

# Mockingbird Manual

by John Maxwell

January 1982

**Abstract:** Mockingbird is a Dorado-based, display-oriented, music notation editor. It provides facilities to record, edit, play and print music. Mockingbird is especially focussed on helping the composer capture his ideas.

**XEROX**

PALO ALTO RESEARCH CENTER  
3333 Coyote Hill Road  
Palo Alto, California 94304

## TABLE OF CONTENTS

<b>1.0 INTRODUCTION</b>	4
1.1 What is Mockingbird?	4
1.2 Conventions Used in This Manual	4
1.3 Acknowledgements	4
<b>2.0 GENERAL OVERVIEW</b>	4
2.1 The Dorado	4
2.2 The Synthesizer	5
2.3 The Screen	5
2.4 The Pop-Up Menu	5
2.5 Commands	6
<b>3.0 FILING IN AND OUT</b>	6
3.1 Automatic Backup	6
<b>4.0 THE VIEW</b>	6
4.1 Scrolling and Thumbing	7
4.2 Scaling	7
4.3 Voices	7
4.4 Carry	7
<b>5.0 SELECTION</b>	8
5.1 Note Selection	8
5.2 Section Selection	8
<b>6.0 NOTES AND RESTS</b>	9
6.1 Movement	9
6.2 Insertion	9
<b>7.0 NOTE ATTRIBUTES</b>	10
7.1 Grace	10
7.2 Rest	10
7.3 Voice	10
7.4 Staff	10
7.5 Stem Direction	10
7.6 Duration	10
7.7 Spelling	10
7.8 Embellishments	10
<b>8.0 MEASURE LINES</b>	10
8.1 Insertion	11
8.2 Movement	11
<b>9.0 NOTE STRUCTURES</b>	11
9.1 Syncs	11
9.2 Chords	12
9.3 Ties	12
9.4 Beams and N-tuplets	12
<b>10.0 SCORE ATTRIBUTES</b>	14
10.1 Keys	14
10.2 Time Signatures	14
10.3 Metrenomes	14
10.4 Sheet Styles	14
10.5 Clef Switches	15
10.6 Octava	15

<b>11.0 SCORE MANIPULATIONS</b>	15
11.1 Delete	15
11.2 Replace	15
11.3 Transpose	16
<b>12.0 LOGICAL MODE</b>	16
12.1 Justification	16
12.2 Checking Voices	17
12.3 Playback and Record	17
12.4 Printing	18
12.5 Automatic Features	18
<b>13.0 PIANOROLL MODE</b>	19
13.1 Special Looks	19
13.2 Recording	19
13.3 Playing Against Oneself	19
13.4 Beating In Measures	20
13.5 Automatic Syncing	20
13.6 Automatic Durations	20
13.7 Automatic Beaming	20
13.8 Overwriting the Pianoroll	20
13.9 Another Way to Enter Notes	21

## 1.0 INTRODUCTION

### 1.1 What is Mockingbird?

Mockingbird is a *composer's amanuensis*, a computer program designed to aid the composer with the capture, editing, and printing of musical ideas. The basic goal of Mockingbird is not to replace the composer, but to aid him by speeding up the process of notating music. Mockingbird is not a publisher's aid, although it does print music; nor is it a performer's aid, although it can play; it is strictly focused of the needs of a composer.

Mockingbird is an *interactive music notation editor*. It knows nothing about the rhythmic, harmonic, or melodic aspects of music except inasmuch as they are represented in common music notation. Its knowledge of music *notation*, however, is quite extensive and sophisticated. Mockingbird is highly interactive, presenting the illusion that the user can reach in and move things around as he desires. This illusion is supported by the fact that the detail of the score is always shown exactly as it might be printed.

Mockingbird runs on an experimental machine called a Dorado, which is a high-speed "personal" computer developed at Xerox PARC. All of the software is written in Mesa, which is an experimental language also developed at PARC. Mockingbird may optionally use a Yamaha CP-30 synthesizer. The user does not have to have the synthesizer to use the editor unless he wishes to record or play back music. Mockingbird alone will allow the user to enter notes and manipulate the score. In fact, we suggest that novice users use Mockingbird by itself before moving on to the synthesizer.

### 1.2 Conventions Used in This Manual

There are a few conventions that are followed in the remainder of the text about how commands and comments are printed. Commands issued by the user are always typed in **bold**. If only one character of the command is actually typed, it will be underlined. Command parameters are named with *italics*. The various mouse buttons, CR, shift and control keys are printed in THIS FONT.

Irrelevant details are printed in this font. Of course, they are always read.

### 1.3 Acknowledgements

Mockingbird is the result of the work of John Maxwell and Severo Ornstein at Xerox Palo Alto Research Center. They were supported in their work by Will Crowther, Gene McDaniel, Doug Wyatt, Mike Overton, and all of the Computer Science Laboratory. Special thanks should be given to Martin Kay for the term *amanuensis*.

## 2.0 GENERAL OVERVIEW

### 2.1 The Dorado

The Dorado is an Alto-like machine with a keyboard, a display, a cursor, and a *mouse*. The mouse is a small pointing device with three buttons that allows the user to move the cursor around on the screen. We shall call the buttons on the mouse (from left to right): RED, YELLOW, and BLUE. The buttons are all black, but this is the standard naming convention. For convenience, we shall speak of *pressing* or *clicking* the mouse buttons. By this we merely mean pushing a button down and holding it there, or quickly tapping a button. In general, RED is used for selections, YELLOW is used for deselections and deletions, and BLUE is used for movements and insertions.

The keyboard has all of the keys of a standard typewriter keyboard. In addition, it has a CONTROL key just above the SHIFT key which is used in combination with the alphabetic keys, just as the SHIFT key is. The CONTROL key may also be used simultaneously with the SHIFT key. This only happens once in Mockingbird. Thus CONTROL SHIFT b is one character, just like SHIFT b (=B) and CONTROL b (=^b). We shall never refer to the synthesizer keyboard as "the keyboard", rather we shall always be careful to refer to it as "the synthesizer" or "the synthesizer's keyboard".

The display is a 606x808 pixel raster-scan display. At this resolution we can display both tilted beams and slurs without annoying the user.

## 2.2 The Synthesizer

The Yamaha CP-30 synthesizer is a 76-key electronic synthesizer which has been modified to interface with a Dorado. The Dorado rapidly scans the keyboard looking for changes in the positions of the keys corresponding to notes struck by the pianist. It can also simulate the striking of notes, producing sound. The interface is carefully arranged so Mockingbird can both play and listen at the same time. This allows the pianist to add new material by playing along with an existing score.

## 2.3 The Screen

Mockingbird's screen is divided into three parts: a command window at the very top, a black bar just under it for the file name, and a window for the score which takes up the bulk of the screen. The command window has a menu of five commands: FileIn, Playback, Quit, Record, and FileOut. The user may invoke any of these commands by moving the cursor over the command until it changes into a bull's-eye, and then clicking any of the buttons (Different buttons *may* have different meanings to a command). The command name will immediately invert itself, and remain inverted until execution is completed. All of the commands will run to completion without user control except Playback and Record, which can be terminated the same way they were invoked.

The black bar just under the command window is used to hold the current file name. This name is for communicating with the file system. To change the file name, move the cursor over the section just under "FileIn" until it changes into a bull's-eye and then click any of the mouse buttons. Any current file name disappears, and Mockingbird waits for the user to type in a new file name. He should type in the name of his choice, using BS (backspace) and DEL (delete) as he desires, terminating with ESC, "." or CR. Mockingbird will automatically append ".music" to the filename.

At the left of the score window is a sensitive area that we call the *scroll bar*. If the user moves the cursor into this area, he will notice that the cursor changes shape. The scroll bar is used for scrolling and thumbing through the score (discussed later).

## 2.4 The Pop-Up Menu

The user can obtain icons from a menu by holding CONTROL and BLUE down together. This will cause a special menu to appear under the cursor. To select a particular icon, move the cursor over it until the cursor takes on the shape of the icon, and then release CONTROL and BLUE. The cursor will keep that icon until the user goes to the menu again. To get rid of an icon, go back to the menu and move the cursor so that it is anywhere outside of the menu. (Moving left is easiest.)

When the cursor has an icon in it, Mockingbird is in a special mode that changes the meaning of BLUE. (CONTROL and SHIFT BLUE are not affected). Normally, BLUE is used for moving non-note items around. When the cursor has an icon in it, BLUE will acquire a new meaning that is dependent on the icon. Consequently, it is not possible to move non-note objects as long as the

icon remains.

## 2.5 Commands

Most of the user actions within Mockingbird have the following paradigm: the user makes a selection and then issues a series of commands that apply to that selection. All of the commands that fit this paradigm are keyboard commands where the user types a single character for the command plus occasional parameters. The characters typed are usually suggestive of the command, i.e. **g** for **g**race and **b** for **b**eam. We will take pains to underline the actual character typed by the user.

It is frequently the case that a SHIFTeD command will undo the action performed by the unadulterated command. Thus, **SHIFT g** *un-graces* and **SHIFT b** *de-beams* the selected notes. Exceptions to this rule will be noted as they are encountered.

There are a class of commands called "look" commands which change how the score is viewed. These commands are like the "look" commands in Bravo. They take *two* characters ( **l**ook plus some aspect of the view) plus occasional parameters. Typical **l**ook commands are **l**ook **j**ustified *n* and **l**ook **h**ardcopy.

The user can abort a command anytime before he finishes typing it by typing **DEL**. Once a command has been issued, there is no way to stop it. *Nor is there any way to undo commands.* If the user makes a mistake he will have to fix it himself or go back to the last version filed out.

## 3.0 FILING IN AND OUT

The user can move a file to or from the file system by using the "FileIn" and "FileOut" commands at the top of the screen. These two commands take the filename in the black bar as their parameter. The user can change the filename by clicking over with the cursor and typing a new name.

Mockingbird always checks its data structure for consistency before filing out and will abort the command if the data structure is in a funny state. This guarantees the integrity of the files that make it out of Mockingbird. If Mockingbird refuses to file out a particular file, it can sometimes be salvaged by a wizard if the user has not quit from Mockingbird.

### 3.1 Automatic Backup

Every so often Mockingbird will pause to file out the current file onto the file "backup.music". If the system crashes or the user accidentally mangles his file, he can always start over again with the last backup file that Mockingbird produced.

## 4.0 THE VIEW

There is an subtle difference in Mockingbird between the view on a score and the score itself. The score consists of the notes, chords, signatures, etc. which make up the substance of the musical idea. The view is the choice of page on the screen, the scale the music appears at, and other aspects which affect how the user sees the score. Generally, the view is changed through **l**ook commands and the score is changed through non-look commands. This section will only cover the more important of the view aspects; other view aspects will be discussed as appropriate.

#### 4.1 Scrolling and Thumbing

Mockingbird uses "Bravo" style scrolling and thumbing. To the left of the score is a scroll bar. If the user moves the cursor into the scroll bar without pressing any of the mouse buttons, the cursor will change into a double arrow. Pressing RED causes the cursor to change into an up arrow.

Releasing RED while still in the scroll bar causes the nearest line of the score to be scrolled to the top of the page. Unless it is already the top of the page, in which case the next line is scrolled to the top.

Releasing RED outside of the scroll bar aborts the command. Similarly, clicking BLUE will scroll the score down. Clicking YELLOW lets the user "thumb" to an arbitrary point in the score, much like a reader may thumb to any page of a book. The location of the cursor in the scroll bar determines where the user will end up in the score. If the user thumbs from two-thirds of the way down the scroll bar, then he will end up two-thirds of the way into the score.

#### 4.2 Scaling

Like Bravo, Mockingbird has a hardcopy mode for viewing the score as it will appear when printed.

To enter this mode, type **look hardcopy**. To return to the standard mode, type **look SHIFT hardcopy**. Mockingbird also has a very low resolution mode called the overview that allows the user to work on large sections of the score at a time. The user may enter and leave this mode with **look overview** and **look SHIFT overview** respectively. The user can also jump from the overview to a particular point in the score by pointing at it and clicking YELLOW.

The user should be warned that any change in scale will change the positions of the line breaks in the score. This means that although a score is properly justified in one scale it probably won't be justified in another. However, if the user returns to the original scale, the score should still be justified there, unless the score has been modified.

#### 4.3 Voices

In polyphonic music, a score may have more than one thematic line running through it. A thematic line is a coherent idea, usually carried by a single instrument, which has note durations independent of the other lines. Thus the flute may be playing a whole note while the violins are running through an arpeggio of thirtysecond notes. We call each of these lines a *voice*. Although most instruments can only handle one voice at a time, the piano can play many voices concurrently. Usually two - one for the left hand and one for the right. The word "voice" is frequently misleading since it usually implies that only one note is played at a time. We explicitly allow a voice to contain chords (or unchorded clusters of notes) as well as single notes.

Mockingbird allows up to 10 voices, numbered from 0 to 9. Every note belongs to exactly one voice, as determined by the user. The user can look at the collection of notes that belong to a particular voice by typing **look voice *n***, where *n* is the number of the voice the user desires. We say that the user is in the "voice mode", where *n* is the "selected voice". All subsequent commands will affect only these notes, and any note inserted will be inserted into this voice. The other voices will be drawn in a very light grey for reference. To look at all of the voices, type **look SHIFT voice**.

#### 4.4 Carry

There are two important aspects of the spelling of a note: how the note should be spelled, and whether or not that spelling should be shown. Every position on a staff allows for five accidentals: double flat, flat, natural, sharp, and double sharp. The combination of the accidental, the position on the staff, and the current clef uniquely determines a pitch. Frequently, there is no need to explicitly draw the accidental for a note, since it can be derived from the current key. There is also a convention in music that if a note is spelled outside of the key, it temporarily changes the key until the end of the measure. Thus any notes downstream of the first note that have the same pitch

and spelling do not need to explicitly draw their accidental. This suppression of accidentals is called *carry*.

Ninety-nine percent of all music is sufficiently regular that the carry rules are useful, but occasionally the music is so irregular that the carry rules only confuse the reader. In these few cases accidentals only apply to the notes that they are immediately in front of. For this reason, Mockingbird has a special mode that allows the user to enable or disable the carry rules at will. The commands for this are **look carry** (to enable the rules) and **look SHIFT carry** (to disable). The default mode is **look carry**.

## 5.0 SELECTION

The most common paradigm in Mockingbird for issuing commands is that the user first select a collection of notes or a section of the score and then issue a number of commands that apply to those notes or that section. To implement this, we have two different modes of selection: note selection and section selection. These modes are always disjoint - the user may use one or the other but not both. Starting a new type of selection will always clear the current selection. **ESC** will also clear the current selection, independent of its type.

### 5.1 Note Selection

The user may select a note by clicking **RED** while the cursor is over it, or by sliding the cursor over it while the **RED** button is down. Users will find it more efficient to select a number of notes by sliding the cursor over them all rather than clicking **RED** once for each of them. Up to one hundred notes may be selected at a time. If the user erroneously selects a note, he can deselect it by clicking **YELLOW** over it or by typing **ESC** (which will deselect *all* of the notes).

Once the desired collection of notes has been selected, the user may issue as many commands as he wishes without losing his selection. However, if he selects a new note after issuing a command, the old selection will be cleared unless the user uses *SHIFT RED*.

### 5.2 Section Selection

The user may select a section of the score by moving the cursor to the left edge of the section desired, pressing the **CONTROL** key, pressing **RED**, sliding the cursor to the right, and releasing **RED**. The area between the start of the selection and the current position of the cursor will flash as long as **RED** is down. If the cursor is moved to the left of the starting position, no selection is drawn. The user can also select the entire score by typing **everything**. Once a section of the score has been selected, the user may issue as many commands as desired without losing the current selection.

The user makes a secondary (grey) selection with *CONTROL SHIFT RED*. (Actually, the user can make a secondary selection by pressing the **SHIFT** key anytime while *CONTROL RED* is down. The original primary (black) selection will be restored and a secondary selection started. Releasing the **SHIFT** key changes the selection back to the primary selection and restores the original secondary selection). Only one primary and one secondary selection is possible at a time. Secondary selection is only used with the **replace** command (discussed later).

If the user wishes to extend either of the section selections after **RED** has been released, he may do so by pressing the appropriate **CONTROL** and **SHIFT** keys and *double* clicking **RED**. In this context, "double clicking" means pressing, then releasing, then pressing the mouse button again within a reasonably short period of time. The mouse button should end up pressed. Mockingbird will pick up the nearest end of the selection and allow the user to extend it in the normal fashion. This feature allows the user to start a selection, scroll up or down a few pages, and finish the selection.



## 6.0 NOTES AND RESTS

### 6.1 Movement

The user can move a note by placing the cursor near it, pressing SHIFT BLUE, and moving the cursor around until the note is captured. Once it has been captured, the user can change its position or pitch by moving it as desired. When the user moves the note vertically, the note's head will move in jumps from line to space to line, acquiring appropriate pitches and accidentals for each position. There are two modes for moving a note vertically: accidental mode and non-accidental mode. The user can change from one mode to the other by clicking YELLOW while still holding down BLUE. The default mode is non-accidental.

If the user is in the non-accidental mode, then Mockingbird will carefully choose the unique pitch and spelling for a note (given its position) which does not need an explicitly displayed accidental. To do this it will take into account the current key signature and any preceding notes whose accidentals carry to the note. Thus, the note will move from pitch to pitch within the current key signature unless it is downstream of a note with an explicit accidental, in which case it will assume the spelling of that note.

If the user is in the accidental mode, then Mockingbird will display all five possible accidentals for a note (double flat, flat, natural, sharp, double sharp) for each position that the notehead is in. The user chooses the spelling he wants just by moving the cursor up and down, in the same manner that he moves the notehead. As the user moves the cursor up, he will see the notehead first with a double flat, then a flat, then a natural, then a sharp and then a double sharp. If the user continues to move the cursor up then the notehead will jump to the next position and start over with a double flat, and so on.

Rests can never be moved to a line, but always must appear in the spaces in between. When they are moved, they will jump from space to space rather than space to line to space. There is no accidental mode for rests.

### 6.2 Insertion

To insert a note, the user must obtain a note icon from the menu (CONTROL BLUE) and deposit it where he pleases. The user deposits a note by moving the note icon so that its head is where the new note should be, and then pressing or clicking BLUE. While BLUE is pressed, the user will be able to move the note around just as if he had picked it up (see above). When BLUE is released, the note will be deposited in its position. (Just clicking BLUE will work, too.) Once the user has the note icon in the cursor, he may deposit as many notes as he desires without going back to the menu. As each note is deposited, it is automatically selected so the user may issue commands to change its attributes (see below). The default note has no note value; it is neither a whole nor a half nor a quarter. It will be inserted in voice 0 unless the user has selected a voice, in which case it will be inserted in the selected voice. If exactly one non-rest note was selected before a note is inserted, then the inserted note will take on the selected note's value. When the user is through, he should go back to the menu and get rid of the icon to avoid subsequent accidental depositions.

To insert a rest, the user should obtain a rest icon from the menu, and then proceed as with a note. The default rest is a quarter rest.

## 7.0 NOTE ATTRIBUTES

The user can change most of the attributes of notes by first selecting the appropriate notes and then issuing the desired command(s). The notes can be selected individually with note selection or collectively with an area selection. Again, only those notes in the current voice will be affected. The commands that follow will have the same meaning in either case.

**g**race - The selected notes become grace notes (**SHIFT g** changes them back).

**r**est - The selected notes become rest notes (**SHIFT r** changes them back).

**v**oice *n* - The selected notes become part of voice number *n*.

[ *n* (bracket *n*) - The selected notes are moved to the *n*<sup>th</sup> staff from the top.

^ (up arrow) - The selected notes' stems are all turned up.

\_ (left arrow) - The selected notes' stems are all turned down.

**1,2,4,8,6,3,7,0** - Sets the duration of the note. If the numbers are shifted, the notes will be dotted.

(**1**=whole, **2**=half, **4**=quarter, **8**=eighth, **6**=16th, **3**=32nd, **7**=64th, **0**=unknown).

### 7.7 Spelling

We discussed earlier how the user could specify the spelling of a note as he moved the note. For historical reasons there is a second way which may be removed sometime in the future. The second way of changing the spelling is to get an appropriate accidental from the menu (CONTROL BLUE) and to set the note with it. To set the note, the user should move the icon directly over the head of the note in question and click BLUE. If the suggested spelling is illegal for that pitch, Mockingbird will flash the screen. Changing the spelling may cause the note's head to move and the accidental may or may not be drawn depending on the context. The advantage of this method is that you don't have to move the note and the pitch is guaranteed to remain the same.

### 7.8 Embellishments

Mockingbird understands three different embellishments: a trill and two different types of mordents. It will correctly display and play all three. To add an embellishment to a note, the user should get the appropriate icon from the menu (CONTROL BLUE) and set the note with it. The latter is done by moving the icon directly over the head of the desired note and clicking BLUE. Currently, the only way to remove an embellishment is to delete the note. Once the user has the icon in the cursor, he may embellish as many notes as he desires without going to the menu again. When he is through, he should remove the icon from the cursor.

## 8.0 MEASURE LINES

The user may insert any of six different types of measure lines into the score: a standard measure line, a special measure line that looks like the standard but has a special meaning for Mockingbird, a double measure line, an end of section mark, and the left and right repeat indicators. The standard measure line is used to mark the boundaries of measures, and as such it is important for justification. All of the other measure lines serve this function, too. In addition, the repeat indicators and the special measure line have their own semantics. The repeat marks indicate sections of the score that should be repeated once. Mockingbird will correctly interpret the repeat marks and obligingly repeat the indicated sections when played. Mockingbird will handle nested repeats, too. The special measure line is used to mark style switches and ends of lines for the justifier (both discussed later). Its icon is the first measure line in the menu, the one directly above the natural sign. When it is inserted in the score, it will extend a little beyond the top and bottom

of the staves, although it is always printed as a normal measure line.

### 8.1 Insertion

To insert any of these, the user should get an appropriate icon from the menu (CONTROL BLUE) and deposit it wherever he desires. The latter is accomplished by pressing or clicking BLUE. As long as BLUE is down, Mockingbird will flash the measure line and allow the user to move it around.

When it is released, the measure line will be inserted. Pressing YELLOW before releasing BLUE will abort the insertion.

### 8.2 Movement

To move a measure line, move the cursor near it and press BLUE. Move the cursor around until the line is captured, and then to the desired location. Release BLUE. To delete a measure line, move the cursor near it and press YELLOW. Move the cursor around until the line begins to flash.

Release YELLOW, and it will be deleted. To abort a delete, move the cursor away until nothing is flashing.

These commands can also be used to move and delete key signatures, time signatures, metrenomes, clef switches, and octava switches (discussed later).

## 9.0 NOTE STRUCTURES

The following commands apply to a group of notes and say something about how those notes are related. Thus the user may select a group of notes and say that they should form a chord, or a beam. These commands are unusual in that they have a different meaning if a section is selected instead of notes. For instance, if a section of the score is selected and **chord** is typed, then Mockingbird will make *its best guess* about how the notes in that section should be chorded. This will all be discussed in the appropriate sections.

### 9.1 Syncs

Mockingbird's notion of voice is not very important-- it is merely an attribute of a note that the user can use for his and the program's convenience. What *is* important is the notion that there are collections of notes which all have the same metrical time (which all appear on the score "on top of" one another). We call such a collection of notes a *sync*, since it frequently synchronizes the voices. Mockingbird has an explicit data structure for syncs that it uses to distinguish between notes that are metrically equivalent and notes that are merely "very close". This data structure is fundamental to Mockingbird's understanding of the score, and is used everywhere in the program.

Although syncs are critical to Mockingbird's understanding of the score, the user rarely has to be conscious of what is synced and what is not. Most of the time Mockingbird makes very reasonable assumptions that match the user's intuition. However, if the user wants to see how things are synced together, he only needs to type **look sync** and Mockingbird will draw vertical lines through those notes that it believes are synced. This isn't foolproof, since the line drawn may pass through a note which is not part of the sync. The user should also check that the left edges of the notes are aligned. If the user wishes to sync notes explicitly, he should select the desired notes and type **sync**. This will create a new sync which consists of only those notes. If the user types **SHIFT sync**, a new sync structure will be created for each note, completely separating them. Chords are always treated as a single note: all of the notes in a chord will appear in the same sync. **look SHIFT sync** returns the user to the mode where sync lines are not drawn.

Actually, I overstated my case when I said that the notion of voice is not very important to Mockingbird. The truth is that both voices and syncs are very important to understanding the interactions between thematic ideas in polyphonic music. However, if the information given by the them ever conflicts, Mockingbird always believes the syncs rather than the voices. Furthermore, we

have found that Mockingbird's algorithms are often faster and simpler if they work with the syncs rather than the voices.

## 9.2 Chords

The user can chord a collection of notes by selecting the desired notes and typing **⌘**chord. This will create a new chord consisting of only those notes. The notes of a chord do not have to all be in the same voice, nor do they have to all be on the same staff, nor do they have to all have the same duration. The user can make a chord with whole notes as well. Typing **SHIFT** **⌘**chord will de-chord all of the currently selected notes.

If the user makes an *section* selection before typing **⌘**chord, then Mockingbird will chord the notes in the selected area as it sees fit. The algorithm it uses is to chord those notes within each voice that are all part of the same sync. If the user types **SHIFT** **⌘**chord when a section has been selected, all of the notes in the area will be de-chorded, just as above.

When the user chords some notes, Mockingbird will default the chord stem's direction based on the positions of the notes relative to the center of the most common staff. If the user wants to change the stem's direction, he should select a note within the chord and type ^ or \_, exactly as he would change a note stem's direction.

## 9.3 Ties

Pairs of notes or entire chords can be tied together with a single command by selecting the desired notes and typing **⌘**tie. If the current selection cannot be segmented into pairs of notes with the same pitch, Mockingbird will abort the command and flash the screen. If exactly two notes are selected, Mockingbird will tie them together even if they are not the same pitch. Both notes will still be played, however. This feature can be used to represent slurs. Typing **SHIFT** **⌘**tie will untie the notes, as might be expected.

To move a tie, press BLUE with the cursor near the center of the tie. Move the cursor around until it "captures" the tie, and then to the desired location. Release BLUE. If there are several ties close together, you may have to come at the tie from the top or bottom, or first move the other ties out of the way.

## 9.4 Beams and N-tuplets

The user can beam a collection of chords together by selecting notes from the desired chords and typing **⌘**beam. Think of individual notes as chords of one note. This will create a new beam which consists of only those chords. Mockingbird will ignore any chords with durations of a quarter or greater. It is perfectly legal to beam chords of mixed durations, and Mockingbird will automatically determine the number and grouping of the beams. Up to sixteen chords may participate in a single beam. Typing **SHIFT** **⌘**beam will de-beam the selected chords.

The user can manually change the position and tilt of a beam with the BLUE mouse button. To change the tilt of a beam, move the cursor near the right end of the beam and press BLUE. Move the cursor around until it captures the beam, and then up or down to the desired tilt. Release BLUE. Similarly, the position of the beam can be changed by picking up the left end of the beam. The beam can also be moved by changing the stem direction of its notes. Whenever ^ or \_ is typed, the beam will be moved to at least three lines away from the nearest selected note.

An existing beam can be turned into an n-tuplet by selecting notes from the beam and typing **⌘**n-**tuple** *nCR mCR*. The parameters *n* and *m* form a ratio *n:m* which determines how fast the notes are played relative to the standard meter. Usually, *n* is the number of chords in the beam and *m* is the

number of beats that those chords fill, but Mockingbird does not require this to be true. Thus a triplet may be represented as a 3:2 n-tuplet or a 6:4 n-tuplet, as the user desires. This flexibility is important when the durations of the chords of a beam differ.

If all of the selected chords do not belong to the same beam, then typing **n-tuplet nCR mCR** will create a new n-tuplet consisting of only those chords. This n-tuplet will be drawn with a bracket over it rather than a beam. However, **SHIFT beam** can still be used to "de-beam" such an n-tuplet. A bracket can be moved or tilted with the same commands as a beam is moved or tilted.

Frequently, the user will want Mockingbird to think of a collection of chords as an n-tuplet, but not display it as such. This happens when the n-tuple pattern is obvious from the context and drawing the numbers will only distract the musician. The user can accomplish this by selecting notes from the desired n-tuplets and typing **SHIFT n-tuplet**. The n-tuplet part will still be drawn (in light grey), but it won't be printed. Please note that this violates our convention of using SHIFT to negate a command. However, we felt that since the user can already use **SHIFT beam** to "de-beam" an n-tuplet, it would be safe to use **SHIFT n** for making n-tuplets invisible.

If the user wants to beam beams (or n-tuplets) together, he can do so by typing **CONTROL beam**. This might be used when the user wants eight sixteenth notes to appear as two beams of four sixteenth notes, or a 12:8 n-tuplet to appear as four 3:2 n-tuplets. This command is not restricted to beams alone, but may be used with mixtures of chords and beams. It can also be used to get around the sixteen note restriction given above. Beams formed with this command are exactly like regular beams and can be de-beamed, n-tupleted, moved, and tilted in exactly the same way. **CONTROL n** works analogously with bracketed n-tuplets, and the resulting n-tuplets are also exactly like normal n-tuplets.

If the user makes a section selection and types **beam**, then Mockingbird will make its best guess about how the notes in that area should be beamed together. The algorithm it uses depends on the time signature and voicing of the score in that area, so be sure that those are correct. Mockingbird will not touch existing beams, so the user must explicitly de-beam those notes he wants Mockingbird to default.

If the user makes a section selection and types **n-tuplet nCR mCR**, Mockingbird will make all of the beams in the selected area beamed n-tuplets. All of the other beam commands (**SHIFT beam** and **SHIFT n-tuplet**) have exactly the same semantics with sections as with notes, except for the **CONTROL** commands which are undefined.

## 10.0 SCORE ATTRIBUTES

There are a number of attributes (key signature, time signature, metrenome, sheet style) which apply to sections of the score rather than individual notes. To change one of these attributes, the user selects a section of the score (either manually or with **everything**) and types the appropriate command with its parameters. Mockingbird will change the attribute only for that section, and will reassert the prior value at the end of the selection unless the selection goes beyond the end of the score, in which case it doesn't bother.

Score attributes can also be change by moving or deleting "signatures" that already appear in the score. This is accomplished in exactly the same way that measure lines are moved and deleted: by using BLUE and YELLOW respectively. Key signatures and clef changes are special in that if they are moved to the far left of the line they will jump to a standard position on the sheet.

### 10.1 Keys

The user can change the key for a section of a score by typing **key keyNameCR**. The *keyName* can be a standard key name or it can be a number. By "key name" we mean a letter, optionally followed by **b** or **#** (for flat or sharp), optionally followed by an **M** or **m** (for major or minor). Thus it will accept names like **A, Bb, C#, d, DM, Em, and F#m**. Mockingbird will also accept a number between -7 and 7 (inclusive) which it interprets as the number of sharps in the key. (Thus **3** would be the key of A (3 sharps) and **-3** would be the key of Bb (3 flats)). The key signature drawn may include extra naturals, if the prior key contains more sharps or flats than the current key contains.

### 10.2 Time Signatures

The user can change the time signature for a section of the score by typing **/nCR mCR**. ("/" is found under "?"). The parameters *n* and *m* give the top and bottom numbers of the time signature. Thus **/3 4** would insert a 3/4 time signature.

### 10.3 Metrenomes

The user can change the metrenome for a section of the score by typing **metrenome nCR**, where *n* is a number from 16 to 1600 which indicates the tempo. The default is 128. Larger numbers are for faster tempos and smaller numbers are for slower tempos. The scale that Mockingbird uses is uncalibrated, it has nothing to do with standard metrenome values. The metrenome mark is never printed on the hardcopy output.

### 10.4 Sheet Styles

The user can change the style of a sheet used for a particular section by typing **look n**, where *n* is a digit in 0 to 9. The notion of "style" includes the number of staves used and the spacing between those staves. Currently, only the digits 0 through 4 have styles associated with them: 1 => one staff, 2 => two staves, 3 => three staves, and 4 => four staves. The digit 0 produces a sheet style appropriate for pianorolls, which we shall discuss later. Typing **SHIFT n** will produce a style with the same number of staves as the style *n* but a wider spacing.

The "signature" for a style change is a measure line that extends slightly beyond the staff set on both sides. This "signature" can be moved or deleted in the same manner as the others. It can appear anywhere in a line, although styles will only change at the beginning of the line. The rule for determining the style of a particular line is that the widest style (usually the one with the most staves) takes precedence over narrower styles.

### 10.5 Clef Switches

The clef switches (currently only bass and treble) are also sectional attributes, but since they only apply to one staff at a time we use a different mechanism for inserting them. To insert a clef switch, the user should pick up the appropriate icon from the menu (CONTROL BLUE), and deposit it on the staff that he wishes switched. The latter is accomplished by pressing or clicking BLUE. As long as BLUE is down, the clef sign will flash. As soon as BLUE is released the clef switch will be inserted and appropriate parts of the score redrawn. If the clef sign is moved to the far left of the line, then it will become part of the sheet. Pressing YELLOW before releasing BLUE will abort the insertion.

### 10.6 Octava

To move an octava that is already in the score the user should move the (empty) cursor to either end of the octava and press BLUE. He should then move the cursor around until it captures the octava. Moving the octava up or down changes its height, moving it left or right changes its range. If the right end is moved to the left of the left end (or vice versa), then no octava is drawn. The user cannot change the staff that the octava is on. Releasing BLUE will reinsert the octava (or delete it if no octava is drawn).

Octava marks are also inserted from the menu. To insert an octava mark, the user should obtain the "8" icon from the menu (CONTROL BLUE). Inserting an octava is very similar to making a section selection. The user should move the cursor to the left edge of the staff that he wants an octava on and press BLUE. Holding BLUE down, he should move the octava to the right until it covers the desired section. Moving the octava up or down will change its height. If the octava is moved far enough up or down, the octava's staff will be changed. If the cursor is moved to the left of the original point, no octava is drawn. Releasing BLUE causes the octava to be inserted, and appropriate parts of the score to be redrawn.

## 11.0 SCORE MANIPULATIONS

### 11.1 Delete

The user can either delete notes individually or delete a whole section of the score, depending on which type of selection he uses. If notes are selected, then they will be carefully removed from all of the structures they participate in and then thrown away. If a section of the score is selected, then everything (notes, measure lines, etc.) will be deleted and the remainder of the score moved over to cover the blank space. Although "signatures" may be deleted, Mockingbird will take care to preserve the score attributes for the following section. This action is conceptually like cutting a section of the score out and pasting the remaining pieces of the score together.

However, if the user is looking at a voice by itself, a sectional **delete** will not cut the paper out from under the score. Rather, it will only the notes in the selected area that belong to the given voice as if they had been individually selected and deleted.

### 11.2 Replace

To replace one section of a score with another, the user should make a primary (black) selection of the material to be deleted, a secondary (grey) selection of the material to be inserted, and type **replace**. Mockingbird will delete the primary selection and insert a copy of the secondary selection in its place. It will also insert key signatures, clef switches, and octavas as necessary to preserve the score attributes of the copy.

If the user is looking at a voice by itself, then **replace** will have a slightly different meaning. First of all, Mockingbird will delete only those notes in the primary selection that are in that voice (as in **delete** above). Second, it will only copy the notes in the secondary selection that are in the voice, ignoring other notes and all other score events. Finally, it will attempt to merge the copy into the primary selection on a measure-by-measure basis, aligning the copied voice with the other voices if possible. If the two sections do not have the same number of measures, then the merge will be aborted.

### 11.3 Transpose

The user may transpose a collection of notes by selecting them (with either style of selection) and typing **transpose nCR**. The parameter *n* is the number of half-steps that the pitch should be moved up or down. Thus **transpose 7CR** will transpose a note up a fifth and **transpose -12CR** will transpose a note down an octave. All user-defined spellings for a note will be lost unless the user transposes by some multiple of an octave. Transposing a section of the score never changes the key of the section.

## 12.0 LOGICAL MODE

As we discussed earlier, there are two different modes for representing music: logical mode and pianoroll mode. Our *logical mode* is the mode in which common music notation rules; music is thought of as a number of voices consisting of successions of discrete note durations (whole, half, quarter, etc.). This is also the mode in which the user is concerned about the aesthetic placement of the musical entities on the sheet of paper. The default mode in Mockingbird is the logical mode. To enter the logical mode from the pianoroll mode, the user must type **look justified n**.

### 12.1 Justification

Justification nominally means arranging the score so that there are an integral number of measures per line. We also use the term to mean alignment and layout. *Alignment* means to re-sync the notes of a measure so that the voices are properly aligned or registered. This is the only time that Mockingbird will ignore the syncs in favor of the voices. Mockingbird only does this when a pair of voices both "fill" a measure and the voices taken together add up to more than a measure. (A voice "fills" a measure when the sum of its durations exactly equals the total duration implied by the time signature for that measure). This feature allows the user to be careless about the placement of notes relative to the other voices as long as he makes sure that he fills out the measure with appropriate rests.

*Layout* means to find an aesthetically pleasing horizontal spacing for the musical entities of a score. (The vertical spacing is either already constrained (as for note heads and measure lines) or is left up to the user (as in beam and tie heights)). There are two aspects to determining a layout: a purely graphical aspect which specifies that objects cannot be too close or they will overlap; and a semantic aspect which says that the distance between two note heads is related to the metrical distances between the notes. Both alignment and layout are always automatically performed whenever the user justifies a section of the score. Of course, after the justification is finished, the user is free to touch up the alignment or layout as he sees fit.

To justify a section of the score, the user should first select the appropriate section and type **look justified n** (or *ESC*). Mockingbird will change the selection before justifying so that it encompasses an integral number of measures. It will then align the voices in the selected measures, layout the section, and then arrange it so that an integral number of measures appears on each line for that section. The end of the selection will always appear at the end of some line. Finally, the remainder of the score will be moved to begin on the next line. This last step may ruin the justification of



the following measures, but it was thought better to only mess with those measures that the user had explicitly selected. The parameter  $n$  affects the looseness of the layout that Mockingbird will perform. A larger  $n$  produces a looser layout. If the user types *ESC*, then Mockingbird will use that last  $n$  that the user specified.

The line breaks for a justified score may be determined in three ways; the user may justify the score line by line, he may justify the entire score and leave the line breaks up to Mockingbird, or he may explicitly designate line breaks by replacing normal measure lines with a special measure line. The justifier always honors the special measure line by making sure that it appears at the end of some line. (The special measure line is the first measure line in the menu; it is drawn with its line extending slightly beyond the boundaries of the staves). Thus, a useful heuristic for justifying a score is to first allow Mockingbird free reign with the entire score until the general spacing is acceptable, and then to rejustify with manually specified line breaks so that the end of the score comes at the end of some page. (The user may also want to arrange the music so that phrases end at the end of a line and page turns come at reasonable places).

Occasionally, the user may want to change a non-standard measure into a special measure to force a line break at the end of that measure. The user should do this by putting the special measure after the non-standard measure rather than by replacing the one with the other. The justifier will arrange the two measure so that they fall on top of one another, and the score will print showing only the non-standard measure.

## 12.2 Checking Voices

An important constraint in common music notation is that the voices of a measure sum up to the time signature for the measure, both separately and taken together. Mockingbird does not require that this always hold, but it does help the user to verify it, if he desires. If the user wants to check that a measure sums up correctly, all he needs to do is type x (for **e xamine**). No selection is necessary. Mockingbird will then grey any non-empty measure which does not sum up to its time signature. If the user is viewing a particular voice, Mockingbird will only check that voice. If the user is looking at all of the voices, it will find the sum of all the voices taken together, taking into account their interactions through the syncs. The user can get rid of the grey by typing x again or by causing the screen to be repainted.

## 12.3 Playback and Record

At the top of the screen are two commands that allow the user to use a synthesizer. Please note that the special microcode *must* be loaded to use these commands. To issue one of these commands, move the cursor over the command until it turns into a bull's-eye and then click any of the mouse buttons. (For Playback only: RED=half time, YELLOW=normal time, and BLUE=double time). The box around the command should then invert. To terminate either of these, repeat the above. (Playback will terminate itself when it comes to the end of the score).

When Mockingbird plays music in the logical mode, it will play the common music notation version of the score, unless some section has no durations assigned. Mockingbird plays as a student might play; very rigidly and meticulously. We have make no attempt to teach Mockingbird "expression" since our purpose in having it play is to be able to proofread the score. However, if the user is concerned with this aspect, he can often kludge in some expression with the help of well placed metronome marks (loudness control is currently impossible).

If the notes of a section do not have any durations assigned, Mockingbird will play from the pianoroll representation for that section if it has one. As soon as it encounters more durations, it will revert to the logical mode. This feature allows the user to move from the pianoroll mode to the logical mode even though his score has not been fully "logicized".

Mockingbird's player is a separate process from the rest of the program. This allows the user to continue to edit the score while music is being played. However, since the score is an unmonitored data structure common to both processes, the user should be warned that changing the basic data structure may cause the player to play strange things. Since this is a transitory effect we decided not to restrict the user's freedom, but rather *caveat emptor*.

When Mockingbird records music in the logical mode, it inserts the resulting notes into the score as if a **replace** had been performed. It is impossible to record against the playback in the logical mode, since it would be very difficult to determine what goes with what. To record against Mockingbird's performance, the user should move to the pianoroll mode.

#### 12.4 Printing

At the top of the screen is a command that allows the user to print the score. Clicking the command will cause a file to be produced which contains the information necessary to print the score. A hardcopy version of the score can be produced at any time by sending this file to a printer. The file can be saved for future use.

#### 12.5 Automatic Features

Mockingbird has a number of aesthetic adjustments that it performs automatically while in the logical mode. These adjustments are completely automatic, there is no way for the user to override them. However, it is probably reasonable for the user to be aware of them.

If the notes in a chord are too tight, then Mockingbird will stagger the notes so that all of the noteheads can be seen. If two noteheads have exactly the same height but different spelling, then Mockingbird will put the noteheads side by side. (The latter is a variant technique in common music notation, the standard way to handle this case is to use a slanted stem that branches off of the main stem).

If the distinct notes and chords of a sync are too close together, then Mockingbird will move some of the notes or chords slightly to the left or right so that all of the stems remain visible. Mockingbird may also stagger some of the noteheads to avoid overlap. The only exception to this is when two notes have the same pitch and spelling but with opposite stems: in this case, the noteheads are allowed to overlap.

If the accidentals of the notes of a sync are too close together, then Mockingbird will move some of the accidentals to the left to make room for the rest. Mockingbird will also avoid collisions between staggered noteheads and nearby accidentals. The rules used for accidental placement were determined inductively by examining published scores.

If the score is in the hardcopy mode, Mockingbird will arrange the lines of the score so that an integral number of lines appear on each page. It will do this even if there are several staff styles on a page. The general algorithm is to find out how much white space is available and then divide it equally between the lines. If the result is too tight, then the last line is moved to the next page. The only way that the user can affect the number of lines per page is to change the sheet styles for some lines. The SHIFTEd styles are especially useful for forcing the score to have fewer lines per page.

### 13.0 PIANOROLL MODE

The second of the two modes supported by Mockingbird is the pianoroll mode. The *pianoroll mode* has music represented much like the old-time player piano rolls. The most important aspects of this is that the time and duration of a note are measured in seconds (rather than beats), and that the positions of the notes in the pianoroll are directly proportional to the time that the notes are played.

The pianoroll mode is most useful when a section of music has not yet been "logicized", i.e. when the voices have not been determined and no durations have been assigned. It allows the user to capture music on the synthesizer and play it back in its raw form. The user can also use this mode to compare the sound of his "logicized" version of a score with the original rendition. To enter this mode, the user should type **look pianoroll *n***, where *n* is the scale factor used to determine the spacing (Larger *n* means more space between notes).

#### 13.1 Special Looks

Mockingbird has a special sheet style (**look 0**) which is especially convenient for pianorolls. This style has the property that the vertical position of a note is directly proportional to its pitch and independent of the staff it is on. The top staff of the style is *two* octaves above the treble, and the bottom staff is *two* octaves below the bass.

Normally, Mockingbird will display accidentals for pianoroll notes just as they are displayed in common music notation. However, this may clutter up the score since the pianoroll layout algorithm doesn't make any provisions for preventing overlaps between notes and accidentals. To get around this, there is a special mode that displays the black keys of the synthesizer "in between" the spaces and lines of the staff. This allows the user to still see the exact pitch of a note even though no accidental is displayed. The user enters this mode by typing **look SHIFT accidental**, and returns to the standard mode by typing **look accidental**.

Also, Mockingbird continues to draw common music notation noteheads for the notes, even though pianorolls normally have variable-length bars. This is done to give the user some visual clues about where he is in the score ("There's that sixteenth note pattern-- I must be in the second movement"). However, if the user really wants to see the variable-length bars, he can type **look SHIFT notehead**. To return to the standard mode, he should type **look notehead**.

#### 13.2 Recording

The user records music in this mode just as he did in the logical mode: he first selects the area he wants replaced, and then invokes "Record" at the top of the screen. Mockingbird waits until the first note is struck on the synthesizer, so there is no need to rush to it after invoking "Record". When the user is through playing on the synthesizer, he must turn off "Record" to see the new notes displayed. The notes that were selected before will have been deleted, just as if a **replace** had been performed.

#### 13.3 Playing Against Oneself

The pianoroll mode has a different type of "Playback" than the logical mode. Note that when "Playback" is invoked, the "Record" command will be inverted at the same time as the "Playback" command is. This means that Mockingbird is prepared to record anything that the user might play *while Mockingbird is playing*. When "Record" is turned off, the recorded material will be aligned with the performed material and merged into the score. This feature allows the user to build up a composition voice by voice.

Recorded material is always inserted in voice 0 unless the score is in the voice mode, in which case it is inserted in the selected voice. We suggest that the user always record new material into an empty voice to keep it separate from previously recorded material. This will make it easier for the user to experiment with different ideas, throwing out the ideas that are unsuitable.

The user may be interested to note that the music tracking cursor behaves differently in the pianoroll mode. In the logical mode, the cursor leaps from note to note as a note is struck. In the pianoroll mode, the cursor moves at a constant pace along the score.

#### 13.4 Beating In Measures

When the user has produced a pianoroll version of his piece that satisfies him, he may begin to convert it into common music notation. One of the first steps in this process is to insert measure lines into the score. This is normally a time-consuming process, and so we have provided a means for speeding up this process. Whenever music is being played, the user can insert measures by just tapping the space bar. A measure line will then be inserted into the score a little before the current cursor position. The user should tap *on* the first beat of the measure, not before. This feature allows the user to insert all of the measures needed in the time it takes to play through his piece once.

#### 13.5 Automatic Syncing

As we discussed earlier, syncing is an vital aspect of how Mockingbird looks at scores. It is important to get the syncing of the score correct before moving to the logical domain. When Mockingbird records music it syncs some of the the notes automatically, but only within a very tight tolerance. The user can go through and manually sync those things that should be synced, but a better approach is to ask Mockingbird to try again with a looser tolerance. If the user makes a *section* selection and types **sync**, then Mockingbird will do exactly this. If there are still mistakes, the user will have to fix them himself.

#### 13.6 Automatic Durations

If the user asks, Mockingbird will make its best guess about what it thinks the logical durations of the notes should be. Its heuristic requires proper measure lines, time signatures, and *voicing*. The heuristic does not try to guess how things should be voiced, this is completely up to the user. To use the heuristic, the user should type **look physical n** (this isn't required, but it helps since it moves the measure lines to a fixed place relative to the notes that it marks), select the desired measures, and type **CONTROL guess**. The heuristic will also chord things together. If the user makes a mistake, he can remove the durations by typing **0**, and the chords by typing **SHIFT chord**.

#### 13.7 Automatic Beaming

It is worth repeating here that there is a heuristic available to the user that makes a guess about how the notes should be beamed. To use the heuristic, make a *section* selection and type **beam**.

#### 13.8 Overwriting the Pianoroll

When the user has "logicized" a pianoroll to his satisfaction, he can replace the recorded pianoroll with a version derived from the logical version by typing **look logical**. This allows the user to play new voices in against a perfect rendition of his score, rather than the less than perfect original.

### 13.9 Another Way to Enter Notes

So far we have discussed entering notes from the menu one by one and from the synthesizer all at once. A third method also uses the synthesizer, but in a slightly different way. The third method is to enter a piece in a voice at a time, *without worrying about playing the notes in time*. This way, the user can take his time positioning his fingers over the keys before he plays them. The resulting pianoroll will be spaced wildly, but if the user justifies the section it will be made more uniform. After inserting measures, the user can merge the voices together by replacing sections while in the voice mode (see **rreplace**).

The disadvantages of this approach are that the user cannot beat in measures or have Mockingbird guess at the durations. The advantages are that the score will already be voiced properly, and very few notes will be misplaced.

## APPENDIX

This appendix gives the nitty-gritty details of actually running Mockingbird. It isn't guaranteed to be in date. If you have any questions, please contact John Maxwell or Severo Ornstein.

## A.1 How to Obtain Mockingbird

Mockingbird and all of its supportive files are kept in a dump file on [Indigo]<Mockingbird>. To obtain this dump file, the user should type **FTP indigo dir/c mockingbird load/c mockingbird.dm** to the Alto Executive. This command line form is preferable to using FTP directly since the user will not have to confirm the transfer of each file.

## A.2 How to Run Mockingbird

Mockingbird will only run on a Mesa 6 partition of a Dorado. Furthermore, if you desire to use the synthesizer, you must use a Dorado with the appropriate hardware (currently Forrestal, #16). You will find instructions for using the synthesizer in section A.3.

To invoke Mockingbird, all you need to do is type **mockingbird filename1 filename2 ...CR** to the Alto Executive. If more than one filename is given, then each one beyond the first will be appended to the end of its predecessor. If no filenames are given, then Mockingbird will start up with a blank piece of sheet music. After Mockingbird has finished loading all of the files (if any) it will display its creation date at the top of the screen. Please note the creation date to see if the system has changed since you last used it. The user can also file in files from the editor. This is discussed in section 3.

If this is the first time that you have used Mockingbird, we suggest that you start by typing **@demo.cm** to the Alto Executive. This will load mockingbird and file in a demo score that you can play with as you read this manual.

Hardy pioneers may find a file called "music.bcd" left on a public partition because of development work currently in progress. This is the newest version of Mockingbird, and has all the newest features and the newest bugs. Use at your own risk.

## A.3 The Synthesizer

The synthesizer is only necessary if you want to record or play music. Most people will find that Mockingbird is sufficient by itself. If you have never hooked up the synthesizer, you should first get a demonstration from someone knowledgeable about the ins and outs. The following explanation is only for those who have already seen it done.

To use the Yamaha synthesizer you must sign up for Dorado #16 (Forrestal) and the floating terminal that resides in the Maxc room. Put a note on Forrestal's upstairs terminal warning people that the terminal is in use downstairs. Make sure that Forrestal is off or in DMT before switching the terminal (switching terminals may sometimes cause a boot). You may obtain a key for the Dorado room from the guard at the desk. *Be sure that you are switching terminals for the correct Dorado!* The old terminal lead should have a small "16" printed on it. Return the key to the guard's desk.

To turn on the synthesizer, push the "POWER" button. The red light next to it should come on. Make sure that there is a cable connecting the "I+II" socket on the right side of the synthesizer with the "MICROPHONE" socket on the left side of the amplifier. (If there isn't an amplifier, check with one of the people in the Voice project, it is actually their amplifier). Turn the power on for the amplifier, and move the "MIX" slider up to "10". You can test the system without the Dorado by flipping a small toggle on the right side of the synthesizer up and playing a few notes. Be sure

to check the volume controls on both the synthesizer and the amplifier. Flip the toggle back down.

To use the synthesizer, you *must* load the microcode. Do this by typing **loadmb music.mbCR** to the Alto Executive. *The special microcode must be reloaded every time the Dorado is triple-booted.* Mockingbird will notify you if you have failed to load the microcode.

When you are finished with your editing session, be sure to turn off the synthesizer and the amplifier and to *reconnect the upstairs terminal*. You should also remove your sign.

#### A.4 Printing

At the top of the screen is a command that allows the user to print the score. The precise meaning of this command depends on which mouse button is used and which scale-mode the score is in. If the score is in the hardcopy mode ( **look hardcopy**) and either RED or YELLOW is clicked, then Mockingbird will print the entire score using the standard font. The resulting press file will be paginated and will be ready to print on any full-Press printer which has *mockingbird.ac* in its font dictionary. (Currently, Stinger in ISL is the only printer that meets these restrictions). If the score is in any other mode, or BLUE is clicked, then only the current page will be printed and it will be printed entirely with splines. The resulting press file will be very large, but it can be printed on any printer that supports splines (Platemaker or the Versatec, for instance). To print the press file on Stinger, quit from Mockingbird and type **Empress music.press [n/c] stinger/hCR** to the Alto Executive (*n* is the number of copies (optional)). Printing on Platemaker or the Versatec requires the aid of a wizard.

#### A.5 Emergency Recovery

If the user finds himself in the debugger he can sometimes recover his file by invoking a procedure designed for that purpose. This procedure will not file out an inconsistent file, so it can never hurt to try it. To invoke the procedure, type:

```
>set root configuration: MusicConfig (capitals are important!)
>set module context: FileImpl
> (space) FO[]<CR> (FO is for FileOut)
```

The procedure will return a boolean that tells whether or not its attempt was successful. If it was, the music will be found in the file named "temp.music".

## MOCKINGBIRD - December 17, 1981

### MOUSE ACTIONS

<u>Context</u>	<u>RedBug</u>	<u>YellowBug</u>	<u>BlueBug</u>
<b>PLAY</b>	slow	medium	fast
<b>scrollbar</b>	scroll up	thumb	scroll down
<b>overview</b>	***	thumb	***
<b>(in score)</b>	select note	deselect note/ delete event/ switch insertion modes	move beam(left=y,right=tilt)/ insert from menu/ move event
<b>SHIFT</b>	extend note selection	xxx	move note
<b>CONTROL</b>	section select	xxx	select from menu
<b>CONTROL-SHIFT</b>	grey section select	xxx	xxx

### COMMANDS

<u>Command</u>	<u>SHIFTED</u>	<u>Section selection</u>	<u>Note selection</u>
<b>(b)eam</b>	clear beams	heuristic beam	beam notes
<b>(^b)eam</b>	debeam beams	xxx	beam beams
<b>(c)hord</b>	clear chords	heuristic chord	chord notes
<b>(d)elete</b>	xxx	delete section	delete notes
<b>(e)verything</b>	xxx	section select all	...
<b>(g)race</b>	ungrace notes	grace notes	ditto
<b>(^g)uess</b>	xxx	guess the durations of the notes	xxx
<b>(k)ey+keyNameCR</b>	xxx	set key to keyName	xxx
<b>(l)ook</b>	xxx	<i>see looks section</i>	
<b>(m)etrenome+nCR</b>	xxx	set metrenome to n (default=128)	xxx
<b>(n)-tuplet nCRmCR</b>	make "invisible"	make beams into n-tuplets	set notes to n/m n-tuplet (3/2=triplet)
<b>(^n)-tuplet nCRmCR</b>	debeam n-tuplets	xxx	n-tuplet beams
<b>(p)ransPose (-)nCR</b>	xxx	move the pitches +/- n half steps	ditto
<b>(r)eplace</b>	xxx	black selection _ grey selection	xxx
<b>(r)est</b>	make rests into notes	xxx	make notes into rests
<b>(s)ync</b>	clear syncs	heuristic sync	sync notes
<b>(t)ie</b>	untie notes/chords	xxx	tie selected notes/chords together
<b>(v)oice n</b>	xxx	set voice to n	ditto
<b>(x)examine</b>	xxx	sumcheck ALL measures	...
<b>(ESC)deselect all</b>	xxx	make null selection	...
<b>(DEL)terminate</b>	xxx	terminate command	...
<b>(/)time nCRmCR</b>	xxx	set time signature to n/m	xxx
<b>(I)staff n</b>	xxx	put notes on the nth staff from top. (top=1).	ditto
<b>(^)stem up</b>	xxx	set note stems up	ditto
<b>(_)stem down</b>	xxx	set note stems down	ditto
<b>(1)whole</b>	dotted whole	set note values to whole	ditto
<b>(2)half</b>	dotted half	set note values to half	ditto
<b>(4)quarter</b>	dotted quarter	set note values to quarter	ditto
<b>(8)eighth</b>	dotted eighth	set note values to eighth	ditto
<b>(6)sixteenth</b>	dotted sixteenth	set note values to 1/16	ditto
<b>(3)32nd</b>	dotted thirtysecond	set note values to 1/32nd	ditto
<b>(7)64nd</b>	dotted sixtyfourth	set note values to 1/64th	ditto
<b>(0)unknown</b>	xxx	set note values to unknown	ditto

### LOOK COMMANDS

<u>Commands</u>	<u>SHIFTED</u>	
<b>(a)ccidental</b>	in-the-cracks notation	key and accidentals are shown
<b>(c)arry</b>	accidentals don't carry	accidentals carry (apply to notes that follow in the measure)
<b>(h)ardcopy</b>	normal scale	hardcopy scale (scale music will be printed at)
<b>(j)ustified n</b>	xxx	justify with scaling of n (n becomes default)
<b>(l)ogical</b>	xxx	turn the pianoroll into a perfect rendition of the logical score
<b>(n)otehead</b>	bar length gives duration	note shape gives duration (whole, half, quarter...)
<b>(o)verview</b>	return to prior scale	go to overview
<b>(p)ianoroll nCR</b>	xxx	horizontal position is linear with playing time (n=scale)
<b>(s)ync</b>	don't show the sync lines	show the sync lines
<b>(v)oice n</b>	display all of the voices	display only voice n. Editing actions will only affect this voice.