at this level, the detection of an error while executing a command will cause that command - and possibly its sub commands - to be lost.)

The user is given three options once an error has been detected.

### Abandon

Do not execute any more commands not yet executed.

### Edit

The action taken by choosing the second option will depend on the nature of the error. If a word has been input that the computer has not recognised the user will be asked to enter an alternative word. If <return> is used then the word is replaced by a blank space - it is effectively deleted from the input line. If too many arguments are supplied the user is given the opportunity to delete the excess arguments. Alternatively if arguments are missing the user will be asked to input them.

### Help

Help information is supplied if the user requests it. This will say what arguments, if any, are required by the current command. Also, if any sub commands are valid in the context of the error these are listed.

For example, the input line

```
XROT 10 YROT ZROT 15 VIEW
```

contains an error - an argument is missing after **YROT**. The program will give the user three options :-

Abandon	Don't execute any of the commands
Edit	The user can supply the needed argument and the new line is then processed.
Help	This will display the help information

This command requires one numeric argument - the angle of rotation in degrees.

## 1.4 The HELP facility

The CAMERON **HELP** facility can be entered in one of two ways, either once an error has occurred (as described above) or by requesting it directly. This second way of accessing HELP is achieved by entering '**HELP**' or '?' at any point in the input line. If a command follows '**HELP**' / '?' then information is given on this command - otherwise information is given on the previous command (if any). If no HELP is available (because the command is not recognised) then the user is supplied with general HELP information. The only difference between '**HELP**' and '?' is in the detail of the information supplied. Each command comes with a help information line. Entering '?' will provide the user with the information line for the command on which you have requested HELP and a list of the sub-commands that are valid at this point. Entering '**HELP**' gives the same information except that the help line for each of the sub-commands shown is also listed. '**HELP**' will therefore provide the user with more detailed information than '?' will.

Once the HELP information has been read the user may 'Continue' or 'Abort'. The later two options are the same as for Error Handling and are described above. 'Continue' simply removes the word '**HELP**' or '?' (and any following words) from the input line and the user can carry on as before.

## 1.5 Mouse Activation

The Mouse can be used to pick out atoms (and elements if a KEY is present). In some cases this is much quicker than typing on the keyboard - especially if the user does not know the names of the atoms concerned. The mouse cursor (an arrow) is present once an appropriate output device has been chosen.

For some commands the mouse is used for functions other than atom/element picking. In such cases - eg text positioning and labelling - the mouse is activated by typing MOUSE as a separate command.

## 1.6 General Input Syntax

When the atoms are in their initial state (ie no symmetry operators have been applied) they are referred to by their normal atom or element names. With commands that allow elements to be affected eg COLOUR, BALL you may use a '\*' to refer to ALL atoms. For example CONNECT O \* 0.0 3.0 will find all bonds between oxygen and any other atom within 3 angstroms. A '\*' can also be used to refer to GROUP names (see DEFGROUP later).

Once pack operations have been carried out the atoms are identified differently. During PACK and ENCLOSURE the user is provided with information similar to :-

```
2 additional symmetry generated units.
```

The atoms generated will be referred to by suffixing their packnumber eg O1\_2. These numbers can be shown by using LABEL GENERATED. Therefore, consider an asymmetric unit which contains N1 and O2. The following will be available after packing:-

```
N1_1      the single atom
N1         all N1 atoms in the packed structure
N          all the nitrogen atoms
N_2       all the nitrogen atoms with pack number 2
*_1       all the atoms with pack number 2
```

These pack numbered atom identifiers are produced by mouse clicking on the atoms concerned. The symmetry operation used to generate the atoms can be found from INFO PACKNUMBER. The operators are also written to the CRYSTALS listing file

When CAMERON is parsing the commands input it checks to see whether the next work on the line is a command or not. Only the first 4 characters of each word are significant. Therefore, the processor will not know the difference between the command COLOUR and the file COLOUR.PST. If you require the later then it must be enclosed in quotes - eg COPY "COLOUR.PST".

## Chapter 2

# How To Get Started

There is a basic 'startup' procedure that can be followed in order to get a picture on to the screen. The following steps are required :-

1)            Load in a spacegroup if required 2)            Load in the unit cell dimensions 3)            Load in the atomic

This can be accomplished with the following commands, which are illustrated for a compound which has a spacegroup P 21, unit cell parameters a=6.0, b=7.0, c=8.0, alpha=90, beta=115, gamma=90. The atomic coordinates are held in a "list5" type file NIGEL.L5.

```
SPACEGROUP P 21 INPUTCELL 6 7 8 90 115 90 LIST5 NIGEL.L5 DEVICE VGABALL ALLFILLVIEW
```

From here the view direction can be controlled via XROT, YROT, ZROT or CURSOR. If you require a picture of all the atoms in the unit cell, and the unit cell also displayed, the commands are :-

```
PACK CELL INCLUDE CELLVIEW
```



## Chapter 3

# Data Input

At present data is read from CRYSTALS generated LIST5 files and CSSRfiles (see the CRYSTALS manual).

### 3.1 Input of CRYSTALS List 5 files

These files do not contain information on the dimensions of the unit cell. It is therefore necessary to use CELL to enter this information prior to the reading of the list 5 file.

#### INPUT

**CELL** The syntax of this sub-command is :-

INPUT CELL a b c alpha beta gamma

ie we require six arguments - cell dimensions in angstroms and angles in degrees. Their values are assigned as shown above.

**LIST5** This command requires as its argument the name of the file that contains the atomic coordinates. The syntax is :-

LIST5 filename.L5

NB The LIST5 command will not be accepted without the previous command CELL.

#### CSSR

CSSR files are input by using INPUT CSSR. As with LIST5 files above it is advisable to input the symmetry information before inputting the atomic coordinates.

Related commands : OBEY, OUTPUT

## Chapter 4

# Outputting Data

### OUTPUT

CAMERON outputs data in CRYSTALS LIST5 and CSSR format.

***LIST5*** This outputs a crystals list5. It must be followed by a filename. The list5 will contain LIST1 unit cell information at the end - such a file can be re-read by CAMERON with no problem.

### ***CSSR***

### ***FRACT***

***ORTH*** These subcommands are needed to specify the coordinate type for the CSSR file ie orthogonal or fractional. The command is followed by the filename of the output file.

Example: `OUTPUT CSR FRACT mydata.css`

Related commands : INPUT

## Chapter 5

# Editing The Atom List

The user is able to edit the names and elemental types of the atoms in the current list. The following commands are available -

### **RENAME**

The syntax is:

```
RENAME at1 at2      or: RENAME e11 e12.
```

Arguments are entered in pairs - the first is renamed to the second - and must be like with like. i.e. you cannot rename an atom to an element or vice versa.

### **RETYPE**

This allows the user to change the element of a list of atoms.

**ATOMS** The syntax is -

```
RETYPE element ATOM at1 at2 at3
```

## Chapter 6

# Obeying Files

### OBEY

If required the user is able to hand over control of CAMERON to an external file. This is particularly useful if a set of operations is to be carried out on a number of pictures because the file need only be generated once and then it can be used over and over again. At the present time CAMERON is able to read files in its own format - ie in with the same syntax as the commands input at the keyboard - or it can read a SNOOPI.INI file as output by CRYSTALS. The syntax is :-

**OBEY nnn.nn**

Note that any commands entered after OBEY (on the same line) are ignored. This is to allow for possible errors in the OBEY file - errors will cause the OBEYed file to be closed. OBEY files may themselves contain the names of other files to be OBEYed.

Related commands : LOG

## Chapter 7

# Graphical Output Devices

The user has to specify a graphical output device before the command **VIEW** can be used. This is done with the command **SCREEN**. Note that if the title screen file **CAMERON.LJP** is present the screen is automatically set to **VGA**.

### **SCREEN**

The following devices are currently supported:-

**VGA** This sends the output to a VGA monitor eg to a PC graphics screen.

**EGA** This sends the output to an EGA monitor.

**SIGMA** This sends the output to a SIGMA terminal.

**TEK** This is used for VAX terminals with Tektronics 4010 emulation.

### **HARDCOPY**

If hardcopy output is required the user must specify a separate device which is then drawn to using **COPY**. Those available are :-

**POSTSCRIPT** The graphical output is sent to a file in POSTSCRIPT format so that it can be printed on a postscript printer.

**ENCAPPOST** The graphical output will be sent to a file in the same way as with POSTSCRIPT, the only difference is that the resulting file will be in Encapsulated form so that it can be incorporated into other packages eg Word Perfect (using the Graphics range of commands).

**CPOST** This generates a file in Colour Postscript (Level 1) format.

**CENCAP** This generates an colour encapsulated postscript file.

Related commands : **VIEW**, **COPY**

## Chapter 8

# View Direction Control

One of the most important features of a graphics package is the ease with which a required molecular view can be obtained. CAMERON contains a large number of options which allow the user to control the view direction.

### 8.1 BASIC CONTROL

Five commands are available to control the view direction by applying rotations to the molecule. These rotations are applied cumulatively - rotating about the x axis by 20 degrees then by -10 degrees will result in an overall rotation of 10 degrees *relative to the starting point*.

#### **XROT**

This command is used to apply a rotation of n degrees about the x-axis which lies horizontally across the screen. The syntax is:

**XROT n**

#### **YROT**

#### **ZROT**

These two commands are identical to XROT except for the axis about which the rotation is carried out. The y-axis lies vertically up the screen and the z-axis is perpendicular to the screen.

#### **ROT**

This command is a 'shorthand' for the previous three commands as it allows you to apply three successive rotations by entering only one command. The syntax is :-

**ROT x y z**

Note that the order of rotation is: rotate about x then y then z. The order of rotation is important - YROT 10 XROT 5 ZROT 15 will NOT produce the same result as ROT 5 10 15 as the rotation matrices are non-commutative.

#### **CURSOR**

This command allows the user to control rotation with the cursor keys. Each time a key is struck a rotation of 5 degrees is applied about the relevant axes. After each key stroke the molecule

is rotated and then re-drawn. To speed up this process the molecule is drawn in line style during the CURSOR rotation and is not scaled. The set-up prior to CURSOR is restored once rotation is terminated. The keys used are:

Rotation about the x-axis

Positive	up arrow
Negative	down arrow

Rotation about the y-axis

Positive	left arrow
Negative	right arrow

Rotation about the z-axis

Positive	Delete key (PC)
Negative	End key (PC)

Any other key stroke will terminate the CURSOR control and draw the resultant picture.

## 8.2 ORIENTATION COMMANDS

More specific orientation of the molecule can be achieved by relating the view direction to the position of certain atoms within the picture.

### PLANE

In some cases it is useful to be able to define a PLANE and then set the view direction to be perpendicular to it. The syntax of this command is :-

**PLANE** at1 at2 at3 ...

At least 3 atoms must be used to define the plane (element names will not be accepted). If more than 3 are used then the program calculates the best plane through the atoms and projects onto this.

**ALL** PLANE ALL will generated a 'best view' of the current structure.

### FACE

Alternatively you may want to view onto a particular crystallographic face of the unit cell. The syntax for this command is:

**FACE** h k l

where h k and l are the Miller indices of the FACE in question.

### ALONG

It is possible to define the view direction as that looking down an interatomic direction. The direction is specified by inputting :-

**ALONG** at1 at2

As with PLANE atoms not elements must be used as the arguments. Note that the view direction is calculated as that looking down the at1 - at2 direction ie at1 is the closest of the two. The reverse view will be obtained by requesting ALONG at2 at1. ALONG also has a sub-command associated with it:

#### **AXIS**

It is possible to define the view direction as looking along a particular unit cell axis. The syntax of the command is:

ALONG AXIS x

where x is A, B or C.

#### **VERTICAL**

#### **HORIZONTAL**

These two commands are similar to ALONG except that the at1 - at2 direction is oriented up (VERTICAL) or across (HORIZONTAL) the screen as required.

#### **AXIS**

The AXIS sub command may be applied to the HORIZONTAL and VERTICAL commands. This rotates the diagram around the z axis so that the required axis is HORIZONTAL or VERTICAL as specified. This is particularly useful when generating hard copy output.

#### **BISECT**

This command requires three arguments, at1 at2 and at3. It will generate a view direction looking along the bisector of the at1-at2-at3 angle. The syntax is:

BISECT at1 at2 at3

where at2 is the apex atom.

### **8.3 Other related commands**

#### **IVIEW**

The 'title' command of this group of commands is VIEW. IT IS A VERY IMPORTANT COMMAND. This requires no arguments as causes a picture to be output to the current output device (see SCREEN). The picture is generated according to all of the parameters that are set up prior to its use. For example:

XR0T 10 VIEW

will rotate the molecule by 10 degrees about x and then draw a picture.

VIEW XR0T 10

would draw the picture first and then carry out the rotation. This rotation would not be observed until the next VIEW is entered.

#### **COPY**

COPY is the equivalent command for hardcopy output. COPY must be followed by a filename to which the output will be send. HARDCOPY must have been used prior to this to specify the output type. The filename may be one that has already been used in the current run of CAMERON in which case the user is given an option to append the information.



**MATRIX**

This command allows the user to save the current view matrix.

**FILE** This is followed by a filename. If the file exists (and it is of the correct format) then any matrices contained within it are available and any matrices stored are appended to the file.

**STORE** This stores the current matrix. It is followed by a piece of text (in quotes "text") to act as a description for the matrix. On storage a number is assigned to the matrix which can be used to retrieve it later.

**RETRIEVE** This command obtains a view matrix from the file set up with MATRIX FILE. This command must be followed by the number of the matrix required. Note that these matrices depend on the unit cell parameters and are specific to a particular structure. They may be used for other structures but this will have very strange results. IF this happens an operation such as ALONG AXIS C will recalculate a correct matrix.

**LIST** This outputs a list of the descriptions and numbers of the matrices currently stored.

**STEREO**

This generates output in the form of stereo pairs.

**DEGREE** The angle of rotation between the stereo pairs is given by using the DEGREE command.

**NOSTEREO**

Reverts output back to normal.

## 8.4 PHOTOGRAPHS

**PHOTO****ON**

**OFF** These commands are used to control the PHOTO facility. VIEW displays the user with information about the current scale and displays the mouse cursor once it has finished. PHOTO ON turns off this and waits for a key press before proceeding. This allows the user to obtain a 'clean' screen for photographs to be taken.

Related commands : SCREEN, HARDCOPY

## Chapter 9

# Include And Exclude

The commands INCLUDE and EXCLUDE are very useful as they control which atoms are drawn at any one time. The syntax for both of the commands is very similar as they are effectively the reverse of each other.

### INCLUDE

### EXCLUDE

These commands may be entered on their own and be followed by atom or element names. The syntax is :-

```
nnCLUDE at1 at2 e11 e12
```

Once an atom has been EXCLUDED from the picture it is not drawn and it is not included in any calculations eg PACK or ENCLOSURE.

**ALL** The ALL sub-command is a 'blanket' command that applies the IN/EXCLUDE operation to ALL of the atoms in the current view. Therefore if we want to exclude more atoms than we want to include we can use EXCLUDE ALL to get rid of all of the atoms and then add in those we want by using INCLUDE.

**GROUP** The command DEFGROUP is available to define groups of atoms that are to be referred to as a whole. The IN/EXCLUDE command can be used with GROUP to remove or add all of the atoms in the group in the picture as required.

**FRAGMENT** The user can specify atoms to be IN/EXCLUDED by defining groups of atoms as fragments. A fragment is defined by a single atom, and consists of all the atoms linked to it by the current CONNECTIVITY TABLE. That is, if atoms are excluded from the picture the bonds ARE NOT broken and FRAGMENT will use bonds involving excluded atoms in its calculations. The user can select this atom by typing or by clicking on it with the mouse. ie. EXCLUDE FRAGMENT C1 will exclude C1, any atoms joined to C1, any atoms joined to those atoms etc.

**CELL** It is often useful to see how the atoms being drawn relate to the unit cell. IN/EXCLUDE CELL is used to control the inclusion of the unit cell (in the LINE style) on the picture.

**AREA** Choosing INCLUDE or EXCLUDE AREA in the PC version allows the user to draw a polygonal area with the mouse. The polygon is created by clicking with the left button on the

position required for the vertices. The polygon is closed by clicking close to the initial position. The mouse cursor is changed to a cross during polygon creation and to an arrow when you are close enough to the initial point for closure to occur. Hitting the <return> key or clicking with the right mouse button will abort the operation.

**SELECT** The select option is equivalent to EXCLUDE ALL followed by INCLUDE AREA. The polygonal area is chosen in the same way as described above and all atoms not within that area are excluded.

## 9.1 Further related commands

For pictures with more than one 'type' of constituent atom three more commands will prove to be useful.

### **MASK**

### **UNMASK**

The MASK command has the same syntax as EXCLUDE ie.

```
(UN)MASK at1 at2 e11 e12 ...
```

It is used to remove atoms from current INCLUDE operations - ie atoms that have been MASKed will not be included when using INCLUDE ALL. For example the user could MASK out solvent molecules from a diagram leaving the 'basic' asymmetric unit to be worked on. UNMASK can be used to re-include the required atoms. The sub commands ALL and GROUP are available for use with both MASK and UNMASK.

### **SWITCH**

The SWITCH command causes any atoms that are EXCLUDED from the picture to be INCLUDED and vice versa. This command can be used in conjunction with MASK as any MASKed atoms WILL NOT be included in the picture after a SWITCH operation.

Consider an asymmetric unit containing an anion, a cation and some solvent. The user could MASK out the solvent atoms. EXCLUDE the anion and obtain pictures of the cation atoms. The SWITCH command then allows the user to examine the anion without having to INCLUDE and EXCLUDE atoms, and the solvent atoms remain excluded throughout.

## 9.2 Generation of Dummy atoms

### **DUMMY**

In certain circumstances it may be useful for the user to be able to generate new 'dummy' atoms from the initial atomic coordinates. One example of this is a compound that contains a cyclopentadienyl ring. The bonding in such a system can either be represented as five M - C bonds or as a single bond to the ring's centre. The syntax for the command is:

```
INCLUDE DUMMY d
```

where d is the name of the created atom. This must then be followed by a sub-command which defines the atoms position.

**COORDS** This specifies the atoms coordinates in unit cell fractions. The syntax is:

Tue Jul 5 2005

COORDS x y z

**CENTROID** Alternatively, the atoms position can be defined relative to others already in the molecule. At least TWO atoms must be entered to define new atoms position and it is placed at the CENTROID of their coordinates. The syntax is:

CENTROID at1 at2 at3 ...

*ALL* A further sub-command is available so that the new atom is placed at the centroid of all of the atoms in the molecule. This will prove particularly useful when trying to alter the spacegroup of a structure eg to convert from P 1 to P -1 as the centre of inversion will lie at the newly generated dummy atom whose coordinates are output once they have been calculated.

## 9.3 Example

To create a dummy atom in the centre of a cyclopentadienyl ring you would use:

```
INCLUDE
DUMMY d1
CENTROID C1 C2 C3 C4 C5
```

## Chapter 10

# Drawing Style Control

The user is able to choose from three different drawing styles. These are:

**LINE**

**BALL**

**ELLIPSE**

The basic syntax for style control is:

**XXXX at1 at2 el1 el2**

where XXXX is LINE, BALL or ELLIPSE as required. The user may specify atoms eg C1, C2 and/or elements eg C O to be drawn in the XXXX style. The user may therefore have all three styles present at any one time within the picture. Alternatively the ALL may be used to convert the drawing style of all the atoms present to XXXX. The initial drawing style used is LINE.

The ELLIPSE drawing style is only available if the user has input data that contains information on the temperature factors (isotropic or anisotropic) of the atoms concerned. The ellipse drawing style represents atoms by their thermal ellipsoid. Note that negative eigen values are reset at input to .001\* the next largest. If the information available is U[iso] only, a circle is plotted whose radius is scaled according to its value.

**ALL**

**FRAGMENT**

**GROUP**

These subcommands can be used to specify which atoms are to be affected by the drawing style command.

Several other commands are available in addition to the basic style commands.

### 10.1 LINE commands.

There are no extra commands following LINE.

## 10.2 BALL commands.

**RADII** The user can specify the drawing radius (in angstroms) of a specified atom or element. The syntax is:

```
BALL RADII C 0.8 N1 1.1
```

which will draw all C atoms with a radius of 0.8 angstroms, and the N1 atom with radius of 1.1 angstroms.

**DEFAULT** This sets all the radii to their initial covalent values. The syntax is:

```
BALL RADII DEFAULT.
```

**COVALENT**

**IONIC**

**VANDERWAALS** This sets the radii of the specified atoms/elements to the appropriate values. The syntax is:

```
BALL RADII COVALENT N
```

The further subcommand ALL can be used - BALL RADII IONIC ALL - to set all atoms if required.

**FILL**

**UNFILL** These are header commands and are used to specify whether the circles drawn in the BALL style are to be filled with colour or not. No arguments are required by these commands.

## 10.3 ELLIPSE commands

**TYPE**

ELLIPSE may be followed by the sub command TYPE which enables the user to control the type of ellipse used to represent the atom. The syntax is:

```
ELLIPSE TYPE at/els/all n.
```

Type 'n' can take any of four values:

- 1 - bounding ellipse only.
- 2 - bounding ellipse and principal ellipses.  
This is the default representation.
- 3 - as 2 but excluding the principle axes.
- 4 - as 2 but with shading.

**ALL** ALL can be used followed by a number to set the ellipse type of all atoms in the drawing.

**Note** that any use of ELLIPSE TYPE will set all the atoms referred to ellipse type even if they were previously in LINE or BALL. Hence,

ELLIPSE TYPE C 2 will have all carbon atoms drawn in type 2 form.  
ELLIPSE TYPE ALL 4 will have all atoms drawn in type 4 form

### ***NEGATIVE***

The NEGATIVE sub command requires one argument. If atoms are input with negative temperature factors the atoms temperature factor is reset using the value specified by ELLIPSE NEGATIVE u. The default value of u is 0.01.

### ***PROBABILITY***

This is used to specify the size of the ellipsoid probability envelope displayed. It is followed by a percentage value.

## Chapter 11

# Connectivity Control

It is important that the user has complete control over the bonds displayed in a drawing. The CONNECT group of options is a complicated one but has been designed so as to provide to user with a fully flexible set of commands.

### 11.1 CONNECT

CONNECT is used in one of two ways, either on its own or with a modifying sub-command.

#### CONNECT

The syntax for this command is:

```
CONNECT X Y dmin dmax
```

This will create two lists of atoms (one from X and one from Y which may be atoms, elements or \* as required) and uses these lists when calculating connectivity. dmin and dmax are in angstroms. The distances between all combinations of the atoms in the two lists are determined. If distances any lie within the specified dmin -> dmax range (and the bonds do not already exist) they will be added into the connectivity lists. To connect up Cu1 O bonds for example you could use:

```
CONNECT Cu1 O 0.0 2.0
```

which would create one list - containing Cu1 only - and another list - containing all of the oxygen atoms - and then search for inter-list bonds. In this way, if any O - O bonds exist within the given range they will NOT be found. Note that if you require connections to be calculated between atoms of the same type you use:

```
CONNECT X X dmin dmax.
```

**ALL** CONNECT ALL requires two arguments dmin and dmax. The result is identical to CONNECT above except that ALL of the atoms are included in the calculation.

```
CONNECT ALL 0.0 2.0
```

will draw all bonds that fall within the 0.0 - 2.0 angstrom range.

**DEFAULT** CONNECT DEFAULT requires no arguments. This command does two things:



- 1)      Reset all of the connectivity radii  
         to their initial values
- 2)      Calculate the connectivity according  
         to these radii.

It is effectively a 'start again' option as it removes any changes in bonding that have been introduced with the JOIN, REMOVE etc options.

**RADII** This is a sub-command of CONNECT and comes immediately after it. It requires the following arguments:

CONNECT RADII X r

where X is the name of an atom or element and r is its new connectivity radius. The connectivity of this atom/element is then redetermined.

*COVALENT*

*IONIC*

*VANDERWAALS* These subcommands set the specified atoms to appropriate connectivity radii eg CONNECT RADII IONIC N. The further subcommand ALL can be used to set all atoms.

**HBONDS** This command enables the user to search for hydrogen bonds within the structure. The syntax is :-

CONNECT HBONDS dmin dmax X Y

dmin/dmax is the range for searching (in angstroms), X/Y are atoms/elements to be included in the search. For a H-bond to be valid it must be bonded to one of the atoms in the list X,Y etc AND be within the range of another of the atoms. For example, if we are searching for H-bonds involving oxygen atoms:

CONNECT HBONDS 0.0 2.5 O

will achieve this. There may well be eg C - H - O linkages within the range but only O - H - O ones will be registered. In this way we can filter the search and set a large value for dmax so that all the H-bonds are discovered without obtaining spurious information. The bond style used for H-bonds is dotted.

**INTER**

This option is intended to be used after a PACK or ENCLOSURE operation. This will find connections between atoms which have DIFFERENT packnumbers - ie. between assymetric units or between different GROUPS if PACK or ENCLOSURE GROUP has been used. The bonds are in the DOTTED style when generated. The syntax is:

CONNECT INTER 0 C 0.0 3.0

as for CONNECT itself.

**FULL**

**DOTTED**

The CONNECT command can also be used if bonds already exist to alter the style. There are three options:

```
CONNECT DOTTED at1 at2
CONNECT DOTTED at1 el1
CONNECT DOTTED el1 el2
```

These change the style of the single bond at1-at2, the style of any el1-at1 bonds and the style of any el1-el2 bonds respectively. Note that in the latter case, el1 and el2 may be the same element ie you can make all C-C bonds dotted.

## JOIN

### *DOTTED*

### *FULL*

Another connectivity header command is JOIN. This is a more specific command which is used to make a new bond. The syntax is:

```
JOIN DOTTED at1 at2
JOIN FULL al1 at2
```

This will create a new bond between at1 and at2 of the specified type.

## REMOVE

This is the reverse command to JOIN. It will break the bond between the specified atoms. The syntax is:

```
REMOVE at1 at2
```

## DISCONNECT

This command is the reverse of CONNECT.

```
DISCONNECT C O 0.5 1.5
```

will get rid of any C-O bonds that are OUTSIDE the range 0.5 - 1.5 angstroms.

**ALL** DISCONNECT ALL removes all of the connectivity information.

**GROUP** DISCONNECT GROUP is intended primarily for use with disordered structures although other uses can be envisaged. For example a substituent may be 'flipping' between two sites generating very short distances between atoms in different groups. It is not possible to DISCONNECT these bonds since some of the distances may appear to be 'normal'. To get over this, each disordered part is defined as a group (See DEFGROUP) and DISCONNECT GROUP is used to remove any bonds that exist between them. The syntax is:

```
DISCONNECT GROUP group1 group2
```

## ATOM

This command deletes ALL bonds involving the specified atom. The syntax is:

```
DISCONNECT ATOM at1
```

## 11.2 Miscellaneous CONNECT commands

There are three other commands which either relate to connectivity or to how the bond is represented on the output device.

### **TAPER**

This controls the bond tapering. It has an initial value of 2.0. Increasing this increases the tapering of the bonds. This tapering is useful as it introduces a 3-D effect into the drawing. If tapering is not required then entering:

```
CONNECT TAPER 0.0
```

will achieve this.

### **THICKNESS**

This command controls the thickness (radius) of the bonds in angstroms. It requires one argument, whose default value is 0.02.

```
CONNECT THICKNESS 0.04
```

will double the radius of the bonds as drawn.

### **TOLERANCE**

This sets the tolerance used when determining whether or not a bond exists. The formula used is:

```
If dist < ( C1 + C2 ) * tol then a bond exists
```

where C1 and C2 are the connectivity radii of the atoms in question. The initial value of 'tol' is 1.1 ie the interatomic distance has to be no more than 10% greater than the sum of the two connectivity radii for a bond to be found. The syntax for the command is:

```
CONNECT TOLERANCE n
```

## Chapter 12

# Control Of Colour

Within CAMERON it is possible to control the colour of each individual atom and bond and also the colour of the labels.

### **COLOUR**

This is the header command for colour control - it may be followed by atom/element names if required. The syntax is:

```
COLOUR C BLUE N YELLOW ...
```

The atom/elements are entered in pairs together with the colour name. A list of the current colours can be obtained by using 'INFO COLOUR'. There are two sets of colour names - those for normal colour and those for greyscale either may be used interchangeably as once the colour type is altered the colours are translated accordingly.

```
COLOUR ALL colour
```

Colours all the atoms the given colour

### **GROUP**

This sets the colour of all of the atoms in a given group. The syntax is:

```
COLOUR GROUP groupname colourname
```

**FRAGMENT** COLOUR FRAGMENT n col will set the colour of all atoms in the fragment attached to atom n.

**BACKGROUND** Sets the background colour.

**TEXT** Sets the colour of the title and other annotation text.

**MENUTEXT**

**BUTTON** Selects the colours of the text and buttons in menu mode.

**LABCOLOUR**

This sets the colour of the labels and requires a single argument - the name of the new colour.

**BONDS**

Altering the colour of bonds is a more complicated procedure. It is followed by atoms/elements that define the bond and a colour name. These arguments are therefore entered in threes. For example, to make all carbon carbon bonds colour blue and all CU1 to oxygen bonds colour yellow you would use:

```
COLOUR BONDS C C BLUE CU1 O YELLOW
```

*ALL* There is a further sub-command ALL which changes the colour of ALL of the bonds eg:

```
COLOUR BONDS ALL PINK
```

makes all bonds colour PINK.

### **GROUP**

This sub-command requires arguments in pairs - the group name and then the new bond colour:

```
COLOUR BONDS GROUP g1 LGREY
```

colours all bonds BETWEEN atoms in group g1 (both atoms in a bond must be in the group for its colour to be altered) to colour LGREY.

## 12.1 Example

To have a picture with the following requirements:

```
All C atoms colour BLUE
All N atoms colour PINK
Cu1 colour YELLOW
All Cu1 - O bonds colour LGREY
All bonds between atoms in group g1 colour RED
All other bonds colour GREEN
All labels colour PURPLE
```

the commands would be:

```
COLOUR C BLUE N PINK CU1 YELLOW
BONDS ALL RED
(do this first)
BONDS CU1 O LGREY
GROUP g1 GREEN
LABCOLOUR PURPLE
```

### **BACKGROUND**

The default background colour is WHITE but this can be changed as required by the COLOUR BACKGROUND colour command.

### **DEFAULT**

The default colours for the elements are as defined in the PROP.CMN file -

B - ORANGE  
 Br, Cl, F, I - LGREEN  
 C - GREEN  
 D, H - LGREY  
 N - BLUE  
 O - RED  
 P - PURPLE  
 S - YELLOW  
 SI - DGREY

These colours can be recovered if altered by using COLOUR DEFAULT which returns ALL atoms to their original colours.

### ***NORMAL***

### ***GSCALE***

These sub-commands allow the user to see how the hardcopy greyscale picture will look. COLOUR GSCALE changes the screen colours to the equivalent greyscale colours. The colour names GREYn (n=1,14) can be used to specify colour changes if required as COLOUR N PURPLE makes little sense on a greyscale picture.

BLACK	BLACK
BLUE	GREY1
GREEN	GREY2
ORANGE	GREY3
RED	GREY4
CYAN	GREY5
MAGENTA	GREY6
LGREY	GREY7
GREY	GREY8
LGREEN	GREY9
LBLUE	GREY10
LRED	GREY11
PINK	GREY12
PURPLE	GREY13
YELLOW	GREY14
WHITE	WHITE

## Chapter 13

# Atom Labelling

### **LABEL**

#### **NOLABEL**

The LABEL and NOLABEL commands control the atom labelling. They are set up in an identical way to INCLUDE and EXCLUDE and the syntax is identical:

```
LABEL C1 C2 ONOLABEL C H1 LABEL ALL NOLABEL ALL LABEL GROUP g1 ....
```

are all valid. Note that atoms will not be labelled if they are not included in the picture.

#### **MOUSE**

All atoms will be labelled if the LABEL command has been used for them. The label positions can be altered using the mouse if required. Note that label positions are recalculated if atoms are included or excluded or a change in the view direction has occurred. Therefore, it is advisable that the 'final' view is obtained before labels are positioned with the mouse.

Mouse labelling is controlled as follows. The user clicks on a position on the screen (once the message Mouse Labelling activated has been seen). If this position is over a label then the label is replaced by a box of the same size as the label. A red cross is drawn over the atom that the label refers to aid identification. The mouse is then used to position the TOP LEFT HAND CORNER of the label by a second click. Alternatively, hitting the N key (for No label) will remove the label altogether.

If the mouse is clicked on an atom which is NOT labelled then a label will appear at the atom centre, this label can then be moved as described above.

If at any time the user wishes to view the current picture without releasing the mouse, this can be done by hitting the 'V' key.

#### **GENERATED**

#### **INITIAL**

These commands relate to the pack number display while labelling. Atoms are assigned a pack number after PACK or ENCLOSURE. This number is then displayed as eg N1.5. These numbers are used to refer to atoms and elements as required. INITIAL (the default) will just display N1 while GENERATED shows N1.5.

**CELL** Controls the cell labelling.

***FRAGMENT*** Allows the user to set labels for a given fragment.

### **FONT**

This sets the point size of the font to be used in hardcopy output. It requires one argument.

***DEFAULT*** This resets the point size to its default value.

### **TEXT**

***POSITION*** This allows the user to annotate a picture. The syntax is:

`TEXT "text string" POSITION x y`

The text must be in quotes. x and y are the position of the text in percentages from the top left hand corner of the diagram. After the text is processed it is assigned a number.

***NUMBER*** The NUMBER command can be used to move a text item after it has been created by using `TEXT NUMBER n POSITION x y`.



## Chapter 14

# Other Picture Controlling Commands

There are a few other commands in CAMERON which deal with control of the type of picture being used.

### **MAXIMISE**

This command is related to the VIEW commands PLANE, ALONG, VERTICAL etc. In these commands the view direction does not define all of the degrees of freedom of the molecule. It is possible to rotate the molecule further so that as much of it as possible is shown on the screen.

### ***ON***

### ***OFF***

These two commands switch maximisation calculations ON and OFF. These calculations are slow - especially for large molecules so they are included as an option.

### **SCALE**

### **DFIX**

### **DUNFIX**

These two commands allow the user to control the calculation of picture scale. SCALE FIX sets the scale to its current value and does not recalculate it, irrespective of changes in the number and position of atoms in the picture. SCALE UNFIX reverts to scale calculations.

### ***SET***

SCALE SET is followed by a number, the value of the scale to be used for all subsequent pictures. This can be altered with another SCALE SET command or by issuing SCALE UNFIX.

## Chapter 15

# Symmetry Input

### 15.1 Input of symmetry operators

CAMERON is a crystallographically oriented program and hence many of its functions require the use of symmetry operators. There are two methods of inputting the symmetry operators, inputting the Spacegroup symbol and inputting the individual operators themselves.

#### **SPACEGROUP**

The SPACEGROUP command is followed by the symbol, which must contain all UPPER CASE letters. If the symbol is in a non-standard setting then the full symbol must be entered. For example P 21 will be interpreted as P 1 21 1 which is the standard setting. If P 21 1 1 is required then the "1's" must be entered to force the choice of unique axis.

The syntax of the command is:

```
SPACEGROUP X X X X
```

where X are "fields" of the symbol, with spaces between fields ie entered as P 21 and NOT P21. The program will then output the operators that have been calculated for the symbol. Note that if a centre of inversion is present the inverted operators will NOT be shown. Also the operators generated from the centring vectors - eg 1/2 1/2 1/2 for body centring - will NOT be shown. The complete list of operators used for packing etc can be found by entering INFO SYMMETRY.

#### **SYMMETRY**

The input of symmetry operators can be done "by hand" if required. There are several steps and sub-commands available to do this.

#### **OPERATORS**

This sub-command MUST be entered even if only the x y z operator is to be included. The syntax is :-

```
SYMMETRY OPERATORS x y z -x y+1/2 -z ...
```

with the operators being entered with the translational part in fractional form ( 1/2 , 1/3 , 1/4 , 2/3 , 1/6 , 5/6 are recognised ) which must come after the x/y/z as required. The fractional part must be linked to the x/y/z part with a + or - sign. Note that NO spaces are allowed within each part of the operator. This is so that ambiguities cannot arise.

#### **CENTRE**

This command is used to introduce a centre of inversion into the symmetry information it must be followed by a centring letter (P,A,B,C,I,F,R) or the command VECTORS.

### ***NOCENTRE***

This command is used to specify that there is no centre of inversion - the centring vectors are the specified by letter or by using VECTORS.

### ***VECTORS***

This command enables the user to introduce centring vectors into the symmetry operators eg for body centring use :-

```
VECTORS 0 0 0 1/2 1/2 1/2
```

### ***USE***

***NOUSE*** These commands allow the user to omit certain symmetry operators from the packing calculations. The command is followed by the operator numbers (found using INFO SYMMETRY) of the operators needed.

***ALL*** This may be used to USE/NOUSE all of the operators. Note that using NOUSE ALL without following it with USE n will result in no atoms being generated after a pack operation!

## 15.2 Example of Space Group Input

In order to end up with the Spacegroup F m m m we require :-

```
SPACEGROUP F M M M
```

or

```
SYMMETRY OPERATORS
```

```
X Y Z  -X -Y Z  -X Y -Z  X -Y -Z
```

```
CENTRE
```

```
VECTORS
```

```
0 0 0
```

```
0 1/2 1/2
```

```
1/2 1/2 0
```

```
1/2 0 1/2
```

## 15.3 Other symmetry related operations

### **SETUNIT**

This command causes the atoms that are to be included in the current picture to be set as the asymmetric unit. Their data then replaces that of the initially input atoms and they are used as the asymmetric unit from now on. This is useful if, for example only half of a molecule is present in the asymmetric unit. The other half can be generated with ADD and then SETUNIT is used to treat all of the atoms as the basic "building block". SETUNIT is a DANGEROUS command in that it cannot be undone - UNPACK will NOT reverse the operation. Therefore the user is prompted for confirmation before the command is executed. The syntax is simply SETUNIT.

**ENANTIO**

This command inverts the hand of the atoms in the asymmetric unit. Note that the operator applied is :-

```
-1  0  0
 0 -1  0
 0  0 -1
```

and no spacegroup changes that may be required are made.

**SPECIAL****ON**

**OFF** Controls the special position calculations that occur during packing to eliminate duplicate atoms. By default this is ON.

## Chapter 16

# Crystal Packing Commands

There are two methods of crystal packing available, **PACK** and **ENCLOSURE**. Both use ALL of the symmetry operators in the **Spacegroup** to generate all of the atoms that lie within a user-defined volume. The commands **ADD** and **MOVE** are available if single symmetry operators are required. The difference between the two commands lies in the method of volume definition.

### 16.1 PACK

The **PACK** command allows the user to define a volume to be filled relative to the unit cell. One of two sub-commands is required to define this volume.

#### **CELL**

**PACK CELL** will cause the program to generate all the atoms that lie within the unit cell. This is the default option if no range for packing is input.

#### **WINDOW**

**PACK WINDOW** allows the user to define the volume in terms of the unit cell axes. It is followed by three pairs of numbers.

```
PACK WINDOW xmin xmax ymin ymax zmin zmax
```

The values of **xmin** etc are relative to the unit cell origin. Therefore to define a volume of which contained all of the **x** axis, all of the **y** axis and the first half of the **z** axis we would use:

```
PACK WINDOW 0.0 1.0 0.0 1.0 0.0 0.5
```

If more than one unit cell is required negative numbers and numbers exceeding one may be used.

One more sub-command may be entered after the **WINDOW** or **CELL** commands. If this command is omitted then the option chosen in the last **PACK** command will be used. If this is the first time **PACK** is used then **CUT** is the default option. Three sub-commands are available:

**CUT** **CUT** will generate all the atoms that lie within the defined volume. ie. the generation is "cut" at the boundary.

**COMPLETE** **COMPLETE** is most useful for molecular crystallographers. It will generate all the asymmetric units that have ANY ATOMS lying within the defined volume.

**CENTROID** CENTROID is similar to COMPLETE except that it calculates the centroid of the asymmetric unit (as though all of the atoms have equal weight) and includes only those which have their centroid within the defined volume. This is particularly useful for molecular crystallographers as it creates a picture without the "odd atoms" at the edge of the unit cell.

## 16.2 Dealing with connectivity

There are three other qualifying commands that apply to PACK:

### **INTRA**

### **INTER**

### **KEEP**

These deal with the treatment of connectivity calculations once the PACK has been carried out.

### **INTRA**

This qualifier means that the connectivity will be calculated within each newly generated asymmetric unit only. Any changes to the bonding - eg with JOIN, REMOVE, CONNECT, COLOUR BONDS - will be undone. This is the fastest option.

### **INTER**

In this case connectivity is calculated once all of the atoms have been generated - therefore if any intra-unit bonds exist they will be found.

### **KEEP**

This is the default option. Connectivity is copied from the unit used to do the packing - this includes colour and style changes if any - into the bond info of the new atoms. This is done for each asymmetric unit as it is generated and is slower than INTRA as it requires more comparisons to be carried out.

## 16.3 EXAMPLES

Therefore, to create a picture containing all of the complete molecules within a cube of side equal to 2 unit cells we need :-

```
PACK
WINDOW 0.0 2.0 0.0 2.0 0.0 2.0
COMPLETE
```

Or alternatively,

```
PACK
WINDOW -1.0 1.0 -1.0 1.0 -1.0 1.0
COMPLETE
```

## 16.4 ENCLOSURE

The enclosure command is more flexible than PACK as it enables the user to choose the "origin" for the atom generation. The first task is to specify this origin which is either a point in the unit cell or an atom.

## 16.5 Choice of enclosure "origin"

### ATOM

If we wish to generate the atoms around Cu1 in order to examine the coordination environment for example we can use:

```
ENCLOSURE ATOM CU1
```

### POINT

Alternatively we can choose the centre of the unit cell:

```
ENCLOSURE POINT 0.5 0.5 0.5
```

The POINT sub-command can be used if an atom does not lie at the point in question eg if we are examining a "hole" within a structure.

## 16.6 Type of volume to be used

There are three different ways of defining the volume of enclosure:

### SPHERE

This will generate a sphere of enclosure about the origin. The syntax is:

```
SPHERE r
```

where r is the radius in angstroms of the sphere.

### ANORTHIC

This is used to generate an ANORTHIC box ie a box whose sides are parallel to the unit cell axes. As this box is directly related to the unit cell its dimensions are given in fractional coordinates. The syntax is:

```
ANORTHIC -x +x -y +y -z +z
```

Therefore, to generate an anorthic box with sides extending one quarter of a unit cell in all six directions from the defined origin we use:

```
ANORTHIC 0.25 0.25 0.25 0.25 0.25 0.25
```

### ORTHOGONAL

This is used to generate a box whose sides are perpendicular to each other. The z axis lies along the current view direction and the x and y axes lie across and vertically up the screen respectively. The dimensions of this box is defined in a similar way to the ANORTHIC box except that they are given in angstroms. For example, to generate a box that is 4.0 angstroms wide in x, 1 in y and 0.5 in z we would use:

```
ORTHOGONAL 2.0 2.0 1.0 1.0 0.25 0.25
```

It is important to note that this volume is related to the CURRENT view direction. The VIEW ALONG AXIS command can be used to orient the picture prior to carrying out this command if required.

**CUT**

**COMPLETE**

**CENTROID** As with PACK a further sub-command can be used if desired to define the type of boundary handling used. CUT, COMPLETE and CENTROID have the same meanings as described above for PACK.

**INTRA**

**INTER**

**KEEP** These sub commands have an identical meaning to those described for PACK above.

## 16.7 EXAMPLE

To generate all the atoms that lie within a sphere of radius 5.0 angstroms about a CU1 atom we use:

```
ENCLOSURE
ATOM CU1
SPHERE 5.0
CUT
```

To generate all the asymmetric units that have any atoms within a box of side 0.5 units around the unit cell centre we use:-

```
ENCLOSURE
POINT 0.5 0.5 0.5
ANORTHIC 0.25 0.25 0.25 0.25 0.25 0.25
COMPLETE
```

And to generate all the atoms that lie inside a box centred on the point 0.25 0.25 0.25 and of sides x=1.0 y=2.0 and z=3.0 angstroms :-

```
ENCLOSURE
POINT 0.25 0.25 0.25
ORTHOGONAL 0.5 0.5 1.0 1.0 1.5 1.5
CUT
```

## 16.8 PACKING MORE COMPLICATED STRUCTURES

The PACK and ENCLOSURE commands always work on the initial data- unless a SETUNIT command has been issued. In some cases however, it is more useful for the user to be able to deal with certain sections of the structure separately. This is most likely to occur where there is more than one distinct unit in the asymmetric unit. This is dealt with by the command GROUP.

**GROUP**

GROUP can be used directly after both PACK and ENCLOSURE. The syntax is:-



```
PACK GROUP groupname CELL ...  
or  
PACK GROUP groupname WINDOW ...  
etc
```

This causes the groups to be packed individually, the CENTROID, COMPLETE and CUT commands are applied to the group and not to the asymmetric unit as a whole. Packs are cumulative, unless a PACK/ENCLOSURE is done without the GROUP sub command in which case the previously generated atoms are overwritten. Groups are defined with the command DEFGROUP.

If required, more than one groupname can be packed at once - they are all treated separately. All groups can be packed in turn if PACK GROUP \* is used.

## 16.9 UNPACK

This command causes all atoms generated via PACK or ENCLOSURE to be removed from the calculations, drawings etc. It has no sub-commands. It also works with ADD and MOVE generated data.

## Chapter 17

# Add And Move - Further Symmetry Related Commands

The PACK and ENCLOSURE commands already detailed allow the user to apply all of the symmetry operators in the spacegroup to the initial set of atoms in order to get a fully 'packed' result. In some cases however, the user may wish to apply only one symmetry operator or to apply ones that are not present in the spacegroup. The ADD and MOVE commands allow this.

### 17.1 ADD

The ADD command allows the user complete control over the symmetry operator used to generate new atoms. The first task of the user is to generate a list of those atoms to be used in the symmetry generation later. One of the following sub-commands is required :-

#### **ATOMS**

The names of atoms to be included in the pack list are specified here. Element names can also be used if required.

#### **ALL**

ALL refers to the atoms that are in the current list. If any atoms have been generated by previous PACK, ENCLOSURE or ADD commands then these will all go into the list.

#### **INITIAL**

ADD INITIAL means that the only atoms to go into the ADD list are those that were initially input.

#### **GROUP**

This is followed by a group name. The group must have been previously defined by the command DEFGROUP.

Once the ADD list has been created the user must then supply the symmetry operators which will act on the atoms in this list to generate the new atoms. The symmetry input is in two parts.

**OPERATOR** The symmetry operator may be input in decimal or fractional form eg

```
x y+1/2 z
```

```
-0.333-x -y -z
ETC
```

Decimal translations may come before or after the axis symbol. The fractions 1/2, 1/3, 1/4, 2/3, 1/6 and 5/6 are accepted by the program, but **must** appear **after** the x/y/z character. There must be spaces between the three parts but NO SPACES within the operator ie

```
X + 1/2 -Y Z
```

will produce an error as it is not possible to tell whether you mean  $X+1/2, -Y, Z$  or  $X, +1/2-Y, Z$ . This strict input syntax is used to eliminate any ambiguities.

**TRANS** Translations can also be applied if required. The translations are applied in unit cell fractions. The syntax is :-

```
TRANS x y z
```

## 17.2 EXAMPLES

To generate an atom at  $x+1/2, y, z$  from an atom at  $x, y, z$  we can use

```
ADD
ATOMS C1
OPERATOR X+1/2 Y Z
```

or we could use

```
ADD
ATOMS C1
TRANS 0.5 0 0
```

The OPERATOR and TRANS commands can be used together if required. We can apply a symmetry operator followed by a translation. This reduces the errors that may occur when trying to combine the two things into one symmetry operator.

The use of INITIAL versus ALL is illustrated below. Start with :-

```
ADD
ALL
TRANS 1 0 0
```

which gives us a molecule at  $x, y, z$  and another at  $x+1, y, z$ . Follow this with :-

```
ADD
ALL
TRANS 0 1 0
```

and we get four molecules,  $x, y, z$ ,  $x+1, y, z$ ,  $x+1, y+1, z$  and  $x, y+1, z$ . Following it with :-

```
ADD
INITIAL
TRANS 0 1 0
```

Gives us three molecules,  $x, y, z$ ,  $x+1, y, z$  and  $x, y+1, z$ .

## 17.3 MOVE

The syntax for this command is identical to that for ADD. Therefore the commands available are :-

**ATOMS**

**ALL**

**GROUP**

**INITIAL**

Which must be followed one (or both) of :-

***OPERATOR***

***TRANS***

The MOVE command applies a symmetry operator and/or a translation to all of the atoms held in the list defined by the ATOMS/ALL/GROUP/INITIAL commands. Unlike ADD therefore, the same number of atoms are present at the beginning and end of the operation.

## Chapter 18

# Distance And Angle Calculations

CAMERON allows the user to calculate distances , angles and torsion angles.

### **DISTANCE**

In order to perform a distance calculation two atoms lists must be generated. The first list is used as a 'starting atom' and the second list is for 'target atoms'. Distances will be calculated between atoms in separate lists BUT NOT within the lists themselves. The list may be generated in two ways:-

**DISTANCE N O**

generates two lists - both of which can all the N and all the O atoms. In this case N-O, N-N and O-O distances will be found.

### **FROM**

**TO** These subcommands allow to specify different starting and target atoms.

**DISTANCE FROM N TO O**

will only calculate N-O distances. Note that the distance arguments may be either atoms or elements as required.

**RANGE** This sets the minimum and maximum ranges for displayed distances. The syntax is :-

**DISTANCE RANGE dmin dmax**

distances are given in angstroms. If only TWO atoms are present in the atom list then the distance will be outputted irrespective of the range. However, as the full calculation will be carried out first there may be a time delay while the calculation proceeds.

## 18.1 Method of Calculation

The distances output make use of the symmetry operators in order to find distances within the given range. The starting atom coordinates are NOT ALTERED but those of the target atoms are. The symmetry operators (and any suitable translations) are used to move the target atoms around. The output produced is:-

```
N1_0      02      2.323
Operator x y z Translations 0 0 1
N1_0      02_4     1.114
Operator -x y+1/2 -z Translations 0 0 -1
```

These show the N1 to O2 distances. The first distance relates to an O2 atom which does not currently exist. The operator and translations shown can be used with ADD to generate the atom if required. The second distance is to an atom (O2\_4) which does exist. ie. if no pack number is given for the second atom it is not present in the current list.

## ANGLE

## TORSION

These commands are used to find angles and torsion angles between atoms that are in the current list. They are entered in sets of three (or four) as required.

## Chapter 19

# Information On Data Held Within The Program

CAMERON holds a number of pieces of information while it is running and in some cases it is useful for the user to be able to access this information.

### **INFORMATION**

This command is followed by a sub-command which specifies the type of information wanted.

**CELL** Outputs the unit cell parameters.

**ATOMS** Outputs the names of atoms stored within the program. It produces two lists - one of atoms currently included and one of atoms currently excluded from the picture.

**COLOUR** This outputs the colour names that are available.

**SYMMETRY** This command outputs the symmetry operators currently stored.

**GROUP** Outputs a list of the currently defined groups and their members.

**PACKNUMBERS** Outputs the symmetry operator and translation associated with a given packnumber. The syntax is INFO PACKNUMBER n1 n2 n3.

## Chapter 20

# Group Definitions

For complicated structures it is sometimes useful to define GROUPS of atoms which can be referred to as a whole later. EXCLUDE, INCLUDE, COLOUR BONDS etc can all be used with the GROUP sub-command.

### **DEFGROUP**

This is the main header command and is followed by the name of the group. Note that it is not possible to have group names that begin with GROUP itself - g1, g2 are valid names but group1, group2 are not. Up to twelve characters are allowed to define the group name.

### **ATOMS**

This is followed by a list of atoms/elements to be included in the group.

### **GROUP**

It is possible to have an atom as a member of up to three groups at once. You can therefore add groups into other groups (see Example).

### **FRAGMENT**

The user can include atoms in a group by defining a fragment. The fragment definition requires a single atom. Any atoms joined to it, and any atoms to those etc are made part of the group.

### **DELETE**

You can also remove atoms from groups if required.

## 20.1 Example

Consider an molecule that contains a tri-phenyl phosphine. A use of the DEFGROUP command would be :-

```
DEFGROUP PHENYL1
ATOMS C1 C2 C3 C4 C5 C6
DEFGROUP PHENYL2
ATOMS C11 C12 C13 C14 C15 C16
DEFGROUP PHENYL3
ATOMS C21 C22 C23 C24 C35 C36
DEFGROUP PPH3
ATOMS P
GROUP PHENYL1 PHENYL2 PHENYL3
```



If you then realise that there are two phosphorus atoms in the molecule P1 and P2 you can use :-

```
DEFGROUP PPH3 DELETE P2
```

to remove P2 as it is not a member of the tri-phenyl phosphine group.

## Chapter 21

# Miscellaneous Commands

### **RESET**

This command allows the user to begin again without having to exit and reload CAMERON. All current flags will be set to their initial states. Note that if a CAMERON.INI file is present it will not be obeyed after aRESET command. The screen will be set to VGA automatically (as on startup) if a CAMERON.LJP file is present.

### **LOG**

The LOG command is used to generate a log of the CAMERON session, this may be OBEYed later if required. LOG must be followed by a filename. If a file is already in use for alog file then the log information will be transferred to the new file.

### **EDIT**

#### ***ON***

***OFF*** This sets the edit option which is used when an error is found in an input line. If edit is on (the default) then the user is prompted as described in Chapter 1. If not the computer will bleep and put up the message "Error: Automatic Abandon", ignore any unexecuted commands and wait for a new input line.

### **MENUS**

#### ***ON***

***OFF*** The user is able to control the program via push button menus at the side of the screen (PC only). These can be turned on or off as required. When the menus are active the user can input commands as normal by typing - they will appear in a box at the bottom of the screen.

### **ISSUE**

This allows the user to execute a single system command. This command should be enclosed in quotes if it is longer than one word or if it corresponds to a CAMERON command.

### **PRINT**

This allows the user to print a hardcopy file. The file will be closed if it is currently attached to CAMERON via a COPY command. **NOTE** that the print job will be very slow - it may be quicker to close down CAMERON first.

## Chapter 22

# How To Stop The Program

### **END**

The command END will cause the program to finish.

## Chapter 23

# Menu Definition File

The labels on the menu buttons and the actions they invoke are in the file `cameron\button.cam`.

The first line gives the number of columns and the number of lines in the menu. After this, adjacent pairs of lines give the text to appear on the button, and the CAMERON commands to be obeyed by the button. The pairs of lines are given first for column 1, then column 2 etc.

## Chapter 24

# Some Useful Ideas

To produce black and white illustrations for papers,try

`colour c blackcolour h blackcolour n blackcolour o blackcolour bond all whiteball all unfillball h fillcolor`

To hack up a packing diagram, cutting out most and just keeping a bit,try (all on one line)

`EXCLUDE ALL INCLUDE AREA`

Then use the mouse to draw round the bit you want to keep.