Bravo Course Outline

by Suzan Jerome

XEROX

PALO ALTO RESEARCH CENTER
3333 Coyote Hill Rd/Palo Alto/California 94304

Table of Contents

Part II

Appendices

Summary of Commands

Preface

The Outline is designed as a self-teaching aid for new Bravo users. It is written in three parts, progressing from introductory level information to Bravo's more complex editing/formatting features. Part I provides sufficient information for a user to prepare memoranda and letters with Bravo, and to format them with preset formatting options. Part II describes Bravo's more complex formatting options. Part III describes remaining Bravo features and references further documentation.

Bravo was designed by Butler Lampson and Charles Simonyi, and implemented largely by Tom Malloy, with substantial contributions from Carol Hankins, Greg Kusnick, Kate Rosenbloom and Bob Shur.

Several people contributed their time and energy to read and comment on the outline. They are: Richard Sweet, Betty Burr, Diana Merry. Their contributions are much appreciated by the author, as, if known in detail, they would surely be by the reader. And, while acknowledgements are the topic, PARC's secretarial staff is heartily acknowledged for providing a great deal of support and gracious assistance to new, non-technical users of Bravo, and for supplying them with some of the folklore surrounding experimental systems.

Introduction

This course in Bravo is separated into three Parts. Part I contains introductory material and is designed to be used in conjunction with Exercises 1 through 8, which are printed separately and should be obtained before beginning Part I.

The course may be discontinued after Part I if user needs are satisfied. Part II contains a number of formatting options, many of which are necessary if Bravo is to be used for technical reports requiring precise formatting and special characters (Greek and math symbols, for example). Part III describes the remainder of Bravo's features.

A Summary of Commands is given at the end of the document. The keyboard and/or mouse sequence for each command is listed, followed by the page number where further explanation can be found within the outline. Since the command summary is presented in dictionary form, it can be used either as a ready reference, or as a partial index for the outline.

For new users the outline may serve as a course in Bravo, and the user should start at page 1. For established Bravo users, the Summary of Commands may be the most interesting section. For those who know Bravo, but are using a feature for the first time, the description of that feature can be located quickly by referring to the command summary/index or to the table of contents.

New Users

Some effort has been made to provide sufficient explanation to make self-teaching possible. However, if a choice exists, the preferred method for learning Bravo is always time spent with an individual already competent in its use. Learning Bravo solely through the use of this Outline will be somewhat difficult and should not even be attempted until several start-up activities have been performed. Before beginning, *ask someone to assist you* in performing the following operations:

1. Have a non-programmer's disk copied (described on pages 2-3 of the *Alto User's Handbook*).

2. Have the disk installed with your Maxc (or last) name and a disk name of your choice (described on page 4 of the *Alto User's Handbook*).

3. If they are not already colored, have the mouse buttons labeled **Red**, **Yellow**, and **Blue**.

4. If they are not already marked, have the following keys on either the Alto I or Alto II keyboard labeled:

   DEL
   BS
   ESC
   CLR
   LF

5. Have the following file copied onto your disk:

<Forms>Form.Memo1

DEL and BS Keys

There are two keys that will be particularly useful to a new user. They are DEL and BS (backspace). DEL is used to delete (abort) a command. It is quite easy to activate a command accidentally since commands in Bravo are single character commands. Striking the DEL key will delete the command and allow you to start again with the command intended. BS (backspace) is used to correct typing errors, whether they occur during a command at the Alto Executive level or while typing text in Bravo.

Exercises

The exercises which accompany Part I are a central part of the presentation. Most of the features are described as part of an exercise in which each feature is *used*. The reviews are also an important element in Part I. They are designed to reinforce material already described before introducing new material. For *new*, non-technical users of Bravo, it is *strongly* suggested that both the exercises and the reviews be performed in the order given.

Summary

An effort has been made to provide ample explanation and to forestall difficulties; however, *there will* be points of confusion and/or Bravo problems which were not foreseen. Therefore, it is a good idea to locate a Bravo user who is regularly available and willing to assist you when difficulties arise.

For the first week or two of Bravo use, it would facilitate the learning process if you were not responsible for producing actual work while learning. Use of the exercises and files suggested will provide an element of ease not to be found in work which is deadline driven; and without the pressure of others depending on you for their work, it may be possible to find elements of humor scattered throughout the learning process.

Part I

Conventions:

What the user types will appear in boldface and underlined--e.g., **get** indicates the **get** Command, for which the user types only **g**.

Examples of system messages are indented and appear in boldface.

CR	=	Type a Carriage Return
ESC	=	Type the Escape Key
↑	=	An up arrow preceding a character indicates that the following character is a Control Character--e.g., **↑b** means that the Control key is *held down while* **b** is typed.

Loading the Disk

On the front of the Alto there is a glass door with a handle across the top. Below the glass door on the *left* is a white switch which is labeled **Load Run**. Below the glass door on the *right* is a white **Load light** and a Yellow **Ready light**. Locate the Load/Run switch, the Load light and the Ready light.

When the white switch is on **Load**, and the white **Load light** is on, the glass door can be opened and a disk inserted. There is a *slight* resistance on the door when it is opened.

Hold the disk flat side down, with the label facing you, and slide it gently into the tray; then push it slowly back until it stops. Now close the door and push the white switch on the left to **Run**. When you do this the white **Load light** will go out. Wait about a minute until the Yellow **Ready light** comes on.

Starting (Booting) the Alto

Once the disk is loaded and the Yellow **Ready light** is on, the Alto can be booted (started).

On the back of the keyboard there is a thick black cable and a thinner black wire; in the middle of them, on the casing of the keyboard, is a "boot" (start) button. Push this button to boot the Alto.

When you boot (do it now), the screen lights up and some general information is displayed at the top of the screen: the "Alto Executive" level is identified; the date and time are given; the number of pages left on the disk, which version of the Alto Operating System (OS) is being used, the number of your Alto, your name, and the name of your disk.

Toward the middle of the screen, is a "herald" or marker for the Alto Executive. It is a greater than (>) sign. A vertical bar blinks on and off in front of it, marking the point at which any keyboard command will appear.

The Alto Executive

The Alto Executive serves as a controller for the entire Alto system. There are a number of different programs on the disk, some of which you will not use. All programs, Bravo included, are initiated from the Alto Executive level.

I. Introduction to Bravo

To access Bravo from the Alto Executive, type:

>BravoCR (CR means type a carriage return, not the letters CR)

Before beginning any operation, notice what the screen looks like, how Bravo displays itself and what information it initially gives the user.

A. Bravo's System Window

Bravo displays two separate areas called windows. The small window at the top of the screen is called the "system window." The system window displays operational information, confirmation of commands, and hints about what the system expects you to type next. When Bravo is first started, the message in the "system window" reads:

READY: Select operand or type command
BRAVO Version 6.1 November 1, 1976

The second line states that Bravo is the name of the system, which version is being used, and the date that version was released.

The first line states that Bravo is "Ready." The Ready message is followed by two possible activities: "select operand" or "type command." The term "operand" need not be intimidating. Select operand simply means "choose the thing you want to perform an operation on." To perform an operation, you need something to perform it on; that something is an operand.

Get Command

Since no text yet exists, there are no operands to select. So, in this state there is only one option: type a command. The command to start with is: "Get a memo form." To begin, type **g** for get (it does not matter whether it is upper or lowercase g.)

Bravo commands are single character commands. When **g** is typed, Bravo supplies the remainder of the word Get. The first line in the system window now reads:

GET: Type or select filename terminated by ESC

The information following Get is a "prompt". A memo form is stored on the disk, and it has a filename. Bravo is asking for the name.

[The term "file" and the term "document" are used interchangeably in this outline--all documents created in Bravo are stored as "files" on the disk.]

In response to Bravo's "Type or select filename" prompt, *type*:

Form.Memo1

The filename will appear *within* the wide black border just below the system window, and a blinking "caret" marks the point at which the characters appear as they are typed. Text appearing within this black border should *always* be a filename.

Bravo says "Type filename terminated by ESC." The Escape Key is used to tell Bravo that a command is finished and that it should now act on the command. When you hit ESC (do it now, if you have not already), you will hear disk movement and see a message in the system window: "Reading file." Shortly after that the file will appear on your screen. Bravo will display another READY sign, indicate the size of the file and verify its filename.

B. Document Window

The large (document) window on the screen now contains **Form.Memo1**. The system window will continue to supply information about current actions, while the bulk of the screen is set aside for creating the document. Each time you get a file, it will be displayed in the document window.

C. First Set of Mouse Functions

Up to this point Bravo has been controlled from the keyboard. A second way to control the system is by using a pointing device called the "mouse." The small white box, *usually* placed to the right of the keyboard, is the mouse. It connects to an arrow (cursor) on the screen and allows you to point at (select) the blocks of text to be altered or expanded. *[See "Hints about Using the Mouse" at the end of this section.]*

Single Character Selection

Move the mouse around on the desk and notice the correlation between the movement there and the movement of the cursor (arrow) on the screen. Place the cursor underneath a *single character* on the screen and push the **Red** button. That single character (did you get the one you were aiming at?) will be underlined, indicating that it is "selected." Once selected it becomes the "operand" (remember that term?). The character selected and *only that character* will be affected by any command given.

A selection may be changed frequently; nothing will happen until you stop selecting and give a command.

Move the cursor to another single character and push the **Red** button again. Notice that the character first selected is no longer underlined--you have a "new" selection. Repeat this kind of selection half a dozen times before continuing.

Word Selection

Place the cursor (arrow on the screen) under any character in a *word* and push the **Yellow** mouse button. The whole word will be underlined (selected). Make half a dozen of these word selections before continuing.

Extending a Selection

Select a single character using the **Red** mouse button. Once that selection is made, *hold down the* **Blue** mouse button and *extend* the selection by moving the cursor along the line, down onto the next line, and finally down to the bottom of the file. Having started the selection with a single character, Bravo "extends" the selection through single characters for as long as the cursor is moved with the **Blue** button held down; when the button is released, the selection will freeze. The **Blue** button can be used to extend in either direction from the original selection point.

Make half a dozen extended selections, starting each time with a single character (**Red** button) selection and then extending the selection with the **Blue** button.

> [The areas between the To, From, Subject fields and the Date, Location, Organization fields are separated by a tab. When a selection is extended over these areas, a long line indicating the length of the tab will appear--Bravo considers a tab as a character, and when selected, the length of the tab is underlined.]

Select a *word* using the **Yellow** mouse button. After that selection is made, hold down the **Blue** button and extend the selection. Bravo now extends through whole words--even if the cursor only touches the first character of a word, the entire word will be selected. Make half a dozen of these extended word selections until extending becomes a comfortable operation.

With this set of mouse functions, any block of text can be selected in **Form.Memo1**. Once text is selected, a number of editing operations can be applied to alter it. Before doing that, look at the following summary of the mouse functions just described:

Red	Selects a single character or space
Yellow	Selects a word
Blue	Extends single character or word selection

Hints About Using the Mouse

At the outset, operating the mouse may be somewhat awkward. There are several techniques which may be useful:

1. Pick up the mouse and turn it over. With your finger spin the metal ball around a bit and watch the cursor (arrow) on the screen.

2. When you are reaching too far or have to lean forward, pick the mouse up and move it back to a comfortable distance. The cursor will not move when the mouse is lifted from the table, since movement of the metal ball causes the cursor's movement. It will also make the movement smoother to put a thick piece of paper underneath the mouse.

3. Use your index finger to push a button, and hold the mouse rather firmly with your thumb and middle finger *while* the button is pushed--i.e., be holding on to the mouse while depressing a button; otherwise you will experience difficulty getting accurate selections.

4. Hold the button down *while moving the mouse.* The cursor will follow the movement; when the cursor (arrow) is underneath the character or word desired, release the button and the selection will freeze. You may find that this gives a greater sense of control.

D. Create a Memo--Exercise 1

In order to fill out the memo form on your screen with the text desired, Bravo must first be told *where* to make a change (select an "operand") and then what the change is (type a command). This sequence will be used throughout the creation of the memo--make a selection, type a command.

Turn to Exercise 1 and arrange your working area so both Exercise 1 and this part of the Outline are visible. Exercise 1 will be used as copy for the memo being created.

The first two lines on the screen (*in the document window*) read:

Heading:
Subject

The word "Heading" is a non-printing directive to Bravo. It states that Bravo is to put a heading on the second and following pages of the memo. As things now stand the heading on the second and following pages will be "*Subject.*" So the first thing to be changed is the heading.

Replace Command

Select the word *Subject* using the **Yellow** mouse button. Bravo has a command for deleting selected text and replacing it with other text. The command is replace. With *Subject* selected, type r for replace.

Replace first deletes the selected text, then allows *any amount* of text to be typed in its place. The text typed to replace unwanted text will begin at exactly the same point as the deleted text began; the blinking "caret" marks that point and continues to indicate where each character typed will appear.

The second page of Exercise 1 contains the heading "Bravo." So, in place of *Subject*, type Bravo. When Bravo has been typed, hit the ESC Key. The caret will disappear and the next block of text to be edited can be selected. However, while the caret continues to blink, anything typed will appear as text in the document--i.e., no *command* other than ESC or DEL can be given, and no selection can be made while the caret continues to blink.

The command sequence to replace one block of text with another is:

Select text
replace
type the text desired
ESC

Replace the To, From, Subject, Date, Organization categories. In every case, the command will be replace. Using Exercise 1 as a guide, fill out those areas, each time terminating the replace command with an ESC. Then go to the next category to be replaced, select it (**Yellow** button), give the replace command, type the text desired, and terminate with ESC.

E. Typing Paragraphs -- Memobody

The category called "MemoBody" differs slightly from the other categories. Instead of a single word or several words, MemoBody will be replaced with the entire text of the memo. To do that you need to know something about how Bravo formats the text.

Control CR (↑CR)

Unlike typing on a typewriter, *carriage returns are not used to break a line*; Bravo will break, format, and right justify the text automatically. However, *you do need to indicate paragraph breaks* to Bravo. This is done with a *Control* carriage return. Holding down the Control key (abbreviated CTRL on the keyboard) while typing a carriage return indicates a paragraph break to Bravo. If a plain carriage return is accidentally typed, backspace (BS Key) over it and type a ↑CR instead.

Use of the Control key in Bravo differentiates characters that simply print on the screen from those that cause a *specific* result. E.g., the carriage return "character" breaks a line and starts a new one directly underneath. A Control Carriage Return starts a new paragraph by opening *uniform* space between each paragraph and maintaining the same formatting. As you proceed there will be a number of control functions--indenting a paragraph, centering it on the page, etc. Most of Bravo's formatting capabilities are applied to *paragraphs*--a block of text entered without a ↑CR *is not a paragraph; it cannot be individually selected and consequently cannot be altered with Bravo's paragraph formatting features.*

Okay, select MemoBody (**Yellow** mouse button), give the replace command (type r), and start typing the text paragraphs--keep typing text without a carriage return until the end of a paragraph, then hit *one* **Control CR**--*hold down the Control key and while it is depressed, hit the carriage return once.*

Then continue typing the next paragraph. The *second paragraph* of the memo is made up of two separate lines of text, but only one *paragraph.* After the first line of text, type a standard carriage return to break the line; after the second line of text, type a ↑CR to cause a paragraph break. When the entire text insert is finished (all of the paragraphs that make up Exercise 1), indicate that to Bravo by hitting the ESC Key.

[The length of the memo is not a form of torture; it is long so you can get accustomed to the keyboard, and have an opportunity to use a number of Control CR's. By the time this text is typed, you will have a feel for Control CR's; for the blinking caret; for Bravo's justification of the right margin; and for the text moving up on the screen as new text is added.]

F. Storing a File

In the storing operation, the result should be to: 1) retain **Form.Memo1** on the disk in the "blank" state you initially saw it; and 2) put the memo just created on the disk with a *separate* filename. Both results can be produced with a single command.

To initiate the command for storing the file, type:

p

for put, then look at the top line in the system window. It says:

PUT: **Type or select filename terminated by** ESC

A *new filename* needs to be typed for the memo. Call the memo: **Memo1.Bravo.** The period in the middle of the filename serves as a divider between the "name" of the file and the "type" of file it is. Its name is "Memo1" and it is a "Bravo" type of file. So continue the put command by typing:

Memo1.Bravo

The filename typed appears within the broad black border in place of the old filename **Form.Memo1.** Once the new filename is typed correctly [backspace (BS Key) to correct any error], type:

ESC

Bravo will then write the file on your disk: you will hear disk movement; the "document window" will go black; Bravo will flash the message "Writing" in the "system window"; it will indicate the length of the file, verify its filename, and display a new "Ready" sign.

Now both **Memo1.Bravo** and **Form.Memo1** are stored. Both results have occurred simultaneously because:

1. When Bravo was asked to get **Form.Memo1**, a "copy" of the file was displayed on the screen. The "original" has been on the disk throughout the process of typing the memo and can be used again the next time a memo is done. It will be the same form and will be called by the same name.

2. When a new filename (**Memo1.Bravo**) was supplied for the memo, the "copy" of **Form.Memo1** was transformed into an entirely different file which now resides on the disk under a separate filename. The result of this action is that the memo is stored as **Memo1.Bravo**, and the form used to create the memo has not been disturbed--it is on the disk, and is still called **Form.Memo1.**

G. Printing a File - Hardcopy

This section describes printing files on the Ears printer at PARC. If you are using a Diablo type printer, turn to Appendix C.

In order to print a file (document) created in Bravo, you must *be* in Bravo (which you are), Get the file you want to print (you already have it), then give the following command:

h

When **h** is typed Bravo completes the word "Hardcopy" The top line in the system window reads:

HARDCOPY: **Type option (C,D,E,S,DEL) or CR to confirm**

The only "option" of concern at the moment is the "c" option--meaning copies. You want two copies. (Typing a CR at this point would result in *one* copy being printed.) To get multiple copies, type:

c

Now the system message is:

HARDCOPY: **Type # of copies, terminated by ESC**

Type:

2

and terminate the copies option with ESC.

Bravo responds with the "Type option (C,D,E,S,DEL) or CR to confirm" message again, providing an opportunity to use other options. Right now no other option is needed, so type:

CR

After this carriage return, you will hear disk movement, the document window will go black, a single number will print on the screen--1, then 2--reflecting

the number of pages in the document. (The page number printed is attached to the cursor--i.e., a number takes the place of the arrow. If no page numbers are visible, move the cursor until a white number is visible on the black screen.) Finally, Bravo responds with:

READY: Select operand or type command
Last page printed: 2

The Hardcopy command is finished, Bravo is ready to do something else, and it states that a 2 page document has been printed. (It does not confirm the number of copies requested.) The document may be picked up in the Ears Room.

The sequence for the multiple copies option to Hardcopy is:

h	begin Hardcopy command
c	Type multiple "copies" option
2	Type the number of copies you want
ESC	Terminate copies option
CR	Terminate command, send document to printer

Possible Problems with Hardcopy Command

The sequence presented indicates what will happen if Hardcopy is working normally. However, another possible message is: **"Palo is Busy - Trying again"**. If this message is not followed by: **"Transmission Proceeding"** within a minute or so, hit the DEL key, which will terminate the command. Nothing will be printed if the command is aborted with DEL, but waiting a few moments and reissuing the hardcopy command will probably be sufficient to get the document printed normally.

Another message you may get from Bravo is: **"Core space running low. Suggest you P and Q..."**

If this message occurs, it means that there is no more space on the disk to continue carrying out instructions. Put your file on the disk and quit from Bravo; then return immediately, get the file, and reissue the hardcopy command. See the section below on Quitting from Bravo.

Quitting from Bravo

To quit from Bravo, type:

q

for **quit**, and then type a confirming CR.

Quitting from the Alto

Quitting from Bravo returns you to the Alto Executive--remember that is where you started from. To Quit from the Alto (turn it off, in effect), do exactly what you did in Bravo--type:

q

for **quit**, and then type a confirming **CR**.

The screen goes black, the little white squares start dancing around and you can **take a break!**

Throughout Part I there are several points at which a break is suggested. If you do not decide to take the break, you should nevertheless quit from Bravo each time a break is indicated. You may return to Bravo immediately and continue with the next session. Quitting at the points indicated minimizes the chance of a Bravo failure.

II. Review

Up to this point a number of operations and specific commands have been described. A list of all of them follows. Make sure they all make sense before continuing.

Load your Disk

Push Run Switch

Boot the Alto

BS Key (delete one character at a time)

DEL Key (Abort a command)

BravoCR (access Bravo from Alto Executive)

Get **filename** ESC

First Set of Mouse Functions

Red	selects character or space
Yellow	selects word
Blue	extends character or word selection

Replace (delete text and replace it with new text)

Control CR (used for paragraph breaks)

Put **filename** ESC

hardcopy

copies
2
ESC
CR

quit CR (quit from Bravo)

quit CR (quit from the Alto)

Turn to Exercise 2 and arrange your work area so both Exercise 2 and this part of the Outline are visible. Beginning with booting (starting) the Alto, a review of everything covered to this point follows. Proceed through the following list of operations in the order given:

Exercise 2

1. Boot the Alto
2. Access Bravo
3. Get the file **Form.Memo1**

4. Using Exercise 2 for copy fill in the form--select the areas to be replaced, give the replace command, type the appropriate text and terminate by hitting the ESC key.
5. Type the body of the memo (the shorter version provided in Exercise 2) using Control Carriage Returns between paragraphs.
6. Put the memo on your disk. Call it **Review1.Bravo**.
7. Print (hardcopy) the memo, use the copies option to get 2 copies, terminate copies option with ESC, and hit a CR to send file to printer.
8. Quit from Bravo and quit from the Alto.
9. Go pick up the completed document.

Break time.

III. Editing a Document

Turn to Exercise 3. It is a copy of **Memo1.Bravo** with some edits indicated. It has been double spaced to make the edits easier to read, but it is *exactly* the same document. By using the commands already described and adding some new ones, all of the edits indicated can be completed. First, boot the Alto, access Bravo and get the document originally created, **Memo1.Bravo** (Type **g**, the filename, and hit the ESC Key).

One of the things now necessary is the ability to move around in a document that is longer than one page. The *maximum* amount of text on the screen is actually less than a full printed page. Therefore, even with a one page memo it will be necessary to manipulate the text in order to control *which part of the memo* is visible at any one time.

A. Scrolling the Document

You are able to move (scroll) through the document by using the cursor on the screen as a marker, and the mouse buttons to indicate how far, and in which direction, the text is to be scrolled.

When the cursor is moved to the left side of the screen, it changes shape in order to visually indicate, in so far as that is possible, the *kind* of result to be expected from the new *context.* Move the cursor to the far left side of the screen--as far to the left as it will go.

On this side of the screen the cursor becomes a thicker, double arrow, pointing up and down. When the cursor is in this shape it indicates an area on the screen called the Scroll Bar. From here the text can be moved up on the screen, making text toward the end of the file visible; or it can be moved down, making text toward the beginning of the file visible.

Red Button - Scrolling Up

The document **Memo1.Bravo** is displayed on the screen. Because it is double-spaced, very little of the text is visible. To move the bottom line on the screen (whatever that line is) to the top of the screen, proceed with the following steps; *do them slowly, giving yourself plenty of time to observe what happens at each stage*:

1. Position the cursor (arrow) on the far left side of the screen so its shape is that of a double arrow, pointing up *and* down.

2. Bring it down directly opposite the last line of text on your screen, keeping it against the left side--still in the double arrow shape.

3. **Hold down** the **Red** mouse button. While the button is depressed, the arrow is still thicker and points in one direction--up. When the button is released, the line of text opposite the arrow will go to the top of the screen. [*Note: while the mouse button is held down nothing will*

happen; if it moves out of the Scroll Bar area, move it back before releasing the button.]

When text jumps to the top of the screen, you may experience some disorientation. This happens because: 1) the change happens very quickly--too quickly to actually track its movement; and 2) frequently a particular position in the text (the beginning of a paragraph, for example) is more difficult to find immediately after the text is reconfigured. This phenomenon will decrease with frequent use of Bravo.

Go through the entire memo, repeating the above operation: Place the cursor on the left side of the screen, double arrow shape, opposite the last line of text on the screen and push the **Red** mouse button. Each time notice which line is opposite the cursor before pushing the **Red** button, and verify that it is the first line on the screen after releasing the **Red** button. In other words, make sure you can accurately predict what will happen each time the **Red** button is released.

Blue Button - Scrolling Down

Having done numerous upward scrolling operations, you arrive at the end of the document, indicated by a dark triangular-shaped character pointing to the left--an "end of file" marker. Return to the beginning using a similar method, different mouse button.

1. Position the cursor on the far left side of the screen, double arrow shape, as before.

2. Bring it opposite the last line of text on the screen, or, if the file is scrolled so that scarcely any text is visible (end of file marker is very near the top of the screen), place the cursor about 3/4 of the way down the screen.

3. **Hold down** the **Blue** mouse button and notice that while the button is depressed, the arrow is thicker and points down. When the button is released, the line at the top of the screen will move down *opposite the cursor.*

Go through the entire document this way until you arrive at the beginning. Each time this operation is performed, notice which line is at the top of the screen *before* releasing the **Blue** button, and verify that the line has come down opposite the cursor position--i.e., do not move the cursor until the change is complete.

The last line of the file was used as a *reference point* in explaining these two scrolling techniques. You need not use the last line; if a move of only a few lines is wanted, position the cursor at the appropriate height along the left side of the screen.

Yellow Button - Relative Scrolling

The **Yellow** mouse button scrolls to a point in the file relative to the top or bottom of the document window and the beginning or end of the file. That is, with the cursor at the top of the document window in its double arrow shape, and just touching the wide black border, pushing the **Yellow** button will scroll the document to the beginning. With the cursor at the bottom of the document window, just touching the boundary line, pushing the **Yellow** button will scroll the document to the end. With the cursor in the middle of the screen, still double arrow shape, pushing the **Yellow** button will scroll to the middle of the document. Any relative point *between* beginning and end can be accessed by placing the cursor closer to the top of the window or closer to the bottom. Try several of these options.

The **Yellow** button is useful with long files for jumping to a *general vicinity.* It is less useful with this memo since one of the other two scrolling techniques can be used to arrive at a more precise point. However, try this method--the document will grow as the Outline continues.

Be *relatively* comfortable with these three scrolling techniques before continuing--go through the document in each direction several times if necessary. These techniques along with the previously learned mouse functions will be used to complete Exercise 3.

Exercise 3

The first eight edits in Exercise 3 can be done with the replace command. However, it may be necessary to scroll the text to access the edits.

> *[Note: While editing the document, text which was visible before a command was given, may disappear the instant you type r, for example. Depending on the operation performed, Bravo uses some of the space allocated to the screen. When it needs more space, it takes it, meaning that some of the text may temporarily disappear. Scrolling up a few lines will make it reappear, and the document can be scrolled in the middle of a command. So, if this happens, move the document up a few lines until the area you were working on reappears.]*

B. Delete Text

Beginning with the ninth edit, "yet" needs to be deleted, but not replaced with anything. The delete command is activated by typing d on the keyboard. [Do not confuse typing d on the keyboard to delete *text* with the DEL key to abort a *command.* They are entirely different operations, not at all interchangeable.]

When d is typed, the selected text disappears immediately. The delete command does not require an ESC to complete it.

When a single word is deleted from between two surrounding words, two spaces will be left between the words remaining. After a word is deleted, the space *immediately following* it is underlined (selected). Therefore, giving another delete command will delete the extra space.

To delete both the word and the extra space with one command, make a **Red** and **Blue** selection (select the *space in front* of the word with **Red** and extend the selection through *the last character* in the word with **Blue**) before giving the delete command.

C. Insert Text

The insert command allows text to be inserted *in front* of the selected text. If you select a space between two words, type an i for insert, the blinking caret will appear just *in front* of the space, and the text to be inserted should *begin* with a space. If you select the wŏrd itself and type i for insert, inserted text should *end* with a space.

If in doubt as to where inserted text will appear, type the first character of the insert and watch where it is placed. If, after seeing that point, it turns out a space is needed, BS (backspace) over the character and start again with a space. Use the ease with which errors can be corrected to your best advantage--if typing a character will provide visual context, do it!--it can always be erased.

Once all text to be inserted is typed, terminate the command with an ESC. The sequence for the insert command is:

> Select the character or word you want to insert text *in front* of.
> i (for insert)
> Type the text to be inserted; the blinking caret indicates where it will go.
> ESC key to terminate the command.

D. Append Text

The append command works *exactly* like insert, except that text *is added after* the text selected, and the command sequence is the same, except that an a is typed instead of an i. Only a is typed; Bravo completes the word Append. The edits that require additional text can be done using either the insert or append command.

E. Put a File Using the ESC Option

With all of the edits finished, put this *new version* of **Memo1.Bravo** on the disk. When changes are made to a file, a new version of the document must be put on the disk with the edits included. However, this time, it is a new version, not a new file, and *the filename need not be retyped.* Instead, type:

p for Put

and then type:

ESC

Bravo will assume from this command that an updated version of **Memo1.Bravo**, with the edits included, is to be put on the disk.

Using ESC in this fashion introduces a new concept about it. In the command above (**p**ut, ESC) the computer is able to respond appropriately because it *already has* the information it needs. Use of ESC in this context signals it to use the information it has since no new information is to be provided.

Having typed **p** followed by ESC, this version of the document will be put on the disk. Now print 2 copies of it.

> *[Incidentally, always Put an edited document* before *sending for Hardcopy. If anything "untoward" happens in the Hardcopy process, you are in no jeopardy of losing the document: a permanent copy resides on the disk, and if need be the hardcopy command can be reissued.]*

Review of Hardcopy Command

The command sequence for multiple copies of a file is:

h
c
2
ESC
CR

Break time.

IV. Formatting

You have created a file, put it on your disk, printed it, scrolled through it, and edited its contents. The next two sections describe: use of boldface, italics and underlining for emphasis; adding *paragraphs* to the end of the document; and changing the way paragraphs look--whether they are centered, indented, or right justified.

A. Boldface, Italics, Underlining--Look Command

Character *looks* can be changed with a look command, followed by indicating the look option desired. Both the look command and the look options are single character commands. The three options for this section are: b for boldface, i for italics, and - for underlining (hyphen for underlining).

Exercise 4

Turn to Exercise 4, and arrange it so both Exercise 4 and this part of the outline are visible.

Get the file **Memo1.Bravo** and select the subject: "Bravo Introduction." Activate the look command by typing l. The top line in the system window now reads:

LOOK: Type Look (B,I,-,↑,0-9,C,J,N,R,L,F,P,X,Y,?,...)

Bravo confirms the look command, requests that you type a look, and lists the options--each character within the parentheses represents a possible look for Bravo. The first edit in this exercise indicates boldface. Type b and the selected text changes to boldface. The ESC key is not needed for this command. The sequence for these look commands is:

Select text
look
b (or i or -)

The next look indicated is italic. To get characters italicized perform the same command, but type i for italics. The three looks are b for bold; i for italics; and - (hyphen) for underlining. Go through the document changing the looks as indicated. When you have finished, do a put ESC, and then hardcopy the new version of **Memo1.Bravo**.

B. Third Set of Mouse Functions--Line Bar

This section begins with more mouse functions, describes how to return special character looks to normal, then text gets added to the memo in order to cover special character looks *while* typing.

Line Selection

Move the cursor over toward the left side of the screen (slowly). Watch it change from a left-ward pointing arrow to a right-ward pointing arrow. When the cursor is just to the right of where it becomes the up and down pointing arrow for scrolling, the arrow points toward the right. With the cursor in this position, push the **Red** mouse button. The line of text opposite the cursor on the screen will be selected. Make half a dozen of these line selections.

Select a single line of text with the **Red** button. Then *hold down* the **Blue** button and move the cursor down a few lines. Notice that the selection is extended through additional *lines.* Make half a dozen line selections, followed by extending the selection until the combination is comfortable.

Paragraph Selection

Now position the cursor next to a paragraph (it does not matter which line in the paragraph) and push the **Yellow** mouse button. The cursor changes its shape to a "paragraph" symbol, and the entire paragraph is selected. *Hold down* the **Blue** button and move the cursor down the left margin. As the cursor encounters a new paragraph, it also is selected. Make half a dozen paragraph selections, followed by extending the selection to other paragraphs until the combination is comfortable.

These selection techniques will be used in the editing and text entry exercise that follows.

*[One likely difficulty is that the cursor will move to the left or right as a mouse button is released. The area to the left of this "Line Bar" area is for scrolling, and the area to the right is for selection of text. Therefore, occasionally when you want to select a line (*Red *button), you may actually scroll the document "up" several lines, or select a single character within the text since the Red button performs both of these functions depending on its location (context) when the button is released. This problem is more serious with the Yellow button because releasing it in the scroll bar area when a paragraph selection is wanted, may scroll the document rather dramatically--the file may be scrolled to the beginning when only a selection was wanted near the end of the file. If this happens, start the selection again, scrolling the document if necessary. This exercise really does test your ability to hold the mouse steady.]*

Exercise 5

Turn to Exercise 5 and arrange your work area so both Exercise 5 and this part of the Outline are visible.

The first two edits require word selection (cursor under the word, **Yellow** button). To change special character looks back to normal, use the look command, followed by hitting the CLR Key. So, the sequence is:

Select text
look
CLR

Select the word "two" which is italicized. Then type I for look, and hit the CLR key. In all cases where the edits indicate cancelling special character looks, this is the process; the only difference will be in the *amount* of text selected and the *way* it is selected.

The third edit indicated requires that two lines of text be changed from bold to normal. Select the first line by placing the cursor in the left margin (within the Line Bar), arrow pointing toward the right and push the **Red** button; then move the cursor down to the next line (arrow still pointing to the right) and push the **Blue** button. The selection should now include both lines of text. Now give a look CLR command. [The two lines constitute a paragraph and could be selected as a paragraph--one click of the **Yellow** button.]

[Another way to clear character looks is by using the look command followed by typing the uppercase character for the look being changed--lookShiftB to clear boldface; lookShiftI to clear italics; lookShift- to clear underlining.

Using the CLR key clears *all* looks on a character. The shift key method clears only the look specified. For the looks described in this exercise the lookCLR method can be used, though it might be wise to try a couple of the shift key clears--as you continue to use Bravo, this method of clearing looks will sometimes be more appropriate.]

Continue through **Memo1.Bravo** making all the "clear" edits indicated. The end of your document will coincide exactly with the end of the *double-spaced portion* of Exercise 5.

Character Looks While Typing

The single-spaced paragraphs, beginning with: **C. First Set of Mouse Functions** constitute the text to be added to the end of the memo. As text is typed, "Control" characters will be used to make portions of the text underlined, italicized or bold.

To begin typing this additional text, select the last paragraph of the memo *as a paragraph*--cursor on the left margin, arrow pointing to the right, push the **Yellow** mouse button. The cursor will indicate a paragraph selection by changing its shape to a paragraph symbol. With the paragraph selected, give an append command. That is, type:

a

Bravo opens a new paragraph block, and the blinking caret appears down the page a bit--the distance of a standard paragraph break, marking the point at which new text will appear.

[If, when the append command was given, the selected text disappeared, scroll up a bit until the blinking caret reappears below the paragraph being appended to.]

The beginning of the text contains special character looks. The two principles to follow in typing special character looks are:

1. When text is to appear in boldface, for example, *hold down* the Control Key *while* b is typed. Nothing will happen on the screen at this point, but the text following ↑b, will appear in boldface.

2. When boldface is finished, hit the CLR Key and bold will be "cleared" or turned off. Then continue typing normally until another special character look is required.

The three special character requirements for this additional text are illustrated in the following chart ("↑" means the Control Key--i.e., hold down the Control Key while you type the character following ↑):

Begin	Character Look	End
↑-	Underlining	CLR Key or ↑Shift-
↑i	Italics	CLR Key or ↑ShiftI
↑b	Bold	CLR Key or ↑ShiftB

Referring to this chart, type the text using Control character and CLR Key combinations to get the special character looks indicated in the text. Remember to type a Control CR *between* the paragraphs.

Take your time with these operations; be sure to get a feel for how special character looks work while typing.

If you forget to hit the CLR Key and have more italicized, bold or underlined characters than needed, backspace (BS Key) to erase the extra characters, hit the CLR Key, then resume typing text.

When all of the additional text has been typed, terminate the append command by hitting the ESC key.

[Note: When appending, inserting, or replacing a paragraph, Bravo opens an entire paragraph block; therefore no Control CR is needed after the last *paragraph--type Control CR's* between *the paragraphs, but at the end of the* final *paragraph, type only an* ESC. *Bravo will automatically make the last block of text a paragraph. If a ↑CR* is *typed after the last paragraph it will result in a blank paragraph at the end of the file. If several blank paragraphs end the file, they may be sufficient to cause a blank page to print. They can be selected and deleted, but you cannot BS (backspace) over a ↑CR. When selected a ↑CR looks like a space,*

and it is the last character in each paragraph.]

Put the file on your disk and continue to call it **Memo1.Bravo** (Put ESC). Then get two copies of the document printed.

Break time.

V. Review

A list of the operations covered *since the last review* follows, and Exercise 6 provides edits to be used in reviewing these operations.

For section III on Editing a Document:

Scroll Bar Mouse Functions

Red	Sends line opposite cursor to top of screen.
Blue	Brings top line of text down opposite cursor position.
Yellow	Scrolls to beginning, middle or end point in document, relative to cursor placement at top, middle or bottom of document window.

Replace Command

Deletes selected text and allows new text to be typed in its place.
Terminated by ESC.

Delete Command

Deletes selected text.
No ESC required.

Insert Command

Opens an area *in front* of selected text where new text can be inserted.
Terminated by ESC.

Append Command

Opens an area *after* selected text where new text can be appended (added).
Terminated by ESC.

Put ESC

Puts a file on the disk; uses *the same filename.*

Hardcopy Command

Multiple copy option:

h
c
2 (or 200)
ESC
CR

Single copy option:

h
CR

For section IV on Formatting:

Look:

b, i, or - bold, italics, or underlining, respectively.

Line Bar Mouse Functions:

Red - Selects line opposite cursor
Yellow - Selects paragraph opposite cursor
Blue - Extends line or paragraph selection

Look CLR

To return special character looks to normal

Paragraph Append

With paragraph selected, opens a paragraph block *after* selected paragraph where one or more paragraphs can be added.
Terminated by ESC.
[No *final* Control ↑CR is necessary]

Paragraph Insert

With paragraph selected, opens a paragraph block *in front* of selected paragraph where one or more paragraphs can be inserted.
Terminated by ESC.
[No *final* Control ↑CR is necessary]

Paragraph Replace

Deletes the text of selected paragraph and maintains an open paragraph block where one or more paragraphs can be typed.
Terminated by ESC.
[No *final* Control ↑CR is necessary]

Special Character Looks During Typing:

Begin	Character Look	End
↑-	Underlining	CLR Key or ↑Shift-
↑i	Italics	CLR Key or ↑ShiftI
↑b	bold	CLR Key or ↑ShiftB

Exercise 6

There are two operations in this review that have not *actually been used*: 1) Text has been inserted, but *paragraphs* have not been inserted; and 2) text has been replaced, but *paragraphs* have not been replaced. Both of these operations are covered in the review.

Turn to Exercise 6 and arrange your work area so both Exercise 6 and this part of the Outline are visible. Then get the file **Memo1.Bravo**.

The steps to perform in editing Exercise 6 are listed below, and their numbers correspond to the numbered edits indicated on your copy of Exercise 6. Refer to this guide if you choose. If there is some confusion about *which* command to use, it will indicate the quickest and easiest command for that edit. You may, in any case, want to refer to it when a paragraph is inserted and when a paragraph is replaced.

1. Select "little," give the replace command, type "small" and terminate with ESC.

2. Select "system" and type l i (look italic).

3. Select " and type d for delete, select second " and type d for delete.

4. Select "O" and replace it with "o", terminate with ESC--same procedure for "T" and "C".

5. Use one of the following techniques:

   Select "line" and append "in the system window" (the appended text should begin with a space).

   Select "states" and insert "in the system window" (the inserted text should end with a space).

   Select the space between "line" and "states" and either insert or append "in the system window" (text typed will either begin or end with a space, depending on the command used.)

6. Select " and type d for delete; select the second " and type d for delete; then select "Ready" and type lb (look bold).

7. Select "two" and type l i (look italicized).

8. Select "operand" and type l CLR Key (look CLR).

9. First select "command" and type l CLR; then extend the selection *back* through "type" and give a look - command.

10. Use any of the options described in 5 above.

11. Select "Get" and type l b (look bold).

12. Make a line *or* paragraph selection of the text and type look CLR.

13. See 5 above; use any of the options described there for inserting or appending text.

14. Select the entire line using the **Red** mouse button, then type l CLR.

15. Select "T" and replace it with "When t".

16. Select "you have" and replace it with "is"; then select typed and append: , it.

17. Select the paragraph which now begins: "When the filename is..." and give an insert command; then type the paragraph of text to be inserted. No ↑CR is required at the end of the paragraph of text.

18. Select the paragraph beginning "Bravo says..." and give an append command. Then type the text of the paragraph to be added. No ↑CR is required at the end of the paragraph of text.

19. Use either of these techniques:

    Select "the" and type d. There will be two spaces between the remaining words. One of the spaces will be selected; use a second d command to delete it.

Select "the" *and* one space on either side of the word before issuing the d command.

20. Select the paragraph beginning "A selection may be..." and replace it with the two paragraphs given. A ↑CR is used between the two paragraphs that constitute the replacement text, but no ↑CR is required at the end of the last (second) paragraph typed.

VI. More Formatting

Up to this point a document has been created, edited, put on the disk, printed, and emphasis has been provided using Bravo's facilities for changing character looks. In this section altering the *format* of paragraphs is discussed.

A. Paragraph Looks - Editing Existing Text

Bravo has a number of formatting options for specifying paragraph *style*. This section introduces several of the more straightforward means; later sections in the Outline discuss more complicated formatting options.

The formatting commands discussed below are paragraph "looks"--that is, a paragraph is selected and a look command applied to it. All of the commands discussed will apply to *entire paragraphs.*

For most forms on the disk, the standard is right justification. The memo form on your disk, for example, automatically provides right justification of text. However, text can be unjustified; it can be centered on the page, and paragraphs may be indented from the left margin.

Exercise 7

Turn to Exercise 7 and arrange your work area so both Exercise 7 and this part of the outline are visible. If it is not already on the screen, get the file **Memo1.Bravo** which contains the same text as Exercise 7. The text of this exercise incorporates the last set of edits, and it is double spaced throughout.

All of the edits indicated can be performed using a paragraph look command, followed by one of three options--c for centered; n for nested (indented 1/2 inch); or ShiftJ for unjustified (because the standard using Form.Memo1 is justification, you will first learn to unjustify, then in the next exercise learn to justify text). [*Note that nesting is a preset indent of 1/2", and several levels of nesting are possible--i.e. nest a paragraph twice to get it indented 1 inch, or three times to indent it 1-1/2 inches, etc.]* The sequence of the command in all three cases is the same:

Select the paragraph to be changed
l (look)
c for centered, ShiftJ for unjustified, n for nested

No ESC key is required to activate these commands; they take effect immediately.

All of the edits in this Exercise are paragraph looks. Either select an entire paragraph with the **Yellow** button, cursor pointing toward the right, or select some part of the text *within* the paragraph. In either case, the paragraph command given will affect the entire paragraph. Try a few of these commands with a word (or single character) within the paragraph selected.

The first paragraph should have justification cancelled. The command is:

lookShiftJ (look unjustified)

The second paragraph needs to be centered. The command is:

lc (look centered)

The fourth paragraph needs to be nested (indented 1/2 inch). The command is:

ln (look nested)

Go through the remainder of the edits, each time selecting a paragraph or some part of a paragraph and issuing the look command followed by the character for the edit indicated--**c**, ShiftJ, or **n**. When all edits are finished, do a put ESC, and print the file if you wish.

Break time.

B. Editing Paragraph Looks - *Exercise 8*

Turn to Exercise 8 and arrange your work area so both Exercise 8 and this part of the outline are visible.

Exercise 8 covers both editing existing paragraph looks, and centering, justifying, and nesting paragraphs *while typing text.*

Centering is obtained by typing lookc to a selected paragraph. Centering is cancelled by typing lookShiftC to a selected paragraph. Notice the only difference is that the "c" is capitalized to cancel centering and lowercase to initiate it. Justification and nesting work the same way--lookj to selected paragraph(s) justifies, and lookShiftJ cancels justification; to nest (indent 1/2 inch) a selected paragraph(s) type lookn; to cancel nesting type lookShiftN. The following chart lists both the method for beginning a paragraph look and for cancelling it:

Begin	Para. Look	Cancel
look **c**	center	lookShiftC
look **j**	justify	lookShiftJ
look **n**	nest	lookShiftN

The second edit requires *two* paragraph commands: first change the paragraph from centered to not centered--lC is the command. Then for nesting, type ln. (*Note the distinction between upper and lowercase characters in these two commands.*)

By referring to the chart and rereading the descriptive material above, perform all the edits indicated in Exercise 8.

When the last paragraph on your version of **Memo1.Bravo** has been edited, begin to read about the second part of Exercise 8, beginning with the single spaced text on page 3.

C. Paragraph Looks While Typing Text - *Exercise 8*

The paragraphs to be typed are formatted with centering, nesting, and with or without justification. All of these format changes can be obtained *while typing.*

This part of the Exercise requires that additional text be appended to the end of the memo. To do that select the last paragraph in the memo (select it *as a paragraph*), type a for append and Bravo will open a new paragraph block for you to type into--it is the same operation performed in earlier exercises when additional text was added.

The first paragraph is standard text and need only be typed--no formatting commands are necessary. The second paragraph needs to be centered. After typing a Control CR to end the first paragraph, the blinking caret will be on the left margin. Type ↑c and watch the caret move to the middle of the screen. Now type the centered heading (making it boldface) and terminate the line with a Control CR.

The blinking caret remains in the middle of the screen as if to center the next paragraph. Paragraphs maintain the format of the preceding paragraph until commands are given to alter the format, then the new format remains in effect until changed. At this point centering is no longer required. To cancel centering type ↑ShiftC--that is, hold down *both* the control key and the shift key while C is typed. The caret will return to the left margin, and the text paragraph can be typed normally.

After a paragraph has been centered, the paragraph(s) following *will not be justified.* Centering cancels justification, and a ↑j must be typed at some point in the text of the following paragraph in order to restore justification.

The chart below lists the commands for obtaining the paragraph looks required to complete the remaining text. The up arrow indicates that the character following ↑ is a control character--hold down the control character while the character following ↑ is typed.

Begin	Para. Look	Cancel
↑c	centered	**↑ShiftC**
↑j	justified	**↑ShiftJ**
↑n	nested	**↑ShiftN**

Refer to the chart above and reread any of the descriptive material necessary while doing this exercise. When the additional text is finished, do a put ESC, and print the file if you wish. *Break time.*

VII. Miscellaneous Features

The following features may be helpful in a number of applications. No exercise accompanies the description; however, use the Memo1.Bravo file for practicing these features.

A. ESC to Repeat a Command

The ESC key can be used to repeat the effects of the last command. So, the command ispaceESC, to insert a space between two words that are run together, can easily be repeated if there are several instances of the same edit. Change the selection for the second instance and type only ESC. Bravo repeats the entire command: ispaceESC. ESC to repeat the last command works with *most* Bravo commands. Remember, though, that it only pertains to the *last* command given, and it will be executed *exactly* as the command was given the first time. Use this feature with several kinds of edits to get a feel for its usefulness. Insert a string of text, then select another point in the document and type ESC. When the same text is to be inserted in two separate places, use ESC to repeat the insert, rather than retyping the text.

B. Control W--Word Erase

The BS key has been used to correct typing errors. Control w erases words. Each time ↑w is typed (hold down the Control key while typing w), one word will be erased. Try using it to erase a few words. Bravo counts a word as everything between two spaces. Therefore if the typing error being corrected is lack of a space between two words, and ↑w is typed to erase only the second word, Bravo will erase both words since there is no space between them to indicate two words.

C. Undo Command

The Undo command, issued when Bravo *is in a "Ready" state,* reverses the effects of the previous command. It will *only work on the previous command* and only *if the selection has not been changed.* So, if you give a look bold command, for example, and decide boldface is not desirable, type u for undo, and the text will appear just as it did before the look bold command was given. The Undo command works with "most" Bravo commands, but only for the *last* command given and only if the selection has not been changed. Try a few of them. An Undo can also be undone, which will restore the results of the initial command--issue a command, then do several undo's; you will *see* how it works.

D. Everything

To quickly select the entire document, type:

e

for everything. Any command given following an everything command will be applied to the entire document.

E. Window Clear

Occasionally, you will want to work on one document, print it and then work on another document. It is not necessary to leave Bravo and return in order to switch documents, but you cannot have two separate documents in *one* document window.

To *clear* the document window of text and simultaneously clear the filename from the black border, type:

wc

for window clear. You should put the file on the disk before issuing this command. However, if the file has not been put, Bravo will ask for a confirming CR before honoring the command. At that point you may DEL out of the window clear command and put the file on the disk. When the document window has been cleared, you may issue a get command to access the next document.

[Note: if the editing session has been a lengthy one (an hour or more), quit from Bravo and return before accessing another document.

F. Normalize Command

The normalize command locates and displays selected text in the document. When selected text disappears as a result of a scrolling operation, a quick way to get it back is with n for normalize. Typing n when Bravo is in a "Ready" state will cause the first line of selected text to appear at the top of the document window. If a word was selected, the line with that word in it will appear at the top of the screen. This command will only work when Bravo is in a "Ready" state--waiting for a command. *It cannot be used to make text visible if it has disappeared as a result of a command you initiated and are still in the middle of.*

Everything Normalize (en)

Combining the everything and normalize commands will result in the document being scrolled to the beginning. Normalize puts the first line of *selected material* at the top of the screen; if the entire document is selected, the first line will be placed at the top of the screen. This command only works with Bravo in a "Ready" state.

G. Upper/Lowercase Character Edits

Occasionally, a group of characters need to be changed either to upper or lowercase. The replace command is tedious for more than one instance. A quick way to make this edit is to select the first character or group of characters to be capitalized. Give a look> command--look greater-than. This will make the character(s) uppercase. Then select the next character(s) and type ESC. ESC, remember repeats the previous command. Numerous changes can be made with very few keystrokes. It works exactly the same way with look<--look less than, but selected characters will be changed from upper to lowercase.

H. Manual Page Breaks and Keeping Text Together

Bravo fills up a page and breaks the page whenever the bottom margin is encountered. Consequently, the page may be broken inappropriately. When that happens, either a manual page break or a command to keep blocks of text together may be needed. When a heading appears at the bottom of the page alone, and the text following begins on the next page, a command to keep heading and text together is appropriate. When the next block of text begins a new section--appendices, bibliography, references, etc.--a manual page break is appropriate. It is best to use manual page breaks only at points where a new page is *always* wanted, and keep commands for blocks of text which need not start a new page *unless* a new page is required to keep them together. Both options are available with the following two commands.

Manual Page Break

In order to break a page at some particular point, a control character is entered into the text which is interpreted as "page break" by Bravo. It should be appended to the last string of text you want to appear on the page. It is entered into the document by typing:

l↑s

Typing a Control S makes the preceding character a control character, and Bravo reads a ↑L as a page break. The resulting character in the file looks like an "I". It will be visible in the document while editing, but will not print when the file is sent to Ears or the Diablo printer. It can be selected and deleted like any other character. This ↑L breaks the page *after* the line of text in which ↑L appears. If it is typed at the end of a line of text, that line will remain on the current page, the *next* line will begin a new page.

Keeping Text Together

A document with subject or section headings will require that several lines of text be kept together in order to avoid having a heading appear at the bottom of the page alone.

To keep lines of text together, a look keep command is used, followed by a distance for the amount of text to be grouped. For text *with a heading* this distance is 80 points; for *text only* paragraphs, the distance is 40 points. To set an 80-point keep on a heading paragraph, *select the paragraph containing that heading*, and give the following command:

look
keep
80
Esc

No change will be visible in the paragraph following this command, but it will go with the document and be honored when it is printed. If the paragraph is a text only paragraph, but it is breaking with only one line of text at the bottom of the page, the command is:

look
keep
40
Esc

A look keep 80 set on a heading paragraph means that *at least two lines of text* will be kept with the heading on one page, or the entire block (heading and text paragraph) will be sent to the following page. A look keep 40 means that at least two lines of a text paragraph will print at the bottom of the page or the entire paragraph will be sent to the following page.

For the moment, do not worry about what points are: the 80/40 distances given will generally suffice for heading and text paragraphs, or text only paragraphs.

VIII. BravoBug

While you are working in Bravo, a copy or *transcript* is kept of each command given. This transcript can be replayed and stopped at a desired point. If Bravo crashes in the middle of an operation, you may return to Bravo, (entering with BravoBugCR instead of entering normally with BravoCR), and replay the session up to the point where the failure occurred. So, if Bravo fails all is not lost--the work can usually be recovered.

A number of circumstances might cause Bravo to crash: a "bug" in the program which the right combination of operations triggers; the Alto is accidentally booted; or there is a power failure. In any of these cases, the Bravo session may be replayed and the file saved. Running BravoBug saves the results of the *current* editing session; assuming the file has been put on the disk at least once, the file itself is not in jeopardy, only the results of the editing session that was interrupted.

If bravo crashes due to a program bug, it will go into a system called SWAT. This is the most frequent kind of Bravo failure, and it provides some information about the crash, the most interesting part of which is the command number at which the program failed. Make a note of this number before beginning the replay. SWAT's message reads:

Bravo problem at command #--

With that number, the session can be replayed up to, but not including the command at which it failed. Then the replay can be terminated and the file put on your disk, thus saving the work done during the session. The sequence of operations and commands are:

1. Bravo crashes and the top of the screen identifies a program called SWAT. Make a note of the command # it says Bravo crashed at, then boot the Alto.

2. At the Alto Executive level, type:

   Bravobug

   and then a CR.

   (Be sure to type Bravobug, not Bravo--if you enter with BravoCR, the opportunity to replay the session will have been lost.)

3. There is about a 30 second delay before Bravo begins the replay. Then it gets the file and begins to replay each of the commands given during your last Bravo session. Once the file has been read in and you see that the commands are being replayed,

4. Hit the space bar once. This will stop the replay.

5. Now Bravo can be told where the replay is to be stopped. The command for doing this is "break before command #." To activate it, type:

b

The top line in the system window now reads:

REPLAY: Type break command # terminated by ESC

Type the SWAT command # and terminate with ESC.

6. After the command # and ESC are typed, the message in the system window reads:

REPLAY: Type blank, P,T,Q,S

The options are blank (space-bar) to replay a single command, P for proceed, T for terminate, Q for quick, and S for slow. Bravo has been given a command number to stop at; now it should "proceed" to that command. Type:

p

for "proceed." Bravo will continue the replay up to, but not including, the command # given, then will stop and say:

REPLAY: Stopped before Command #--

7. Now the replay can be terminated by typing:

terminate

followed by a CR.

8. Once this is done, you will get a "Ready" message from Bravo. *Put the file on your disk, quit from Bravo, reenter Bravo from the Executive level (this time with BravoCR), get the file and continue the editing session.*

Leaving Bravo and reentering frees up space on the disk and begins a "new" transcript. Giving Bravo ample disk space minimizes failures.

When no command number is available from SWAT, or the number given is incorrect, use the following commands to control the replay:

space	Stops the replay. After the replay is stopped, hitting another space will replay *one* command. Hitting it again, will replay one more command, etc.
proceed	Resumes the replay at full speed.
break before command #	Even without a command number from SWAT this break feature can be useful, but you will have to guess how many commands were given by mentally replaying the session. For example, give it a break # which is ten commands down the line, then reevaluate the

situation and give another break point. Each time Bravo stops the replay, it will tell you at which command number it has stopped. Remember that Bravo must be told to proceed (type **p**) after the **b**reak # is given.

terminate**CR** — Do not terminate the replay until no more commands need to be replayed. Once the replay is terminated, **p**ut the file on your disk, leave Bravo and return normally to continue the work that was interrupted.

slow — This command will slow down the replay. The text characters go in one at a time. It makes for a very slow replay. You can speed it up again (quick is its standard pace) by typing,

quick — To make the text go in all at once.

The last two commands listed here--slow and quick--you may or may not need. Quick is never needed unless slow has been used since quick is Bravo's standard speed.

Make a point of doing several replays--boot the Alto in the middle of an editing session if need be--so that an interruption with a document you care about does not cause a panic. In fact, *now* is the best time to do that. Practice some of the miscellaneous features described in Section VII, then boot the Alto in the middle of the session and run **BravoBug**--there may be few times when you have less to lose.

IX. Alto File Directory

Before reading this section, quit from Bravo and return to the Alto Executive level where all of the directory operations described in this section are performed.

Files on your disk are kept in a directory which can be interrogated. There are several conventions for doing that, and there are also conventions for naming files.

To get your Alto Directory displayed on the screen hit the TAB key. Do that now.

The top third of the screen displays a series of file names. They proceed in alphabetical order from left to right *across* the screen, *not* from top to bottom. At the bottom of this list there is a question:

More?

Type **y** to see more of the filenames; type **n** to discontinue the directory command. Type **y** a couple of times until there are no more files to look at.

In this form your directory is somewhat difficult to decipher because all system files are lumped with your own files. There are several methods for separating your files from system files.

A. * Option

The Alto file structure is: **name.extension.** Note the dot (period) between name and extension; it *must* be typed. Your memo is called: Memo1 and its extension is Bravo. The whole filename, then, is **Memo1.Bravo.**

You can specify a particular *set* of files by using a * to take the place of either the name or the extension. Type:

***.Bravo**

and then hit the TAB key. A list of every file that ends with .Bravo will be displayed on the screen.

There is a limit of 39 characters for a filename. Abbreviated filenames are useful both because of this limit, and because typing long filenames each time you get or put a file in Bravo is time consuming. However, when the abbreviations are so clever and descriptive that they cannot be remembered a week later, this * option will help track them down.

B. Naming Conventions

Grouping your files with one of several naming conventions will facilitate accessing them quickly.

If .Bravo is used as an extension, all .Bravo files can be accessed with the *.Bravo option. If further differentiation is needed extensions can be used to indicate something about the kind of document it is. That is, use .Memo for all the memoranda created, .Letter for all letters, .Report for all reports, etc. Then when you need to see that group of files on your directory, either as a reminder of a filename or for some other operation, the * option described above can be used to get only that group of files displayed--*.BravoTAB; *.MemoTAB; *.LetterTAB; *.ReportTAB, etc.

Bravo accepts only the *exact* filename when you ask it to get a file. A discrepancy of only one character will cause it to print a message to the effect that it cannot find the file. If files are grouped by extension, it will be easy to interrogate your directory (at the Alto Executive level) and quickly verify the name of the file.

C. Deleting Files from Your Directory

Files that are no longer useful--that is, unlikely ever to be needed again--can be deleted from the disk. Realize, however, that once all versions are deleted the file is gone forever; there is no way to get it back.

There will be no immediate need to delete files from your disk. Until quite a few have been created, or your disk begins to get full, do not worry about deleting. Once you decide to delete files, the command is given while in the Alto Executive. It is:

>del filenameCR

Type **del** (the characters d e l), *then a space*, then the **filename** to be deleted, then a CR to activate the command. The filename will be listed and deletion verified.

If you did the first *review* in this outline, it can be deleted in order to try this command--it was called **Review1.Bravo**. To delete it, type:

>del Review1.BravoCR

When "cleaning up" your disk becomes a regular occurrence, look at the *Alto User's Handbook* and read the manual for the system called DDS, which has facilities for both interrogating your directory and deleting files from it. In the early stages of your Alto/Bravo experience, the naming conventions described above and this means of deleting files will suffice.

D. ESC, CR, and TAB--Where, When, and What Happens

The commands requiring one of the above terminators have been listed, but it may be somewhat confusing to keep in mind where and under what circumstances CR is used instead of ESC, or TAB instead of either CR or ESC.

ESC

The Escape key is most frequently used for operations in Bravo. Most of the commands are terminated with ESC. However, there are exceptions:

quitting from Bravo requires a CR instead of ESC;
sending for hardcopy requires CR instead of ESC;
terminating a Replay requires CR.

You may find other exceptions to the rule; this is not intended to be an exhaustive list. When in doubt about which one to use, look at the top line in the system window. Bravo generally displays which response it expects to hear.

CR

Uses for CR in Bravo have been mentioned above. CR is used in the Alto Executive for accessing systems like Bravo--**BravoCR**-- and for activating a command to delete a file from the disk--**del filenameCR**.

TAB

The TAB key *as an activator* is used solely in the Alto Executive and then only in connection with getting your directory of files displayed on the screen. The TAB key is used to get your entire directory displayed, and the ***.Bravo** option is terminated with a TAB. It is used in Bravo to set tabs, and to type tab characters into text, features not covered until Part II.

E. Other "Forms" on Your Disk

There are a number of forms on your disk like the form used for doing a memo. Use one of these forms whenever possible to get your work done. Use of forms makes formatting easier and provides a consistency of format which is harder to achieve if a file is started "from scratch."

In order to see which forms are on the disk, use a derivative of the * option. All of the forms on your disk are called Form, and then followed by .memo, or .letter, or .SeminarNotice, etc. Since there is one constant (Form), all instances of something called Form can be accessed by typing:

Form.*

followed by hitting the TAB key. A list of all the forms on your disk will be displayed, and the *extension* will make it clear what the form is used for.

All forms work much like the memo form. The blanks are filled in using the replace command, paragraphs are separated with Control Carriage Returns, and formatting commands may be used while typing or editing.

The letter form contains a note where the body of the letter will appear explaining that a look keep 175 is attached to the paragraph. This is the same principle as the look keep 80/40 option discussed for keeping text together, and it is there in order to keep the body of the letter together with the signature area following it. However, if the letter is long, this 175 keep distance will have to be changed. It should appear only on the *last* paragraph of the letter. First delete it from the *first* paragraph with look keep 0 Esc; then put it on the *last* paragraph with look keep 175 Esc--in both cases select the text *as a paragraph* before issuing the command.

The other forms on your disk are *more or less* decipherable. However, both Form.Report and Form.TableofContents will be quite confusing until later sections in this Outline have been read and the features used.

If there *is* some confusion about one of the forms, do not hesitate to ask questions of fellow Bravo users. Getting quick verbal answers from people who have already used a facility will save time, and you will find that they are happy to assist you.

X. Caveats/Miscellaneous Information

A. Caveats

Bravo responds to single character commands which eliminate typing out entire words; however, it also means that unwanted commands can be activated easily--in fact, it is almost impossible to use Bravo without doing this. There are a great many single character commands in Bravo, many of which have not yet been discussed. *When* a command you do not want or do not recognize has been activated, hit the DEL Key to abort the command. If the command activated is one that takes effect immediately (like delete), give an undo command to reverse the effect of the command.

The time to use DEL or undo is *immediately after a command has been activated.* That means paying close attention to messages displayed in the system window. When a command is activated, Bravo displays a message confirming the command. Once past the point of activating the command, something meaningful in that command context may be typed, and you may pass a point at which DEL *or* undo can be used to correct the problem.

To reinforce the risk involved in not watching the system window, consider the following:

The word "edit" needs to be inserted in a text string.

Instead of typing **ieditESC**, "edit" is typed without first typing **i** for insert.

The result is:

e	Command to select everything in the file.
d	Command to delete everything in the file.
i	Activate the insert command (finally).
t	"t" is accepted as the text insert.
ESC	The insert command is terminated.

The file now on the screen contains a "t".

After **d** is typed, an undo command would restore the file unchanged. After **i** is typed, neither DEL or undo will restore the entire file. At this point, getting the text back requires retrieving it from one of Bravo's temporary storage areas. The following series of commands:

buffer
1 (numeral one)
ESC
replace
ESC

will restore the entire file. See Section XIII on Moving/Duplicating Text.

When some situation occurs which you do not understand, there is no great rush to correct it. Sit there for several minutes and sort out what happened; one clue is the message at the very top of the screen. Bravo's command messages print the name of the command activated in capital letters--APPEND will be the first word on the first line in the system window after typing a for append. That information provides a starting point. Having made some determination about what happened, select one of the possible responses offered in Section B, below, as a corrective measure.

B. Trouble Summary -- Levels of Response

Problem: An unwanted command is activated and *noticed* before any change to the document has occurred.

Response: Hit the DEL key to abort the command. Bravo displays a "command terminated" message when this response is acceptable.

Problem: An unwanted command is activated and is unnoticed until an unwanted result has altered the document. No other command or selection has yet been made.

Response: Type undo. For "most" Bravo commands undo will reverse the action of the *last* command.

Problem: An unwanted command(s) is activated and executed. Undo either does not work on that command, or does not have sufficient effect because two or more commands have been executed.

Response: Do a replay of the session up to a point prior to the catastrophe; then put the file on the disk, leave Bravo, return and start a new session. This response assumes that fixing the file would take longer or be more trouble than a replay.

C. Miscellaneous Information

1. The length of a Bravo session should not exceed an hour or so, and it is wise to put the file several times over the space of an hour. Bravo keeps a transcript of all commands, and the transcript uses some of the space on the disk which Bravo needs to function. Periodically, quit from Bravo, reenter and begin the editing or text input again. Be sure to put the file before leaving, however.

   Overstaying your welcome in Bravo, may result in the message: **"Suggest you P and Q"** (Put and Quit). When you get this message *immediately* put the file, quit from Bravo and reenter again. This action starts a new

transcript and frees space on the disk used by the previous transcript.

> [*Do not attempt to do just a few more edits before responding to this "suggestion" from Bravo. Bravo will not, in any case, allow more edits, but by repeatedly issuing commands, you may pass a point at which you* can *quit. Bravo will almost always Put the file, even when it will not honor a quit command. If Bravo will not respond to a quit command, but the file has been put on the disk, you can boot the Alto to get back to the Alto Executive and from there begin a new Bravo session.*]

2. The length of any Bravo file should not exceed 35,000 characters. A document longer than 35,000 characters should be broken into two files. The length of a file can be checked each time Bravo is entered, and each time the file is put on the disk. At both times, Bravo verifies the filename and indicates its length. (35,000 characters varies depending on the text, but with single spaced, fairly dense text, it is in the neighborhood of 10-12 pages.)

Go on to later sections if you wish; however, it will be easier to read and understand them if you have used and are comfortable with the information presented in Part I. The later sections provide less explanation, concentrating more on step by step sequences of commands and brief explanations of their result.

Part II

Conventions:

		What the user types will appear in boldface and underlined.
		System messages to the user are indented and appear in boldface.
CR	=	Type a Carriage Return
ESC	=	Type the ESC Key
↑	=	An up arrow preceding a character indicates a Control character--i.e., the Control Key is *held down while* the character following ↑ is typed.

This part of the Outline is a continuation of Part I for those users who need or want to use more of Bravo's features. It assumes that you have used and are comfortable with all features described in Part I. It is not accompanied by prepared exercises; the emphasis is on a step-by-step presentation of commands, and explanations are considerably more brief than they were in Part I.

Though no exercises accompany Part II, it is *strongly suggested* that each feature be *used* since no amount of explanation can duplicate the information transmitted by doing. The **Memo1.Bravo** file used for exercises in Part I is an ideal practice file. It or any text file of similar length (2-3 pages) will suffice.

XI. Formatting Continued

The sections following describe some of Bravo's more involved editing and formatting facilities, those that were not described in Part I and are not necessary until the user needs to change character sets (fonts), to control the placement of text with more precision, to offset text, and to interrogate Bravo in order to obtain point or inch positions for text placement.

A. Character Looks While Typing

This section describes super and subscripting characters; the following section covers changing the character set (font) in which text appears.

Superscripting

To superscript a character(s), type ControlShift↑ before the text to be superscripted and the CLR key following it. [In this case ↑ should be taken literally--i.e., it refers to the up arrow character on the keyboard, rather than indicating the Control key. Hold down *both* the Control key and the Shift key and type "↑".] Typing:

ControlShift↑2CLR

superscripts the number 2; any text appearing after CLR will *not* be superscripted. If Control↑ is typed as a lowercase character--if the shift key is omitted--subscripting will be the result. See the next paragraph.

Subscripting

To subscript a character(s), type Control← before the text to be subscripted and the CLR key following it. Thus, Control←$_2$CLR subscripts the 2. (Hold down the Control key and type ←, then type a 2 normally, then hit the CLR key to cancel subscripting.)

Once super or subscripting has been initiated with Control↑ or Control←, any amount of text can be typed, and it will continue to be super or subscripted until the CLR key is typed.

Super and Subscripting - Editing Existing Text

To apply super or subscripting to existing text, use a look command followed by ↑ or ←. These looks are most easily cleared with a look CLR. They *cannot be cleared* with a shift key character since both characters reside on the same key--therefore, ControlShift← or lookShift← will result in superscripting. However, look up 0 ESC will cancel super or subscripting without cancelling other looks. See G. Offsets Up and Down, for further information.

B. Font Changes

A variety of fonts (character sets) can be obtained easily and quickly. Appendix A provides a summary sheet which lists the *number* of each font *on your disk*, gives a sample of the character style and size, and provides correspondence tables. The fonts are numbered 0-9. Zero is the font used by Bravo unless another font is specified--it is the "default" font. [The term default refers to a number of definitions that are "preset" in Bravo and that will consistently occur unless changed by the user. There is a default left and right margin, a default font, etc., and they will automatically be in effect unless changed.]

In this outline, only the fonts already on your disk, numbered 0-9, are discussed. If a font that is not on your disk is needed, consult the *Alto User's Guide* and/or talk with someone who can assist you.

While Typing

To change fonts while typing text, hold down the Control Key and type the number of the font desired, then type the text which is to appear in that font, then hit the CLR key to restore standard font for subsequent text. An example:

> ↑1This is an eight point font which is better thanCLR a ten point font for footnotes.

Editing Existing Text

To change the font style of *existing text*, use a look command followed by the number of the font. That is, select the text to be affected, type l for look, followed by the number of the font--for the 8 point font above it would be l1.

Below is a chart illustrating how to get superscripting, subscripting and font changes, both while typing text and when editing existing text.

Look:	To Begin:		To Cancel:	
	Typing	Editing	Typing	Editing
superscript	CtrlShift↑	lookShift↑	CLR	lookCLR or lu0
subscript	Ctrl←	look←	CLR	lookCLR or lu0
0-9 fonts	Ctrl#	look#	CLR or ↑0	lookCLR or l0

When CLR is used to cancel a font, the text will always be returned to font 0--the default font, and CLR simultaneously clears bold, italic, super or subscripting, etc.--i.e., after typing the CLR key text will appear in font 0 with *no* special character looks remaining.

[Note: Font 2 is a 24 point font, containing only 4 characters, all uppercase. They are XERO, meaning that Font 2 is used only for obtaining a replica of the Xerox logo. This font can only be used for these four uppercase characters; any other character will produce a black rectangular bar, indicating that a character has been typed which Bravo cannot represent. Delete (BS over) these bar characters and begin again with another font or with the correct uppercase characters in font 2.]

Look ?

When using a number of fonts, it is easier to interrogate Bravo, rather than look up font numbers. Typing look? to a selected character will cause Bravo to display a message in the system window describing the character--its font number and any special looks, such as italics, attached to it.

C. Points, Inches, Centimeters

Several brief references have been made to points. It is time to say more about them since the following sections deal with point and inch specifications. However, it is not necessary to figure out distances to use them in Bravo--when a distance specification is needed, you may ask Bravo what the distance is by pointing to a place on the screen.

A point is a unit of measurement in the printing industry--72 points equal 1 inch. Points are used to measure character size, leading (spacing) between lines of text, and leading between paragraphs. Bravo uses this unit of measurement in its formatting operations. It has been using points all along, but you have not had to be aware of them. In the formatting functions described below, points will be more obvious because Bravo displays them, and because you have the option of typing them as a specification.

Beginning with Section E distance specifications will have to be made. Points are being mentioned here because the term is used extensively for the two commands described in Section D. However, for these two options no distance specification is required; the commands described in Section D are of the *preset* variety.

D. Paragraph Looks While Typing

In addition to centering, justifying and nesting paragraphs *while typing* (discussed in Part I), there are two methods for controlling the amount of space *between* paragraphs (the amount of white space that constitutes a paragraph break). These paragraph looks are particularly useful *because* they can be used *while typing text.* They are control characters with preset distances, just as nesting is preset at 1/2" (36 points).

When a larger break (more white space) is needed to separate headings from the text following, one of these commands can be used to obtain it. The default paragraph break in Bravo (the one in effect for the *forms* on your

disk) is 12 points. ↑o doubles this break since it provides an additional 12 points of leading--it adds 12 points to whatever is already set.

↑o *(Control o)*

Controlo or looko, meaning "open" will produce an additional 12 points (1/6") of leading between paragraphs. If a larger paragraph break is desired, typing ↑o (lowercase o) *at any point in the paragraph* will produce 12 additional points of white space preceding that paragraph. If editing existing text, the command is looko. This additional leading can be cancelled by typing ↑ShiftO--ControlShiftO while typing text, or lookShiftO when editing existing text.

↑q - *Control q*

A similar option for paragraph leading is available using ↑q, but the amount of space is cut in half--↑q provides 6 additional points of leading. It is exactly the same as ↑o in terms of the way it is input--↑q to start it while typing; ↑ShiftQ to cancel it while typing; and lookq to a selected paragraph(s) to start it while editing; lookShiftQ to selected paragraph(s) to cancel it when editing.

So, with the combination of ↑o and the preset default for paragraph leading, 24 point breaks can be obtained between headings and surrounding text, and a standard paragraph break can be restored with the use of ↑ShiftO. An example of how this works follows:

> This is the end of a text paragraph with 12 point leading.↑CR
>
> ↑o*This is a heading line with 12 additional points of leading added, causing a double break between this paragraph and the one preceding it.*↑CR
>
> This is the first paragraph following a heading; the ↑o typed above has been left in effect, making the break preceding this paragraph 24 points--the same as that preceding the heading paragraph above.↑CR
>
> This is the second↑ShiftO paragraph following a heading; its leading has been returned to 12 points by the use of ↑ShiftO.

↑q works just like ↑o, but provides half the space. If a 24 point break is too large, use ↑q instead of ↑o for an 18 point break. If a 12 point break between paragraphs of text is too small, use ↑q to expand the space to 18 points. These two control characters can be used as much as possible for special formatting needs, and they will eliminate having to go back and edit in the desired format.

E. Paragraph Looks with Distances Specified

Section D described the last remaining paragraph looks which can be put into a document *while typing*. The formatting commands described in this section cannot be applied *except as an editing function to existing text*. They provide great flexibility for placing the text on the page since an exact position can be specified. The tradeoff is that they are more complex to use. So, the following looks apply to entire paragraphs, and in each case they must be followed by a distance in points or inches. First, there is a description of their purpose, then a description of how to use them.

Before attempting to use the commands following, read the rest of Section E and all of Section F to discover both what these commands do and how to make them do it.

Margins

look:

left	Controls all lines along the left margin of a paragraph (default left margin is 85 points--1.2").
right	Controls all lines along right margin of paragraph (default is 527 points--1.2" of white space on right).

Paragraph Format

look:

first	To indent the first line of a paragraph. A look left command overrides look first because it affects *all* lines of a paragraph.
paragraph	Indents second and following lines of a paragraph. A look left command overrides look paragraph because it affects *all* lines of a paragraph.
x	Sets the amount of leading (space) between lines of text. Default is 1 point for single spaced text. Double spaced text can be gotten with look x 6. Leading *between* 1 and 6 points provides less densely printed lines.
y	Sets the amount of leading between *paragraphs.* Need not be used if ↑o or ↑q satisfy the requirement--they are 12 and 6 points, respectively. Maximum paragraph leading by using looky is 63 points. If more than that is needed, you will have to insert a blank paragraph(s) and set its y distance, or see the section on vertical tabs in Part III.

The sequence for all of these commands is:

Select a paragraph or paragraphs
look
left (for example, but r,f,p,x,y work the same way).
distance
ESC

F. Determining Distances on the Screen

You do not have to know anything about the distance of one point or calculate distances in either points or inches in order to use these facilities. At the precise point at which a distance is needed, Bravo will display it. After typing ll (look left) to a selected paragraph, the mouse can be used to point to a position on the screen, and Bravo will display its distance. The distance is displayed in the top, left-most set of squiggly brackets in the system window. The left most set of brackets displays the distance in points, the middle set of brackets displays it in inches. Once the distance appears in the left most set of brackets, hitting the ESC key will cause that distance to be applied to the paragraph(s) selected.

The "distance" referred to is measured from the *left edge* of an 8.5 x 11" sheet of paper. 0 is the far left edge; 612 is the far right edge. The distances specified to Bravo are for horizontal positions between 0 and 612 (usually between 85 and 527--the default left and right margin settings). Exceptions to this distance rule are x and y which are vertical measurements, though still given in points.

There are a variety of methods for determining the distance desired. The four methods given below should provide ample options.

Many of the methods for determining distances require the use of a "copy" or "secondary" selection. Secondary selections are made exactly like the primary selection, but they appear with *dotted* underlining. Secondary selections occur in the middle of a command, allowing the user to point to and thereby interrogate Bravo about specific points on the screen.

1. For the paragraph looks: left, right, first, paragraph, the desired distance can be obtained by using the mouse buttons in one of the following ways:

   After typing: looklook or lookr or lookf or lookp

   Red — selects a character, and the horizontal position of that character will be displayed within the brackets of the system window. So if you want the left margin to line up with some particular character in the paragraph above, select that character, Bravo will read its position and display it; hitting ESC will apply that distance to the paragraph(s) selected.

   Yellow — Use the yellow button to select a paragraph. If you want a paragraph(s) to have a left margin, for example, which is the same as another paragraph, type ll and then make a secondary selection of the paragraph it should match. Bravo will read that paragraph's left margin setting and display it--hitting ESC will cause the left margin of the paragraph being changed to be reconfigured with that setting. This principle works the same with lr, lf, lp.

Blue If there is no character to select, and no paragraph to get a setting from, or if either option is inaccurate, hold down the blue button and move the cursor to *any* horizontal point on the screen. Bravo will constantly update and display the position of the cursor. When the Blue button is released at a particular point, the selection will freeze and its distance will be displayed in the system window. Hitting the ESC key will cause Bravo to use the displayed distance for the paragraph setting indicated--left, right, first or paragraph.

2. *y or x Settings*

When setting either the amount of white space between paragraphs (y), or the amount of white space between lines within a paragraph (x), only the **Yellow** mouse button is functional. Following a look x or look y command, neither the **Red** or **Blue** mouse buttons have any effect--they both provide *horizontal* distances and Bravo knows of no horizontal distance when either y or x are being set.

After typing either ly or lx, you may make a secondary selection *of the paragraph to be changed*; Bravo will display its x or y setting. Points can be added to or subtracted from it, the new distance can then be typed and the command terminated with ESC. Or you may select another paragraph in the document (scrolling to it if necessary) which already has the x or y setting desired; Bravo will read the distance and hitting ESC will adopt that distance for the paragraph(s) being changed.

Standard distances for x are 1 point = single spacing; 6 points = double spacing, settings between 1 and 6 provide less densely printed lines. Some experimentation with y will have to be done, but a good rule of thumb is 12 points between text paragraphs, and 24 points separating headings from the text paragraph following.

3. *\ to Obtain Default Setting*

If you want to change a paragraph to its default setting, but are unsure of the default, type \ (backslash, not /). Bravo will display the default setting for the look being changed--85 points for left margin; 527 points for right margin; 1 point for leading between lines of text; 12 points for leading between paragraphs; 85 points for both first and paragraph. To adopt the default distance displayed, hit ESC. (The backslash option, like the secondary selections described above, works *after* you have typed look left, lp, lf, etc.)

4. Type the Distance

Some emphasis has been placed on using the distance Bravo displays by hitting the ESC key. However, you always have the option of typing the distance; it will appear in the left most set of brackets in points, and its inch equivalent will appear in the middle set of brackets. Typing ESC following the number will apply that distance to the look being set.

If you decide to type in the distance, the format to use for points, inches, centimeters is as follows:

144	A number without a decimal point and no indication of measurement following it, is assumed by Bravo to be in points.
2.0 2" 2.0"	A number with a decimal point or followed by " is considered to be in inches.
5.1cm	cm must follow the number if you are specifying centimeters, and there is *no space* between the number and cm--e.g., 4cm.

Look ?

Typing look? to a selected *paragraph* will cause Bravo to display all settings on that paragraph--its left and right margins, its y setting, its x setting, and a number of other possible paragraph settings. Bravo may not have room to display all the information; *scroll the system window* up to see if there is more information about the paragraph.

Plus or Minus Options

Bravo will add to or subtract from an existing distance on the screen. If you want a paragraph indented 1 inch, for example, do not worry about adding 72 points to the current setting; type look1+1"ESC (look left + 1 inch) and 1 inch will be added to the margin of the paragraph. Plus and minus options work with *most* Bravo commands that require a distance.

G. Offsets Up and Down - (look up, look down)

The offsets discussed in this section require that a distance be typed after indicating the offset desired (up or down). These offsets supplement super and subscripting, and need not be used unless the preset offset is inadequate. When a character is superscripted (Control↑), or subscripted (Control←), the offset is 4 points. If this is too much or too little, the offset up or down option can be used to assign the distance desired.

These offsets can only be applied to existing text; they cannot be used while typing text. So, to offset text *up* 6 points, for example, select the character(s) to be affected and give the following command:

look
up
6
Esc

Offsetting text down works the same way, except that down replaces up in the command.

Hints about Offsetting

An offset distance will be *measured from the baseline of text.* So, if you superscript a character (4 points), and decide that 6 points would be a better offset, that offset cannot be increased to 6 with the command lookup2Esc. Bravo will count from the baseline and offset the character(s) 2 points, meaning that it will actually move *down* 2 points from the initial offset of 4 points. The way to increase or decrease an offset using this command is either to give the total offset--lu6Esc (look up 6)--or use a + option with the distance specified--lu+2Esc (look up + 2).

Bravo adds leading to text lines in order to accommodate offset text. Therefore, use the smallest offset possible in order to avoid wide gaps between lines of text.

H. Miscellaneous Look Commands - Formatting/Formatting Aids

The looks described below provide several methods for formatting a document and for ascertaining what it will look like when printed.

Look All - (la with secondary selection)

When several *levels* of formatting are used, switching between them can be facilitated by asking Bravo to make one paragraph or series of paragraphs look like a paragraph set up elsewhere in the document. This can be done with a look all command.

Use of the look all command allows a "secondary" (dotted underline) selection to be made in order to identify the paragraph whose formatting is to be copied. Often the document must be scrolled to locate a paragraph that contains the desired format. You can scroll through the document as much as you like after typing la (look all); nothing will happen until a secondary selection is made and terminated with Esc. When the Esc key is hit, formatting from the copied paragraph is applied to the paragraph(s) being changed. The sequence of the command is:

Select the paragraph(s) *to be changed*
la (look all)
Scroll the document if necessary
make secondary selection of paragraph to be matched
ESC

Look Same - (ls with copy selection)

Look same works exactly like look all except that it applies to *character style only*--no paragraph formatting will be changed as a result of the look same command. If, for example, text is both in another font and in boldface, and a separate block of text is to match the text, select the block of text to be changed, give the ls (look same) command, make a secondary selection of the text it is to match, followed by ESC. The block of text to be changed will be altered to match the font and the boldface of the secondary selection.

BS for Overprinting

Using BS (the backspace key) to overprint characters can be done both while typing text and as an editing function for existing text. Typing:

<↑BS> (<ControlBS>)

will result in: ⁒. So, the principle for getting overprinting *while typing* is: first type the initial character, then ↑BS followed by the character which is to overprint the first character typed.

To get existing characters to overprint, a look command is used. If you have the following two characters:

<>

and want ⁒, overprinting the two characters can be obtained by selecting < and typing lBS (look BS).

To clear overprinting, select the character(s) to be affected, and type either lookCLR or lookShiftBS (backspace key).

Look Visible

The command: looky (lowercase v) initiates a "visible" mode in which characters that usually cause white space, are made visible--i.e., tabs, spaces, carriage returns, control carriage returns.

This command is useful when: 1) it is unclear whether text contains a series of spaces or a tab; it is unclear whether there is a control carriage return or an ordinary carriage return, etc., and 2) there may be a problem with the file which you cannot immediately identify without being able to see the characters involved.

To get rid of visible mode, type lookV (uppercase V).

Look visible pertains only to selected text--a paragraph, one line, or the entire document can be made visible, depending on the amount of text selected when the command is issued. It is possible to edit, store the file, print it, etc. while text is in visible mode. When cancelling visible mode, all of the text affected should be selected before typing IV.

It will be fairly easy to determine the visual representation for spaces and tabs, but the difference between carriage return and control carriage return requires careful attention. A control carriage return differs from an ordinary carriage return only in that it has a line across the top of the CR. Do a look visible to text that contains both and note the difference.

Look Hardcopy (lh)

The number of characters on a line when the document is displayed on the screen does not exactly duplicate the number of characters there will be on a line when the document is printed. In printed form, a line of text will contain more characters than it does on the screen. The look hardcopy command duplicates what the document will look like when printed.

Use of the look hardcopy command will affect the entire document, and the character set (font) used to duplicate the printed document is quite a bit smaller than the font generally displayed. Editing with this font involves a certain amount of eyestrain. However, an area of the text can be edited to ensure that lines break properly. Editing in hardcopy mode is also considerably slower--particularly if there are numerous font changes in the document.

To cancel hardcopy mode, type lookHardcopy (note the uppercase H to stop it and the lowercase h to start it). If you experience difficulty getting out of hardcopy mode, type e for everything before giving the lookH command.

XII. Tabs

There are three *types* of horizontal tab settings available in Bravo. In Sections A through C each type of tab is described and the method for setting it given.

No tab setting in Bravo can be actually set while typing text--they are all set with paragraph look commands, and they all require a distance specification. Consequently, to enter tabulated text initially, you must type the text, hitting the TAB key at each point where a tab will be needed, then select the paragraph and give a command or series of commands to reconfigure tab stop(s) to the position(s) desired.

A. Regular Interval Tab Settings

Bravo has "default" tabs preset at 1/4" (18 point) intervals. They act just like tab settings on a typewriter--a tab between text will move to the *next* 1/4" point on the screen, rather than 1/4" *from the point where the tab is hit.*

This regular interval tab of 1/4" can be reset to any interval--1/2" or 1", for example. The *paragraph look command* used to reset it is:

Select paragraph(s)
look
TABkey
=
1.0"
ESC

The = in this command is used to tell Bravo that *all* tabs are to equal the distance specified--in this example that is 1.0".

If this command is applied to paragraphs early in the document, it will remain in effect for paragraphs appended to them. If it is used as an editing function after all of the text has been typed, all paragraphs to be affected should be selected before the command is given--if all paragraphs are not contiguous, several individual selections will have to be made.

Regular interval tabs are being discussed first because they are the tabs in effect until others are set. However, for *almost all* formatting requirements involving text paragraphs or tabulated columns, *they will not be appropriate.* Tab requirements for hanging indents, and for tabulated columns are best achieved by using the tabs described in B, below.

B. Specific or "Named" Tab Settings

This type of tab is the most generally useful tab available in Bravo. Any horizontal position across the screen may be set with a tab stop. A line of text may begin on the left margin (85) and have tabs set at 120, 250 and 400 points--this is simply an example; any combination of tab stops can be set.

With "named" tab settings, *up to* 15 tabs can be set on a line, identified (named) with the numbers 0-9 and the letters a-e. 0 (zero) is the first tab and e is the 15th tab.

The paragraph look command for setting specific tabs is:

Select paragraph(s)
look
TABkey
0 (for example)
distance
ESC

After typing **look**TAB**0**, the mouse buttons may be used to determine the distance of the tab stop. E.g., .

Red to select a character on the screen which the tab is to line up with.

Yellow to select a paragraph with tab 0 already set, and "copy" that setting to the paragraph(s) being changed.

Blue to select any horizontal point across the screen where the tab is to fall.

The distance selected with any of these mouse buttons will be displayed in the left most set of squiggly brackets in the system window--points within the left brackets; inches in the middle brackets. When the distance desired appears in those brackets, hitting the ESC key will cause Bravo to apply that distance to the tab being set.

The "anatomy" of this command, both what you type and what the system responds with, is:

To a selected paragraph(s) type:	**look** **TAB**key
Bravo responds:	**LOOK: Type tab code (0-9, a-e, =)**

meaning, do you want a specific tab? If so, *which* tab: 0-9, a-e? Or are you setting a regular interval tab, in which case = is the tab code used. Answer this question with the number of the tab being set:

0

for example.

Having typed **0**, Bravo responds with:

LOOK: Specify tab stop, confirm with ESC.

At this point the mouse buttons may be used to determine the distance from the left margin where the tab will be set; or the distance can be typed. After doing one or the other, type:

ESC

and the distance specified will take effect in the document.

The same procedure is followed to set tab 1, and tab 2, and tab 3, etc. through tab e, the 15th tab. Named tabs must be set individually.

To move to a named tab stop without typing the intervening tabs, hold down the TAB key (as though typing a shift character) while typing the number of the tab. E.g., tab 5 is the position desired; hold down the TAB key *while* you type 5, and the cursor will move to tab 5 directly, bypassing intervening tab stops.

C. Columns at Equal Intervals (look table)

There is a look table command in Bravo for setting a group of tabs at *equal intervals.* The table command sets "named" tabs. It is being discussed as a "different" tab because setting it, and the applications for which it is used, bear little resemblance to the named tabs discussed above. The look table feature is only useful for *equal* amounts of white space between columns. If columns of variable width are wanted, use the "named" tabs described in Section B.

In the following table, default margin distances are used in response to Bravo's "specify left edge" and "specify right edge" request during the command. Any edge area can be specified, but the command has *no effect* on the *margins* already set on the text paragraph(s). That is, if the margins are set at 85 and 527, edge specifications of 120 and 500 will not change the margins in the text to 120 and 500.

Column 1 **Column 2** **Column 3** **Column 4** **Column 5**

The sequence for setting up this column is:

1. Type the text with a tab between each column, using whatever tabs are set--Bravo's default tabs, or tabs in effect due to an earlier tab setting.

2. Select the line of text *as a paragraph.*

3. Issue the following command:

look	
table	
85	(left margin)
ESC	
527	(right margin)
ESC	
5	(number of columns)
ESC	

Note: After typing look table, copy selections may be made with the mouse to determine the margin settings desired; or the \ option may be used to get default settings displayed (e.g., type \

ESC for both margin settings, instead of typing the numbers 85 and 527).

When the *final* ESC is typed, Bravo reconfigures the columns, spacing them out at equal intervals.

Like all tab settings in Bravo, these will remain in effect until changed.

A general rule for tabulated columns is that the heading line is a paragraph (ends with a ↑CR) and subsequent lines of text end with CR's. That is, a table containing a heading and 12 lines of tabulated text would consist of 2 paragraphs--the heading line is a paragraph, and all 12 lines of text are *one* paragraph.

Caveat

An important caveat is that the look table command begins with tab 1 when it resets tab positions. Thus, if you append to a paragraph that has a tab 0 set, the new paragraph will pick up that 0 tab, and the look table command will not reset it. If 0 is set beyond the position specified as the left edge of the table, it will cause an unbalanced reconfiguration of the table. To solve this problem, *be sure that tab 0 is set equal to or less than the distance given as a left margin setting*--e.g., if the left margin is 85, be sure that tab 0 is set at 85 or less. An easy way to handle this caveat is to issue the command: lookTAB=18 to the paragraph before giving the look table command. This command sets the default plain tabs and will ensure that no named tabs are conflicting with the table command. [Incidentally, to find out all tab settings on a paragraph, select the paragraph and type look?--it may be necessary to scroll the system window up, but both the distance and "name" of each tab will be displayed (the message: "plain tabs" indicates regular interval tabs).]

Copy All Tabs Option

After giving the look table command, the **Yellow** mouse button may be used to copy all tabs from another paragraph in the document. This option works like look all, but gives you the option of copying only the tabs, without margin or other formatting being copied along with tabs. It does not matter whether the tabs are set at equal distances--this option copies any set of tabs. To use this option, the sequence is:

Select paragraph(s) on which tabs are to be changed
look
table
make secondary selection of paragraph with desired tabs
ESC

When the secondary selection is made, both the delete buffer (far left) and the insert buffer (middle) display the message: {(all tabs)}. When ESC is typed, the tabs are applied to the paragraph(s) being changed.

XIII. Moving/Duplicating Text -- System Buffers

The three sets of squiggly brackets (braces) in the system window contain various pieces of text, and the text changes periodically. These brackets serve as temporary storage areas (buffers) for Bravo, and they contain text manipulated in the course of editing or creating a document. The buffers are numbered as follows:

1 {last deletion} 2 {last insertion} 3 {search key}

So, the buffer on the left is numbered 1, and contains text most recently deleted from the document. Buffer number 2 contains text most recently inserted. Buffer number 3 contains the text of the most recent search--this buffer is described in XIV. "Search and Substitute Options."

One of these buffers will be marked with a "*" to indicate that it is the "current" buffer. Any of the buffers may be *made* current with the command buffer#ESC. Before typing ESC there is an opportunity to type or select text which will take the place of the text already in the buffer. If the text already there is wanted, type ESC, and the command will simply make the buffer current.

The contents of these buffers will be displayed in visible mode--i.e., spaces, tabs, CR's and ↑CR's will be represented. Also, since the buffer is small and cannot contain larger blocks of text, the text is elided. Bravo displays the beginning of a text string, then an ellipsis (...) to indicate that more text intervenes, and then the end of the string of text.

A. Delete Buffer

When text is deleted from a file, the deleted text goes into the delete buffer (buffer 1). It will reside there until another deletion is made, at which point the more recent deletion will take its place.

Text in the delete buffer may be restored to the document. Occasionally, text needs to be deleted from one point in a file and moved to another point. The delete buffer makes this possible without retyping the text.

For example, a paragraph at the end of a document should appear nearer the beginning. The sequence for effecting this move is:

Select the paragraph to be moved.
delete the paragraph.
Select the point at which it is to appear nearer the beginning of the document.
Give either an insert or append command, whichever is appropriate.
Hit the ESC key.

After typing ESC, the deleted paragraph will be inserted in its new position in the document.

This operation works because following a delete command, *the delete buffer is current.* Whenever an insert, append or replace command is "defaulted" (ESC is typed instead of supplying text from the keyboard), *Bravo uses text from the*

current buffer.

B. Insert Buffer

The insert buffer (buffer 2) works much like the delete buffer, though it is not functional in moving text from one point to another unless the text is to appear in *both* places--be copied in a second location, but not deleted from the first position.

If there are several insertions involving the same text, the text need only be typed once. Text inserted into the document, also goes into the insert buffer. Therefore, if you select a second point in the document and type insert ESC the text from the insert buffer (the current buffer following an insert) will be placed in the document at the position selected.

These buffers may be confusing until they have been used a few times. A good way to get a sense of what is happening is to delete a string of text and notice the change in contents of the delete buffer. Then give an insert command with another point in the document selected, and hit the ESC key. The text from the delete buffer will appear in the document; it will also appear in the insert buffer and that buffer will get the *, making it the current buffer.

At the outset these buffers seem like a shell game in which you try to guess where the pea is; however, they can be quite useful in moving text without retyping it, and duplicating the same text in different locations.

C. Copy Selections with Append, Insert, Replace

Following an append, insert or replace command, a secondary selection of text appearing elsewhere in the document can be made. ESC following the secondary selection will cause the selected text to be placed at the position indicated by the primary selection. The sequence is:

Select *point* at which text is to be appended, inserted, or the text which is to be replaced.

Type a, i or r

Make a secondary (dotted underline) selection of the text to be copied, and type ESC.

The text of the secondary selection will be copied to the position initially indicated. The text of the secondary selection is *not deleted and moved*; it is a *copy* placed at another location--it will appear in *both* places following this command. If text is to be deleted and moved to another position, see Delete Buffer, above.

XIV. Search and Substitute Options

Bravo provides several facilities for jumping to a particular point in the document; for substituting one string of text for another; and for making *selective* substitutions.

A. Jump Command

When editing a lengthy document, scrolling to a point several pages away may not be the quickest way to access the next edit. The jump command allows you to type a string of text in the document which Bravo is to find and display at the top of the screen.

The jump command begins its search with the *second line on the screen, and continues searching to the end of the document. It does not start again at the beginning.* Therefore, make sure that the second line of text on the screen appears at an earlier point in the document than the text being located. An easy way to search the entire document, is to type en (everything normalize) before giving the jump command.

The sequence for the command is:

jump
type the text string to be located
Esc

Text typed will appear in the Search Buffer--buffer number 3. Bravo will pay attention to capitalization and to punctuation, but will not differentiate text on the basis of font size or character looks--i.e., you need not type the text in boldface because it appears in the document in boldface, but you must capitalize characters that are capitalized in the document. In fact, you cannot specify only boldface instances of a text string without jumping to the intervening instances of the text string which are not boldface.

Once ESC has been typed, Bravo will search the document; when the text string is found, the line containing it will appear at the top of the screen.

If there are several instances of the string of text, Bravo will find the first one and place it at the top of the screen. To find the next instance of that same string, type ESC--ESC causes Bravo to repeat the last command. You can do this several times if necessary, though choosing a *relatively* unique string of characters the first time will be less time consuming.

B. Substitute Command

The substitute command allows you to type a new text string which is to be substituted for an existing text string. E.g., substitute $$$$$ (for) Bravo.

Substitute *operates only on the text selected*--within a paragraph, within a line of text, throughout the entire document. A certain amount of care should be

taken when making widespread substitutions, however. Bravo will take the command quite literally. If you are substituting "this" for "that" and Bravo finds an instance of t h a t which appears within a word, it will change it to t h i s. When substituting one word for another, it is best to type a space in front of the word before turning the substitute command loose on your document.

The sequence of the command is:

select the amount of text to be affected
substitute
type text *you want*
Esc
...FOR (question asked by Bravo)
type text you *do not want*
Esc

The text *to be substituted* appears in the middle (insert) buffer; the text to be changed appears in the Search Buffer, since that is the text Bravo will search for in order to insert in its place the text in the insert buffer.

When the substitute command is activated, Bravo asks that you "type or select" the text to be substituted. A string of text in the document may be selected instead of being typed from the keyboard--the selection is a secondary selection and will appear with dotted underlining.

C. **Find Command** - Selective Substitutions

The find command has two uses. It can be used just like the jump command for locating a particular point in the document. However, there are two differences between Jump and Find:

> Find starts the search *from the end of the current selection,* not from the second line of text on the screen, and
>
> When it finds the text and places it at the top of the screen, *it makes that string of text the current selection*--it will be underlined and take the place of whatever text was previously selected. The jump command does not change the selection in this manner.

So if you use the find command to get to a point which is to be edited, that text will be selected and a command can be given to alter it.

The second use for find is to change *some* instances of a text string, but not *all* instances. With the substitute command there is no opportunity to say *which* instances are changed--they all are.

To *selectively* substitute text using the find command, the sequence is:

find
type the text to be located
Esc

When the text has been found and is the selected string at the top of the screen, type:

replace (do not use insert or append)
type the new text
Esc

Now, to move to the next instance of that string of text, type:

find
Esc

and Bravo will move to the next instance, display it at the top of the screen, and select it. From this point on type either:

yes

to repeat the replace command--to make the substitution--and then move on to the next instance; or type:

Esc

to *make no substitution* and go on to the next instance.

The find command does not constrain you to continue hitting Esc to the end of the document. When all the instances to be changed *have been* changed, simply start something else--Bravo is in a "Ready" state after either y or Esc are typed, so the command can be discontinued at any point. Be sure all the changes have been made before interrupting with another command; once the order of the command has been interrupted, the entire command will have to be initiated again, starting with find and typing the text, etc.

This ends the material presented in Part II. Part III describes the use of multiple windows; vertical tabs for placement of text; document profile options such as line numbers, odd and even headings, and multiple column printing; Bravo's calculator; and some introductory information for Maxc users. If you do not have a need for these features, there is no need to go on to Part III.

Part III

Part III covers the remaining Bravo features: Windows: Page Boundary; Vertical Tabs; Document Profile; and some standard Calculator functions. There is also a section for Maxc users, though it covers only a small portion of the File Transfer Program and describes how to print Bravo files from Maxc.

After completing Part III there will still be a vast amount of information about the Alto and Maxc operating systems which has not been touched on in this Bravo outline. Further Alto documentation is contained in the *Alto User's Handbook*, and Maxc documentation in the *Tenex Executive Manual*, and the *Tenex User's Guide*.

XV. Windows

All editing and text entry operations to this point have involved the use of one document window. It is possible to open many windows and have *a separate file* in each window, or to have many subwindows, each providing a different view *into the same document.* While many windows are possible, their use is not recommended, and this section discusses the use of only one additional window or subwindow at a time. There are two reasons for this approach: 1) Handling more than two windows strains the system and increases the possibility of a Bravo failure; and 2) describing only one additional window at a time can be done with greater clarity.

There are two *types* of window available: 1) a *new window* in which *a separate file* can be displayed; and 2) *a subwindow* which provides an additional view into the *same file.*

Two separate methods for manipulating windows are discussed. Sections A and B discuss new and subwindow manipulation using a *specific* mouse button for each operation. Section C discusses a keyboard option which allows any of the mouse buttons to be used after the desired operation has been indicated with a single character keyboard command. Read through both methods and choose the one you prefer. Both methods require mouse selections; however, with the method described in Section C, it is not necessary to remember what each mouse button means--five single character commands must be memorized, after which any mouse button can be used. Many people prefer this method of window manipulation because fewer distinctions need to be remembered. It would be a good idea to choose one or the other of these methods since remembering both will be quite confusing. However, it will be necessary to read all three sections since information about how to use windows--possible applications, moving text blocks across window boundaries, etc.--is contained only in Sections A and B. It is not repeated for the Keyboard Options described in Section C.

A. New Window *(two windows/two documents)*

A second window makes it possible to excerpt text from another file on your directory for use in the file currently being created; to reference information in another file without printing the file; and to join two separate files together.

When a new window is created, it has a broad black border identical to the original window, and the filename of any document read into the new window will appear within its border.

Open New Window

Using this method of window manipulation, a mouse button specifies both the *kind* of window and where its boundary is to be placed. To create *a new window, into which a separate document* can be read, type w for window, place the cursor against the left margin at the vertical height where the new window boundary is to be placed, then hold down the **Blue** button for at least a second. While the button is still depressed, the window *boundary* can be changed by moving the cursor up or down; when the button is released the boundary will freeze.

When a new window is opened, the end-of-file marker in the new window is selected. Selecting anything in a document makes the window containing the selection "current." Therefore, with the end-of-file marker selected, a command to get a file will read that file into the newly created window. The window containing a file with *selected text* is the only window that will be affected by keyboard commands. To make a window "current" or active, simply select any block of text within it.

With two files, one in each window, text can be transferred back and forth between them. To take a block of text from one file and insert it into another file, select the point *at which text is to be inserted*, type i for insert, then make a copy selection of the text in the other file, and hit the ESC key. This is the same sequence that would be followed with *a single file.* Deleting text from one point and moving it to another, or inserting text from one file into another file, *works the same across a window boundary as it does within a single file.* If text is to be deleted from one file and placed in another, select the text to be deleted, delete it, select the point at which it is to appear in the other file, type i for insert (or a for append), then hit the ESC key--again, the same operation that would be performed when dealing with a single file.

To join two separate files together, select the last paragraph of the first file, type a for append, then make a copy selection of all, or the pertinent part of, the second file, and hit ESC.

A new window can also be used to facilitate formatting a complex document. Once the formatting levels are established, a *file* with several paragraphs, each containing a particular format needed in the document, can be created and read into the new window. Then when one of these formats is needed in the document, it can be adopted with a copy selection to the look all command.

Change New Window Boundary

If the window originally opened provides too much or too little space, its boundary may be changed. Type **w** for window, then place the cursor in the broad black border of the window boundary to be changed. Now hold down the **Red** button and while it is depressed, move the cursor to the new vertical position for the boundary. The existing boundary will follow the cursor's movement until the button is released, then the boundary will freeze.

Destroy New Window

When the additional window is no longer needed, it can be destroyed. Type **k** for kill, place the cursor in the window *to be destroyed* and push either the **Red** or **Blue** button. **Red**, *with the cursor placed in the bottom window* will kill the bottom window and give the full screen to the top window. **Blue**, *with the cursor placed in the top window* will kill the top window and give the full screen to the bottom window. *Be sure to put the file on your disk before killing a window in which changes have been made to the document it contains.* If the file has not been put, Bravo will require a confirming CR to destroy the window; however, it will not explain *why* confirmation is being requested.

Clear Contents, Leave Window

To "clear" a window of its contents and leave the window, type k for kill, then place the cursor in the window to be cleared and push the **Yellow** button. This action is equivalent to window clear. Again, if changes have been made, put the file before clearing the window, though if the file has not been put on the disk, Bravo will ask for a confirming CR before clearing the window. This should be taken as a reminder to put the file if changes you care about have been made.

Summary of New Window Commands

The following chart summarizes the commands for creating a new window, moving its boundary, clearing and destroying it. After typing either w or k, the mouse buttons work as follows:

window

Red	Moves window boundary
Blue	Opens new window and places its boundary at height of cursor.

kill

Red	*With cursor in bottom window,* kills that window, gives full screen to top window.
Yellow	With cursor in top or bottom window, erases the *contents of that window,* leaves the window.
Blue	*With cursor in top window,* kills that window, gives full screen to bottom window.

The yellow mouse button with a window command has no application for a *new window,* but it is functional in creating a "subwindow" which is discussed in Section B, below.

B. Subwindow *(two windows/one document)*

A *subwindow* differs from a new window in that subwindows *look into the same document.* The original window and the subwindow can be scrolled independently, making it possible to view two separate areas within a file, but both windows contain *the same file.* Any change made to the document in *either* window will take effect in the other window--i.e., *you are always working on one document.*

Inserting text from one point in the document at another point works the same way across a subwindow boundary as it does with full screen display of the

document. Deleting text from one point and inserting it elsewhere also works the same. The only difference is that both areas can be made visible at once, making scrolling during the command unnecessary.

A subwindow is also useful with certain formatting tasks. With levels of formatting, several paragraphs can be added to the end of the document, each containing a formatting level used in the document. Then when that level is required for new text, it can be adopted easily with a look all command, using one of the paragraphs at the end of the document as the copy selection. Having these formats established and continually visible saves either scrolling through the document to find an instance of the formatting desired, or giving a series of commands to format new text.

Open a Subwindow

To open a subwindow using this method of window manipulation, type window, then place the cursor along the left side of the screen at the height the subwindow is to appear, and hold down the **Yellow** button for about a second. Subwindows have no broad black border to accommodate a filename since they *cannot contain a separate file.* A thin black line marks their boundary.

Change Subwindow Boundary

If the subwindow originally opened is too large or too small, its boundary can be changed in exactly the same way as a new window boundary is changed--type w, place the cursor on the existing boundary line, *hold down* the **Red** button while a new boundary is traced with the cursor; when the button is released, the boundary will freeze.

Destroy Subwindow

When the subwindow is no longer needed, type kill, place the cursor in the bottom window and push the **Red** mouse button. The distinction made between **Red** and **Blue** for killing a *new window* can also be made for a *subwindow.* However, since the *same file is in both windows and any changes made in one window take effect in the other,* it does not matter which window is destroyed. The only advantage to choosing which window to kill is in keeping one of the selected points visible after the other window is killed. When this option is not a consideration, an easy method for killing *subwindows* is to type k, place the cursor in the bottom subwindow and push **Red.**

Because a subwindow contains the same document as the original window, there is a mouse button which should not be used if the intention is to clear *only one* of the windows. Kill and the **Yellow** mouse button perform the operation: erase the contents, leave the window. Be careful of this combination when working with subwindows. Its only use is to *clear the entire screen* of *all text*--i.e., using it with subwindows assumes that a new document is to be read in for which a subwindow is also desired. It *cannot be*

used to clear the subwindow only. Also, a kill, **Yellow** cannot be "undone"--this is one of the places in Bravo where undo does not work. If you type k followed by **Yellow** in a subwindow and have not put the file, it may require a replay to get it back--Bravo asks for a confirming CR when the file has not been put, but it will not explain *why* confirmation is requested. Typing DEL instead of a confirming CR will cancel the kill command and the file can be put on the disk normally.

Summary of Subwindow Commands

The following chart summarizes the commands for creating a subwindow, moving its boundary, destroying it, and clearing the contents of both the original window and the subwindow. After typing w or k, the mouse buttons work as follows:

window

Red	Moves subwindow boundary
Yellow	Opens a subwindow and places its boundary at cursor position.

kill

Red	With cursor in bottom subwindow, kills the subwindow, gives full screen to top window.
Yellow	With cursor in either window, clears the entire screen--both windows--of all text.
Blue	With cursor in top window, kills that window, gives full screen to bottom subwindow.

With window, the **Blue** mouse button has no application for *subwindows* since **Blue** creates a *new window.* Using either **Red** or **Blue** as they are described above in combination with kill, will produce the same result--the document, all edits included, will be displayed full screen.

C. Creating, Moving, Destroying Windows--Keyboard Option

Sections A and B above described window manipulation using *specific* mouse buttons for operations desired.

A second option for window manipulation consists of the window command with the following sub-options:

window	new	To create a *new window*
window	split	To create a *subwindow*
window	move	To adjust window boundary
window	destroy	To destroy a new or subwindow
window	clear	To clear the contents, leave the window.

After you type wn, ws, wm, wd, or wc, any mouse button may be used to either indicate a boundary or select a window. With this method of window manipulation, no distinctions need be made between which mouse button to use--the operation is described by the second character, rather than by a specific mouse button, and once the operation is specified, all mouse buttons are synonomous.

To create a new window, type wn and use any mouse button to indicate where the new window boundary should appear.

To create a subwindow, type ws and use any mouse button to indicate where the subwindow boundary should appear.

To move a window boundary, type wm and use any mouse button to trace the desired new or subwindow boundary.

To destroy a new or subwindow, type wd, place the cursor in the window to be destroyed and push any mouse button--Bravo displays the message "select window" after wd is typed. If a *new window* is being destroyed and the file has not been put on the disk, Bravo requests a confirming CR before honoring the command.

To clear the contents of a *new window*, type wc. Bravo responds with the message: "select window." Any mouse button may be used to select the window to be cleared. If the file has not been put on the disk, Bravo requests a confirming CR before honoring the command. Window clear with a *subwindow* will clear the entire screen of all text; however, if the file has not been put on the disk, Bravo requests a confirming CR before honoring the command.

D. Page Boundary

While editing or creating a document, Bravo makes no attempt to indicate where the pages will break when the file is printed. Page Boundary can be used to determine page breaks before sending for final copies of the document, and while the document is reconfigured into pages, it can be edited until the pages break correctly. Page Boundary can also be used to determine whether a block of text will fit *on one page*--again, it can be edited until the text fits one page before being sent to the printer.

Use a file that is *at least* two pages long when experimenting with Page Boundary for the first time; otherwise it will be difficult to see how all of its features work.

When Page Boundary is activated, Bravo will assume that *the first character* of the current selection is the first character on a page and will calculate page

breaks from that point. If page breaks are being determined for the entire file, scroll the document to the beginning and select the first character that will appear on page 1 before initiating the page boundary command. Or, if there is a known page break (because a ↑L--manual page break--has been entered), you can begin after the page break by selecting the first character to appear on the page following the ↑L.

Activate the Page Boundary command by typing the LF (linefeed) key on the keyboard.

After typing LF, Bravo will break the screen at roughly the middle and open a subwindow. The third line in the subwindow will be underlined, indicating that it is the line with which the next page *will begin.*

Typing LF again will cause Bravo to move to the next page. Then the third line in the subwindow will indicate the line with which that page will begin.

Page breaks for the entire file can be determined by typing successive LF's until the end of the file is encountered.

Page Boundary also duplicates hardcopy in that the *line breaks* indicated duplicate those in the printed document. Either line breaks or page breaks can be edited while Page Boundary is in effect. Manual page breaks can be entered where appropriate; or look keep commands can be used; and any line editing necessary can be done.

The page boundary command leaves the line which starts a page at the top of the screen; the first line of the next page will be the third line in the bottom subwindow and it will be selected (underlined). Each page should be edited individually for line or page breaks before going on to the next page. To verify the effects of any editing, reselect the beginning of the page (top line in top subwindow) and repeat the LF command. Bravo will honor the edits and reconfigure the page accordingly. When each page has been edited, and all lines and pages break satisfactorily, send the file to the printer.

To get out of Page Boundary and return to full screen, standard font display, two separate actions must be taken--type:

window
destroy

Then place the cursor in the bottom subwindow and push any mouse button. (The other method is to type: kill, place the cursor in the bottom subwindow, and push the Red mouse button.)

The above action cancels the second window opened by Page Boundary. To restore standard font display, type:

everything look Hardcopy **(elShiftH)**

XVI. Vertical Tabs

By setting paragraph looks with distance specifications, and/or specific tab stops, text can be placed on the page at precise *horizontal* positions. The same precision is possible for *vertical* positions on the page with the use of vertical tabs.

A vertical tab is set on *a* paragraph--the *same* vertical tab is not set on a series of paragraphs, as horizontal tabs may be. If the same vertical tab is set on two or more paragraphs, the text of those paragraphs will overprint--second paragraph prints *on top of* first, third paragraph prints on top of second on top of first, etc. When a vertical tab is set the text *will be placed as specified*, even if that means overprinting existing text.

The command to set a vertical tab is:

look
z
distance
ESC

The command to *remove* a vertical tab is:

look
z
\ (backslash, not /)
ESC

There are several points to keep in mind when using vertical tabs:

1. A vertical tab can only be set on existing text--it is a paragraph look with a distance specification, a combination not possible in Bravo while typing.

2. The distance set for a vertical tab is measured *from the bottom of the page*--e.g., setting a vertical tab of 2.0" will cause text to begin printing 2 inches *up from the bottom of the page.*

3. A vertical tab setting on a paragraph "goes with" the paragraph, just as left margin and horizontal tab settings do. Therefore, appending to or inserting in front of a paragraph with a vertical tab, will cause the new paragraph(s) to "pick up" the *same* vertical tab. Overprinting of paragraphs will be the result.

4. Horizontal tabs *may* cause a page to break. Vertical tabs do not cause a page break, though if set on the *first* paragraph of several pages of text, subsequent page breaks will occur normally.

5. If a vertical tab is used to place text as the top line on the page, its y leading and the x leading of the first line will be omitted. If a vertical tab is used to place text in the middle or at the bottom of a page containing other text, the vertical tab specifies the start of the y leading, not of the text itself. E.g.,

if a paragraph set with y leading of 24 points is assigned a vertical tab setting of 4", and other text precedes it on the page, the 4" vertical tab will specify the point at which the y leading begins, and the text will appear 24 points down from 4". Heading text entered into the document profile and set with a vertical tab always has y and x leading honored. A simple way to deal with this inconsistency is to check the y setting on a paragraph to receive a vertical tab (look ? to the paragraph), remove any y leading, then set the vertical tab for the specific point where the text is to appear.

There are a variety of applications for vertical tabs. One of the major uses is to specify text placement for headings in the document profile (discussed in the Document Profile section following). However, there are formatting functions *within* the document for which vertical tabs are useful. Several formatting applications are discussed below (not an exhaustive list) in which vertical tabs are combined with manual page break directives to achieve desired effects.

A. Title Page

The example below describes formatting a page which contains a centered title 4 inches down from the top of the page, and a paragraph at the *bottom* of the page for organizational credits--"This project was funded and supported by...."

Both paragraphs are typed *(as separate paragraphs)*. Then a vertical tab is set on the *title* paragraph:

look
z
7.0"
Esc

Then a separate vertical tab is set on the credits paragraph:

look
z
2.0"
Esc

At the end of the text (credits) paragraph, a manual page break is entered, ensuring that any text following that paragraph will appear on the next page--i.e., after the period ending that paragraph, type Its.

No change in the text will appear on the screen, but the directives entered will be honored when the document is printed.

B. Footnotes

Bravo has no actual footnote feature; however, it is possible, if painful, to format footnotes which appear at the bottom of the page. For only a few (half

a dozen) footnotes, it may be worth the trouble. If there are many footnotes, strong consideration should be given to an end-of-chapter footnote style.

No effort should be made to format bottom-of-the-page footnotes until a *final* copy of the text in which they appear has been obtained. The text of a footnote is entered after the last line of text that can appear on the page, still leaving room for the footnote at the bottom. The footnote is a separate paragraph, and gets a vertical tab setting depending on the length of text it contains--the example below assumes 1.0" of footnote text.

Example

A final copy of the paper or report has been printed and footnotes are to be entered. Decide where to break a footnoted page by determining how much text can appear on the page when a footnote must appear below the text and when a 1" margin is to remain. E.g.,

Last line of text to appear before the footnote.

(white space)

footnote text(l↑s)

(1' bottom margin)

After the last line of text, append the footnote paragraph, and end the footnote text with a l↑s. Then select the paragraph and type:

```
look
z
2.0"                    (1" is left for bottom margin)
Esc
```

The footnote, containing 1 inch of text, will print 2 inches from the bottom of the page, leaving a 1 inch margin below it. It will be necessary, however, either to include the y setting on the footnote paragraph into your 2" measurement, or to cancel the y setting on the paragraph (look y 0). Otherwise, the 2" specification will indicate the point at which the y leading begins and text will appear below 2" by the amount of leading set on the footnote paragraph. (See 5 on p. 75.)

The footnote ends with a manual page break (l↑s) to ensure that any text following it will begin on the next page.

This process does not sound very odious; however, in practice it gets cumbersome since as the text is broken on one page, subsequent text moves down on the page and for the second and following pages where footnotes appear, it gets more and more difficult to determine where the text should be broken without reprinting the file to see how text is reconfigured by earlier footnotes. Page Boundary can be useful in this regard, though editing extensively while in Page Boundary proceeds slowly.

C. Figures/Illustrations

If figures or illustrations are to be added to a page, there will be constraints on where the text describing them appears. Vertical tabs make it possible to dictate where descriptive text appears and can also be functional in providing white space for illustrations.

Example

A block of text has been typed, the next thing to appear in the document is a figure which will take up 6 inches on a page, and a block of descriptive text directly underneath the figure which will require 2 inches. The figure and the text describing it will appear on a separate page.

At the end of the last line of text to appear on the page *preceding* the figure page, type l↑s to break the page, then type the descriptive text for the figure, then type another l↑s.

Now set a vertical tab on the figure text, indicating where it will appear on the page--lz3.0"ESC, for example. The result of these actions is that a page has been set aside with 6 inches of white space to accommodate the figure, and the descriptive text has been placed on the page to appear directly underneath the figure.

Vertical tabs serve a wide variety of formatting needs--the above examples are given only to provide a "feel" for how they might be used. It is unlikely that the first use of vertical tabs will proceed altogether smoothly. When an unexpected result occurs in the document, take your time in figuring out what happened. Reread the five points at the beginning of this Section to see if one of them provides an explanation.

XVII. Document Profile

Most forms on the disk contain a document profile of some type, with directives for headings, page numbers, margins, etc. When none of the forms satisfies your needs, a document profile may be devised which does.

The document profile given below contains *an example line* for each possible option; it does not define a specific document. In general, any of the lines may be omitted, and fields within a line may be omitted, depending on the profile needs for the document. The categories are typed with either spaces or tabs for separation; and control carriage returns are used for *necessary* paragraph breaks (see below).

A. Document Profile Options

Page Numbers: Yes X: 527 Y: 10.5" First Page: 79 Not-on-first-page
Columns: 2 Edge Margin: 60 Between Columns: 40
Margins: Top: 1.3" Bottom: 1" Binding: -13
Line Numbers: No Modulus: 5 Page-relative First Line: 1
Odd Heading: Not-on-first-page
XVII. Document Profile
Even Heading:
Bravo Course Outline

This document profile consists of *four* paragraphs:

1) "Page Numbers" *through* "Odd Heading" is *one* paragraph.

2) **XVII. Document Profile** is *one* paragraph.

3) "Even Heading" is *one* paragraph.

4) **Bravo Course Outline** is *one* paragraph.

On *all* of the paragraphs, a *profile property* must be set, marking the information as a series of directives. To set the profile property, select *all four paragraphs* and issue the command:

look
;

To remove the profile property, the command is:

look
: (shift;)

Also, the heading lines (*actual text for headings in the document*) are given a vertical tab setting for placement on the page. If headings and page numbers both appear at the top of the page, they will carry the same vertical tab setting. Page numbers in this profile (see explanation below) appear 1/2" from the top of the page on the right margin. The odd and even headings should

also appear 1/2" from the top of the page so they print on the same line as page numbers. To set vertical tabs on each heading, select them *individually* and to each heading paragraph issue the command:

look
z
10.5"
Esc

Having given both the "profile property" command and the vertical tab commands, the document profile is complete. It should appear as the *first thing in the document.* It is wise to type at least one paragraph of the actual document before returning to set the profile property and vertical tabs for headings. If the heading paragraph containing the profile property and vertical tab is appended to, subsequent paragraphs will also carry profile property and vertical tab settings. The result will be an: "error in document profile" message from Bravo when the file is printed. This problem is somewhat difficult to discover since exhaustive changes may be made to the document profile before realizing that settings on the actual text of the document are causing the error.

B. Document Profile - Line by Line

Page Numbers: Yes X: 527 Y: 10.5" First Page: 79 Not-on-first-page

If page numbers are not wanted, "no" follows Page Numbers. X is measured *across* the screen; 527 is the default right margin, the horizontal point at which the page number will appear. Y is measured up and down on the screen; 10.5" is a vertical tab setting which will place the page number 10-1/2 inches up from the bottom of the page, 1/2" down from the top of the page.

First Page: 79 stipulates the number with which page numbering should begin. If First page does not appear, page 1 is assumed. Not-on-first-page indicates that no number is to be placed on the first page of the document. If Roman numerals are wanted, type the word Roman in the Page Numbers line:

...First page: 79 Roman Not-on-first-page

for example, and type Roman in Uppercase if Roman numerals are to appear in uppercase.

Private Data Stamp: No X: 3.5" Y: -.6"

This line can be used to get Xerox' Private Data Stamp printed on each page. The X and Y coordinates are used for placement on the page just as they are with page numbers. If the Private Data Stamp is to be used, a password must be obtained (see the CSL manager's secretary), and it will be necessary to consult the *Alto User's Handbook* since font sets other than those on the disk must be obtained.

Columns: 2 Edge Margin: 60 Between Columns: 40

If single column (full page) printing is to be used, this line can be omitted since full page is the default. If multiple column printing is wanted, type the number of columns. Edge margin sets the amount of white space on the *left and right* sides of the page. Between Columns sets the amount of white space between two columns of text. (See Double Column Printing, below.)

Margins: Top: 1.3" Bottom: 1" Binding: -13

The top and bottom margins may be set as desired. If binding appears, it assumes that the document is to be printed on both sides of the paper. Binding sets the amount of white space to be added to the inner side of each page, facing the binding--i.e., text is displaced along the binding by the distance set, making the margin *along the binding* slightly wider than the margin on the outside of the page.

The binding distance is calculated by first centering the text on the page--default margins (85 left and 527 right) center the text. Then the amount of displacement is decided based on the margin distance desired. E.g., margin distances left and right are 85 points (85 left margin; 612 full page -85 points = 527 right margin). If the margin distance along the binding is to be 98, and the margin distance along the outside 72, binding is calculated: 72-85= -13 binding distance.

Line Numbers: No Modulus: 5 Page Relative First Line: 1

If line numbers are requested, they will appear along the left margin. Modulus indicates the interval at which line numbers are printed--Modulus: 5 means, number every 5th line. Page-relative indicates that line numbers are to begin anew on each page. First line indicates the line at which line numbers should begin. Modulus: 5, Page-relative, and First Line: 1, will result in the first line of the document being numbered l; then the 5th line is numbered 5; the 10th line numbered 10, etc. And this numbering will begin again at 1 for each page of the document. If Page-relative is omitted, the lines will be numbered consecutively throughout the entire document.

Odd Heading: Not-on-first-page

The Odd Heading prints on odd numbered pages (1,3,5). If Not-on-first-page appears with Odd Heading and not with Even Heading, the first page to get a heading will be the second page of the document. A ↑CR is typed to end the Odd Heading line.

The actual heading typed after this Odd Heading directive can be centered, placed on the left margin, can be boldface, and/or in another font. A ↑CR is typed after the text of the heading. Heading text may consist of more than one line, but it cannot consist of more than one paragraph--normal CR's may be used, but two separate paragraphs with separate vertical tab settings (for top and bottom of a page, for example) may *not* be used.

Even Heading:

Bravo Course Outline

Even Heading works just like Odd Heading except that the heading text supplied will appear on even numbered pages only. The Even Heading line is ended with a ↑CR.

The heading text can be centered, boldface, and/or in a different font. A ↑CR is typed after the text of the heading. Heading text may consist of more than one line, but it cannot consist of more than one paragraph--normal CR's may be used, but two separate paragraphs with separate vertical tab settings (for top and bottom of a page, for example) may *not* be used.

C. Other Heading Options

A document profile with Odd and Even Headings assumes that the document will be printed on both sides of the paper--even page numbers on the left side; odd page numbers on the right side. When the document is *not* to be printed on both sides of the paper, Odd and Even headings are inappropriate--a simpler heading can be used:

Heading:
Bravo Course Outline

This heading consists of two paragraphs. Both are given the profile property (look;) and **Bravo Course Outline** is given a vertical tab setting of 10.5" if it is to print 1/2" down from the top of the page as default page numbers do. The headng text may be centered, boldface, and/or in a different font. Text for this heading, like odd and even headings, may consist of more than one line of text, but it cannot consist of more than one paragraph.

Page numbers beginning on the second page of the document are the default in Bravo, and with this type of heading directive, the default is for the heading to begin on the second page of the document also. Therefore, with these two document profile lines, the result will be that on the *second and following pages*, the heading will appear on the left margin, and the page number will appear on the right margin, both at the top of the page.

This type of heading can be combined with the other profile options. For example, if page numbering is to start at some point other than page 2, enter the line:

Page Numbers: Yes X: 527 Y: 10.5" First Page: 79

The Page Number/First Page option is useful when the length of a document dictates that it be prepared in two or more files; page numbering may be *set for the second file*, then when both files are printed, page numbering is consecutive.

D. Double Column Printing

To obtain double column printing Bravo must be given distance specifications for:

1. Edge Margin (amount of white space on the left and right edges of the page)
2. Between Columns (amount of white space separating the two columns of text)
3. Right margin setting in the document.

Edge Margin (em) and Between Columns (bc) settings are decided on the basis of preference. The right margin setting for text, however, must be calculated by first deriving the column width for text, and adding the left margin setting.

In the following explanation, edge margins are 60 points, and the between column distance is 40 points. Using these figures, the width of a text column can be calculated, and when that figure is known, the right margin to be set in the document can be calculated.

left edge (margin)	= 60
between columns	= 40
right edge	= 60
	160

The total space available across a page is 612 points (8.5" x 72pts/inch), less 160 (amount of white space) = 452 points of space remaining to accommodate 2 columns of text, and 452/2 = column widths of 226 points. So,

left edge (also left margin)	= 60
column width	= 226
right margin	286

Margins in the document are therefore set at 60 and 286:

lookleft60Esc
lookright286Esc

The document profile line reads:

Columns: 2 Edge Margin: 60 Between Columns: 40

Once margin and document profile settings have been assigned, Bravo will configure the text in two equal columns, break the pages and resume double column printing on subsequent pages.

Double Columns - Miscellaneous Information

1. To verify the distances, see that all 612 points of space are accounted for:

left edge	60
1st column	226
Between columns	40
2nd column	226
right edge	60
	612 points

2. If heading and page numbers are to be used, their placement should match the far right margin in effect for double columns. E.g., the X setting for page numbers should not be 527 (the default right margin); it should be set to correspond to the right margin of column 2--lm 60 + cw 226 + bc 40 + cw 226 = 552 right margin. Thus, for the example used above the X setting for page numbers would be 552.

   And the left margin set when heading *text* is typed in the document profile should be the same as the left margin set in the document--in the example above it is 60 points.

3. Either finish all text entry and editing before resetting the left margin at 60, or give the command windowedge60ESC. If one of these methods is not used, setting the left margin at 60 will truncate some of the text along the left side of the screen--text placed left from 85 points will *print* as specified, but it will not show up on the screen unless window edge is reset.

   [The window edge is set at 85 points--the default left margin--unless a new window edge boundary is specified. To restore the window edge to 85 points, give the command window edge 85 ESC. Note that applying a IF (page boundary) command or a look hardcopy command returns the window edge setting to 85.]

E. Double Column Printing - Centered Title

To center a title over Column 1, the title is simply typed and centered with the margins set appropriately--for the example above, margins throughout the document are set at 60 (left) and 286 (right). Bravo will center the title over Column 1, and continue with double column formatting.

However, if the title is to be centered across the full page with double column printing following, several additional formatting specifications must be made.

In the following example, the distances being used are: Left margin (lm) 60; column width (cw) 226; between columns (bc) 40; right edge (re) 60.

With multiple column printing a ↑L (l↑s) means go to the *next column*, not to the next page. Two consecutive ↑L's means go to the next page. To center a title across the full page with double columns following, proceed as follows:

1. The document profile reads:
   Columns: 2 Edge Margin: 60 Between Columns: 40

2. Margins for the *text* of the document are set at 60 and 286.

3. The title is inserted and centered with margins *for that paragraph only* set at 60 and 552 (full page centering). The right margin for full page centering is calculated by adding: lm 60 + cw 226 + bc 40 + cw 226 = 552, or by subtracting (em) 60 from 612 = 552.

4. Then a vertical tab is set on the title paragraph--look z 9.0" ESC, for example.

5. Decide how much text can fit in Column 1 based on how far from the top of the page it will begin to print--3" for this example. Then set a vertical tab on *the first paragraph* of Column 1 of 8". Where column 1 is to end, type a ↑L (l↑s) which will be read as a directive to go to the next column, not the next page.

6. Set a vertical tab on *the first paragraph* of column 2 of 8" so it will print at the same height as column 1. At the end of column 2 (you must determine where that is) type two consecutive ↑L's (l↑s twice), which will be read as a page break directive.

7. If standard double column formatting is to proceed from page 2 throughout the remainder of the document, nothing further need be done. The document profile reads: Columns: 2 Edge Margin: 60 Between Columns: 40, and the margins in the text are set at 60 and 286--that is all the information Bravo needs to format double columns.

More information and several hints about starting a file "from scratch" is contained in Appendix B.

XVIII. Calculator

Bravo has a simple calculator. For full details about it see the *Alto User's Handbook*. This section describes only its most standard features.

The calculator can be used either by typing numbers directly into it from the keyboard, or by selecting numbers within the file and *entering* them into the calculator.

A. From the Keyboard

At any point *while Bravo is in a ready state*, numbers can be typed to the calculator; no command need proceed the number. When the first numeral is typed, the system window reads:

NUMBER

and the number typed is displayed in the insert (middle) buffer of the system window. For example, 1500 is typed from the keyboard and it appears in the insert buffer. To enter it in the calculator, type \ (backslash, not /). After typing backslash (*enter*), 1500 is placed in the search (far right) buffer, where a running total is displayed.

To divide 1500 by 3, type:

3
/ (division operator)

and the total in the search buffer will change to 500.

To multiply the total (500) by 3, type:

3
* (multiplication operator)

and the total will change to 1500.

To subtract 500 from the total, type:

500
- hyphen (subtraction operator)

and the total changes to 1000.

To add 500 to the total, type:

500
+ or = (addition operators)

and the total returns to 1500.

To get a percentage of the total (1500), type:

5
%

and the total changes to 75.00.

B. Selections from a File

To enter a number from the file into the calculator, select it and type

\ (enter)

That number can then be affected by any of the operators--i.e., select another number in the file and type + or =, -, *, % or /. The total in the search buffer will display the new sum. Or a number can be typed from the keyboard followed by one of the operators and the search buffer will display the result. For example,

Select the number 500 in the file
type \ (enter)
select the number 3 in the file
type *

or:

Select the number 500
type \ (enter)
type the number 3
type *

In either case, whether 3 is selected from the file or typed from the keyboard, the search buffer will display the sum 1500.

C. Decimal Point Options

The number of digits after a decimal point is initially set at 2. However, it can be changed. 0-9 are the options. The command to change it is:

calculator
#

There are a variety of features not described here which can be found in the "Bravo Manual" of the *Alto User's Handbook.*

XIX. Maxc Users

Discussion of Maxc (PARC's timesharing computer facilities) is beyond the scope of this outline. However, *brief* mention of the File Transfer Program available between Maxc and your Alto does seem appropriate, along with information about printing Bravo files from Maxc. The *Alto User's Handbook* contains an *FTP Manual*, and the *Tenex User's Guide* describes many of the subsystems available on Maxc.

A. File Transfer Program

This discussion of FTP assumes that you have an account on Maxc and have done some form of login on the Alto.

There are a variety of uses for file transfers; several of which are:

1. To backup important files by storing them on Maxc.
2. To make Bravo files generally available to other users by putting them on Maxc.
3. To access relevant files from Maxc in order to supplement "forms" files; obtain newly implemented features; update existing Alto systems.

In fact, it would be difficult to use an Alto for an extended period *in the Parc environment* without needing to access or store files on Maxc.

The FTP (File Transfer Program) transfers a file on your disk to Maxc; or retrieves a file from Maxc and puts it on your disk.

To the Alto Executive type:

FTP MAXC (*type* the space preceding Maxc)

and then type a CR. After the CR is typed, the screen will be reconfigured into 3 windows, the middle one reads:

```
Alto Pup FTP User
*OPEN Maxc
<Maxc1 Pup FTP Server Version # Date
*
```

The last * is followed by a blinking bar, and it marks the point at which any keyboard command will be displayed.

Only two FTP operations are discussed here. They are store a file from your disk on Maxc; or retrieve a file from Maxc and put it on your disk.

Store

To store a file from your disk on Maxc, type:

s_ (s and one space)

FTP will complete the phrase:

sTORE local file

Then it will wait for a filename to be typed. Type the filename and a CR. FTP will then respond with:

as remote file filename

and will pause for confirmation. It repeats the filename in order to verify that the file is to be called by the same name when stored on Maxc. A CR confirms the request and FTP proceeds to transfer "a copy" of the file on your disk to Maxc--the file will reside in *both* places after completing the command.

FTP indicates what type of file it is transferring (text or binary), its length; then displays the message: Done. Another * with a blinking bar appears and you can either quit from the program or transfer more files. To quit, type:

q
CR

Quitting *automatically* returns you to the Alto Executive level.

Retrieve

To retrieve a file from Maxc, the same process is followed, except that:

r_ (r and one space)

is typed and the transfer proceeds in the opposite direction. The message printed is:

rETRIEVE remote file **filenameCR** as local file **filename**

Remote refers to Maxc; local refers to your Alto. These two terms will indicate in which direction the transfer is being made.

B. Printing Bravo Files from Maxc - Ears Version

If Bravo files are being transferred to Maxc for *backup and storage only*, nothing other than actually transferring them need occur. However, if you or someone else intends to send a file to Ears directly from Maxc, an Ears version must be created and transferred to Maxc before printing can occur

successfully. There is a hardcopy option in Bravo for creating Ears versions of Bravo files for printing from Maxc.

After typing h for hardcopy, Bravo responds:

HARDCOPY: type option (C,D,E,S,DEL) or CR to confirm

C is for copies; D is for Diablo printer; E is for Ears version; S is for "start at page #"; DEL aborts the command; and CR to confirm. Type:

e for an Ears version

Bravo responds:

Type filename for Ears format file, terminated by ESC
(Ears file, no printing)

Type the name of the file (usually the same *name* it had), and use the extension "Ears." E.g., the file created in Bravo is called: **Memo.Bravo.** The filename for an Ears version is: **Memo.Ears.**

After the filename and ESC are typed, Bravo makes an Ears version of the file and puts it on your disk. When that action is complete, Bravo responds with:

READY: Select operand or type command
Last page printed: #

The "last page printed" message is misleading in that the file *has not been printed.* In connection with creating an Ears version, this message should be taken only as confirmation that the operation is complete.

An Ears version is useful *only for printing from Maxc.* It cannot be edited in Bravo or in any editing system on Maxc. Ears versions also take up more space on a disk than their Bravo source files. Therefore, since the Ears file cannot be edited in Bravo and can only be printed from Maxc, once it is transferred to Maxc, it should be deleted from your Alto directory.

C. Send Message

The MSG subsystem on Maxc is useful for sending and receiving on-line messages. Documentation for MSG can be obtained by printing the file: <Doc>MSG.Doc. *From Maxc,* the command to print the file is:

@Ears <Doc>Msg.Doc followed by a CR.

Conclusion

The Outline has attempted to cover Bravo's fundamental features, but it has not and cannot describe either the vast number of possible combinations, or the variety of ways to achieve similar results. Experimentation and continued use of Bravo will reveal both.

Appendices

A: Font Summary Sheet

B: Starting a File "from Scratch"

C: Hardcopy (Printing) Options

Font 5: Times Roman 12

ABCDEFHIJLMNOPQSTUVXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()=+\|[]{}←↑;:'",.<>/?

ABCDEFHIJLMNOPQSTUVXYZ
abcdefghijklmnopqrztuvwxyz
!#$~&( )=+\/[ ]{ }↑;:'",.< >/?*

ABCDEFHIJLMNOPQSTUVXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()=+\|[]{}↑;:'",.<>/?

Font 0: Times Roman 10

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&( )-=+\/[ ]{ }←↑;:'",.< >/?*

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

Font 1: Times Roman 8

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&( )-=+\/[ ]{ }←↑;:'",.< >/?*

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

Font 9: Helvetica 18 (bold only)

ABCDEFGHIJKLMNO
PQRSTUVWXYZ
abcdefghijklmno
pqrztuvwxyz
!@#$%~&*()-=+\|
[]{}←↑;:'",.<>/?

Font 6: Helvetica 10

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&()-=+\|[]{}←↑;:'",.<>/?*

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

Font 7: Helvetica 8

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&()-=+\|[]{}←↑;:'",.<>/?*

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

Font 8: Gacha 10 (fixed pitch, no bold or italic)

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrztuvwxyz
!@#$%~&*()-=+\|[]{}←↑;:'",.<>/?

Font 2: Logo 24

XEROX

Font 3: Math 10 (no italic or bold)

∀∈ℂ∇∃‡⊂⊄⊆⊃⊅⊇)∉∅∝⇄ℝ≈⊥∪∨≡×↴▸
ℵ↷©∂≃↔⇒ℏ⌜⌝⌞⌟‖¬∘⊢⊨®⊣↖↙↘≪≫∩⊣
†%∞¢÷≒∧˙∓≠±/½<>¼¾→↓∎§≐°∋.˙.≤≥∅¿

Here is the keyboard layout for Math:

1	2	3	4	5	6	7	8	9	0	-	=	\
†	%	∞	¢	÷	≒	∧	˙				±	½
□	△	◇	⊕	⊖	⊗	∠	★	•	○	∓	≠	/

Q	W	E	R	T	Y	U	I	O	P	[	]	←
⇄	≡	∃	ℝ	⊥	↴	∪	⊆	∅	∝	¼	¾	↓
⊨	⋘	≃	®	↖	∩	↙	⌜	∘	⊢	<	>	→

A	S	D	F	G	H	J	K	L	;	'
∀	≃	∇	‡	⊂	⊄	⊃	⊅	⊇	§	°
ℵ	↗	∂	↔	⇒	ℏ	⌝	⌞	⌟	∎	≐

Z	X	C	V	B	N	M	,	.	/
▸	×	ℂ	∨	∈	¢	)	≤	≥	¿
⊥	⋙	©	↘	↷	¬	‖	∋	∴	∅

Font 4: Hippo 10 (no italic or bold)

ΑΒΞΔΕΦΓΗΙΚΛΜΝΟΠΘΡΣΤΤΩΧΨΖ
αβξδεφγηισκλμνοπθρστυωχψζ
!@#$%~&*()-=+\|[]{}←↑;:',./

Here is the keyboard layout for Hippo:

1	2	3	4	5	6	7	8	9	0	-	=	\
!	@	#	$	%	~	&	*	(	)		+	|
1	2	3	4	5	6	7	8	9	0	-	=	\

Q	W	E	R	T	Y	U	I	O	P	[	]	←
Θ	Ω	E	P	T	Ψ	T	I	O	Π	{	}	↑
θ	ω	ε	ρ	τ	ψ	υ	ι	ο	π	[	]	←

A	S	D	F	G	H	J	K	L	;	'
A	Σ	Δ	Φ	Γ	H		K	Λ	:	
α	σ	δ	φ	γ	η	ς	κ	λ	;	'

Z	X	C	V	B	N	M	,	.	/
Z	X	Ξ		B	N	M			
ζ	χ	ξ		β	ν	μ	,	.	/

Appendix B: Starting a File from Scratch

When Bravo is called, the end-of-file marker is the only thing displayed in the document window, and it is selected. Either an insert or append command with Bravo in this state will allow text to be entered.

A document requiring only page numbers--no other document profile option--does not require a document profile line for page numbers. The pagination default is that any document containing no directives will be numbered consecutively beginning with page 2.

Bravo considers paragraphs to be the areas between two ↑CR's. It will not respond to paragraph commands until at least one ↑CR has been typed. For example, when the end of file marker is on the same line as the text being typed, no paragraph commands (like ↑c to center) will be accepted. Therefore, an insert command given with only the end of file marker on the screen must either begin with a ↑CR, or the first paragraph must be typed and ended with a ↑CR before Bravo will respond to paragraph formatting commands.

When a file is started from scratch it is sometimes easier to enter all paragraphs requiring individual formatting before returning to give formatting commands. E.g., the following format:

Bravo Course Outline
(3" from top of page)

Introduction
(60 points down from title)

Text of first paragraph
(24 point paragraph break)

Text of second paragraph
(12 point paragraph break)

can most quickly be set up by typing all of the paragraphs without stopping to format them, then issuing individual formatting commands until the point at which the format standardizes--with the second *text* paragraph, for example. When starting from scratch, no paragraph breaks are specified; therefore, a ↑CR does not provide any white space between paragraphs until some y setting is given to at least one of the paragraphs. (In fact, no formatting is set--paragraphs have no breaks assigned, justification is not in effect, etc.) When the first four paragraphs vary widely in the y setting desired, it is best to input all of them, then return and set the paragraph format individually--centering one and assigning a

vertical tab, justifying and setting y distances on others, etc. Any paragraph carrying a vertical tab setting *must* be set individually to avoid appending paragraphs which will carry the same vertical tab and thus cause overprinting.

If the document is to have highly structured levels of formatting, it *may* be possible to set up things like tab requirements early in the document, then use them throughout the document without stopping to set them at each level. For example, the levels of formatting are:

I. Text (tab at 120)

A. Text (tab at 145)

1. Text (tab at 170)

All of these tabs can be set very early in the document since *setting* named tabs has no effect unless they are *used*, but once set they remain in effect for appended paragraphs and can be used when needed.

This approach assumes fairly careful planning before the document is typed. However, a few minutes spent setting up a consistent format will save lengthy editing sessions later.

When there are several formatting levels, it may be useful to set up sample paragraphs with text describing their format, and have these paragraphs at the end of the document. These formats can then be adopted easily with a copy selection to the look all command. Having these sample paragraphs displayed in a subwindow further facilitates formatting since there is no need to scroll the document to locate desired formats.

Appendix C: Hardcopy (Printing) Options

The hardcopy command has 6 options, four of which were discussed in the Outline: multiple copies, CR for one copy, DEL to abort the command; and ears version.

When hardcopy is initiated, Bravo prints the message:

HARDCOPY: Type option (C,D,E,S,DEL) or CR to confirm

The options yet to be described are: D for Diablo printer; and S for "start at page #".

The sequence for "start at page #" is:

hardcopy

HARDCOPY: Type option (C,D,E,S,DEL) or CR to confirm

s

HARDCOPY: Type page # to start print at, terminated by ESC

4ESC (for example)

HARDCOPY: Type option (C,D,E,S,DEL) or CR to confirm

CR to send file.

When "start at page #" is used it takes some time for Bravo to go through the document and figure out where the page specified begins. Once that is accomplished, the file will be printed starting at that page and continuing to the end of the document. This option is useful with the Diablo printer since printing the entire file proceeds more slowly than with Ears.

Diablo Printer

The D option is for Diablo printer. To use this option, make sure paper is loaded in the printer and positioned correctly for printing to begin, then after initiating the hardcopy command, type:

d

HARDCOPY: Type option (C,D,E,S,DEL) or CR to confirm
Diablo Output

CR

HARDCOPY: Type command (C,R,DEL) or blank to proceed
Ready to print page 1

Typing a space (blank) at this point will cause the first page to be printed. C will initiate continuous printing (presuming perforated paper is loaded), and R is to "reprint last page." Unless continuous printing is in effect, the message will be displayed after each page is printed, providing time for the next sheet of paper to be loaded--type a space to print the next page, an r to reprint the last page or discontinue the entire command with DEL. However, if continuous printing has been initiated, the entire file will be printed without interruption.

If text is underlined on the screen, the Diablo printer will underline it; it double underlines text which appears in italics on the screen; it completely ignores text which is in a different font on the screen, but leaves the amount of blank space that text would require *if* printed; all margin settings are honored; text with a vertical tab setting will be placed as specified; page breaks occur normally; and headings and page numbers are printed as requested.

Summary of Commands

Contents

Introduction

The summary lists Bravo commands in two groups: single character keyboard commands are listed in alphabetical order; and all look commands are grouped alphabetically, with special character looks (; ? -, etc.) at the end of the Look command section.

Almost no explanation of commands is provided in the Summary; only the sequence of keystrokes required to execute each command is given. However, each command is followed by page numbers, referencing the point in the outline where further explanation is available. Thus, the summary may be used either as a quick reference for a particular command sequence, or as a partial index for the outline.

I. Keyboard Commands

Append

Make a selection
append
type text to be added
ESC

See:

append text string, p. 17
append paragraph, pp. 21-22
append with secondary selection, p. 63

BravoBug

To control the replay, use the following commands:

space	Stops the replay. After the replay is stopped, each space causes *one* command to replay.
proceed	Resumes replay at full speed.
break before command #	Interrupts the replay at the command number specified.
terminate CR	Terminates the replay.
quick	Causes typed text to go in all at once. Quick is default mode--it is not needed unless slow has been used.
slow	Slows down the replay so text goes in character by character, as it was typed initially.

See Section VIII, pp. 35-37.

Buffer

To make one of the three buffers *current*, type:

buffer
(1-3)
ESC

After typing the buffer number, there is an opportunity to type or select the text the buffer is to contain. If ESC is typed without taking advantage of this option, the buffer, with the text it currently contains, is made current.

See Section XIII, pp. 62-63

Delete

Make selection
delete

See pp. 16-17

Diablo Printer Commands

See Appendix C, pp. 95-96.

Document Profile

See Section XVII, pp. 79-85.

Double Column Printing

See Section XVII, pp. 83-85.

Everything

To select the entire document, type:

everything

See p. 32.
Also see everything normalize, p. 33.

Find

find
type text to be found
ESC

Begins search from end of the current selection. When text is found, it is placed at the top of the document window, and it becomes the current selection.

To make selective substitutions, proceed as follows: When text string is found and selected,

replace
type text desired
ESC

Then type:

find
ESC

to locate next instance. From then on, type either:

yes to make the same substitution

or type:

ESC to make no substitution and go to next instance.

See Section XIV, pp. 65-66.

Get

To read one of the files on your directory into Bravo:

get
type or select filename
ESC

See Section I, pp. 3-4.

Hardcopy (Print a file)

Multiple copies:

hardcopy
copies
(of copies)
ESC
CR

Single copy:

hardcopy
CR

See Section I, pp. 9-10.
Also see Appendix C for Diablo Printer options, pp. 95-96.

Insert

Make selection
insert
type text to be inserted
Esc

See:

insert text: p. 17.
insert paragraph: p. 21-22.
insert with secondary selection: p. 63.

Jump

To jump to a specific text string in the document, type:

jump
type text string
ESC

Jump begins its search with the second line on the screen and proceeds to the end of the file; it does not "wrap around"--start again at the beginning.

See Section XIV, p. 64.

Kill

To destroy or clear a new or subwindow, type:

kill

followed by one of the following mouse buttons:

Red	With cursor in bottom window, destroys bottom window, gives full screen to top window.

Yellow With cursor in top or bottom *new* window, clears contents, leaves window. With cursor in *either* subwindow, clears entire screen of all text, leaves subwindows.

Blue With cursor in top window, destroys top window, gives full screen to bottom window.

See Section XV, pp. 68-74
Also see **Window** in Command Summary.

LF

See **Page Boundary**

Mouse Functions

There are three *sets* of mouse functions:

1. To select strings of text:

   Red Selects a single character or space.

   Yellow Selects a word.

   Blue Extends a selection through single characters or words, depending on the initial selection.

2. Scroll Bar:

   Red Moves line of text opposite cursor to top of document window.

   Yellow Scrolls to a relative position in the document--cursor at top of document window scrolls the document to the beginning; cursor at bottom of document window scrolls the document to the end; cursor in the middle of the document window scrolls the document to middle, etc.

   Blue Moves line at top of document window down opposite cursor position.

3. Line Bar:

   Red Selects the *line* of text opposite the cursor.

   Yellow Selects the *paragraph* opposite the cursor.

Blue	Extends the selection through lines or paragraphs, depending on the initial selection.

See:
Text selection, pp. 4-6.
Scroll Bar, pp. 14-16.
Line Bar, pp. 19-20.

Normalize

To bring selected text (no matter where it is in the document) to the top of the document window, type:

normalize

See p. 32.

Page Boundary

To initiate the Page Boundary option, type:

LF Key

To cancel Page Boundary, type:

window
destroy
place cursor in bottom window
push *any* mouse button

Then type:

everything
lookH (uppercase H)

See Section XV, pp. 73-74.

Put

Put
type or select filename
ESC

To store a new version of the *same* file--i.e., use the same filename, type:

Put
ESC

See pp. 8-9 (Put), and pp. 17-18 (Put ESC).

Quit

quit (from Bravo or from the Alto Executive)
CR

See p. 10.

Replace

Make selection
replace
type replacement text
ESC

See pp. 6-7.

Substitute

Select the amount of text to be affected
substitute
type text string desired
ESC
type text string to be changed
ESC

See Section XIV, pp. 64-65
Also see **Find** (selective substitutions) in Command Summary.

Undo

To cancel the effects of the *last* command, type:

undo

Undo works only on the last command given, and only if the selection has not been changed.

See p. 31.

Window

After typing:

window

The mouse buttons work as follows:

Red	Moves boundary of new or subwindow--hold down **Red** and move cursor slowly to desired boundary. When **Red** is released, boundary will freeze.
Yellow	To create a subwindow
Blue	To create a new window

See Section XV, pp. 68-73.
Also see **Kill** in Command Summary.

Window Clear

Typing:

window
clear

clears the document window of all text and simultaneously clears the filename from the black border below the system window.

See p. 32.
Also see pp. 72-73 (multiple windows).

Window Edge

The window edge (left margin of the screen) is set at 85 points unless it is specifically changed. To change the window edge, type:

window
edge
60 (for example)
ESC

To restore window edge to the default, use the same command, with 85 for the distance.

See p. 84.

Window Destroy

To get rid of a new or subwindow, type:

window
destroy

and use *any* mouse button to select the window to be destroyed.

See pp. 72-73.
Also see all of Section XV, pp. 68-74.

Window Move

To move the boundary of a new or subwindow, type:

window
move

and use *any* mouse button to trace the new boundary.

See pp. 72-73
Also see all of Section XV, pp. 68-74.

Window New

To create a *new window*, type:

window
new

and use *any* mouse button to indicate the desired new window boundary.

See pp. 72-73.
Also see all of Section XV, pp. 68-74.

Window Split

To create a subwindow, type:

window
split

and use *any* mouse button to indicate the desired subwindow boundary.

See pp. 72-73
Also see all of Section XV, pp. 68-74.

Look Commands

A brief description of each Look command is given below. The pages following contain individual command sequences for each Look and page numbers for more detailed explanation within the outline.

Look:

**	all	Alters paragraph format to match that of secondary selection.
	bold	Boldface text.
	BS	BS key for overprinting.
	center	Paragraph look for centering text.
	CLR	To cancel special *character* looks--n/a for clearing *paragraph* format looks.
*	down	To offset text down from baseline.
*	first	Paragraph look to control placement of first line.
	hardcopy	Duplicates line break and character placement when printed.
	italic	Italicized text.
	justify	To justify right margin.
*	keep	To keep blocks of text together--i.e., control where page breaks occur.
*	left	To set the left margin.
	nested	Indent paragraph by 36 points (1/2")--preset paragraph look.
	open	Adds 12 points of leading to paragraph break--preset paragraph look.
*	paragraph	To set margin for the second and following lines of a paragraph.
	q	Adds 6 points of leading to paragraph break--preset paragraph look.
*	right	Sets right margin.
**	same	Alters character style to match secondary selection.
*	TABkey	To set "named" or regular interval tab.
*	table	To set contiguous set of equidistant tabs.
*	up	To offset text up from the baseline.
	visible	To get spaces, tabs, CR's, etc. visibly represented.
*	x	To set leading between lines--6 pts. = double, 1 pt. = single spaced.
*	y	To set leading between paragraphs--12 pts is default leading.
*	z	To set a vertical tab. (See Part III)
	0-9	Font changes.
	-	Underlined text.
	↑	Superscripted text.
	←	Subscripted text.
	>	Change selected text to uppercase.
	<	Change selected text to lowercase.
	?	With paragraph selection displays paragraph settings; with character selection displays font and character looks.
	;	Sets profile property in document profile specifications. (See Part III)

* Require distance specification.
** Require a secondary selection.

II. Look Commands

All

Select paragraph(s)
look
all
select secondary paragraph
ESC

See p. 55-56.

Bold

Make selection
look
bold

To cancel boldface, type lookB.

See p. 19.
For boldface text *while typing*, see p. 22.

BS (Backspace for overprinting)

Make selection
look
BS key

To cancel overprinting, type lookShiftBS.

See p. 56
For overprinting *while typing*, see p. 56.

Center

Make selection
look
centered

To cancel centering, type look**C**.

See pp. 28-29.
For centering text *while typing*, see p. 30.

CLR

To cancel special character looks, select the text and type:

look
CLR key

See pp. 21.
To clear character looks *while typing*, see p. 22.

Down

Make selection
look
down
type distance in points that text is to be offset.
Esc

See pp. 54-55.

First

Select paragraph(s)
look
first
type or select horizontal position for first line of paragraph.
Esc

See Section XI, 50-54.

Hardcopy

look
hardcopy

To cancel hardcopy mode, type look**H**.

See p. 57.

Italics

Make selection
look
italics

To cancel italics, type lookI.

See p. 19.
For italicized text *while typing*, see p. 22.

Justified

Select pararaph(s)
look
justified

To cancel right justification, type lookJ.

See pp. 28-29.
For right justification *while typing*, see p. 30.

Keep

Select paragraph
look
keep
type distance
ESC

See pp. 33-34.
Also see Manual Page Break, p. 33.

Left

Select paragraph(s)
look
left
type or select horizontal position for left margin
ESC

See Section XI, pp. 50-54.

Nested

Select paragraph(s)
look
nested

Nesting is a preset paragraph indent of 1/2". To cancel nesting, type lookN.

See pp. 28-29
To nest paragraphs *while typing*, see p. 30.

Open

Select paragraph(s)
look
open

Look open adds 12 points of leading to the break preceding the paragraph. To cancel open, type lookO.

See pp. 48-49.
Use of open *while typing*, see p. 49.

Paragraph

Select paragraph
look
paragraph
type or select horizontal position for 2nd and following lines.
Esc

See Section XI, pp. 50-54.

q

Select paragraph(s)
look
q

q adds 6 points of leading to the break preceding the paragraph. To cancel q, type lookQ.

See pp. 48-49.
Use of q *while typing*, see p. 49.

Right

Select paragraph(s)
look
right
type or select horizontal position for right margin
ESC

See Section XI, pp. 50-54.

Same

Make selection
look
same
make secondary selection
ESC

See p. 56.

TAB key

To reset the interval at which *plain tabs* fall:

Select paragraph(s)
look
TAB key
=
type distance in points or inches for plain tab interval
ESC

To set specific ("named") tabs:

Select paragraph(s)
look
TAB key
type tab identifier (0-9 or a-e)
type or select horizontal position of tab stop
ESC

See Section XII, pp. 58-60.

Table

Select paragraph(s)
look
table
type or select left edge of table
Esc
type or select right edge of table
Esc
type number of columns
Esc

See Section XII, pp. 60-61.
look table caveat, p. 61.
Copy all Tabs Option, p. 61.

Up

Make selection
look
up
type distance in points that text is to be offset
Esc

See pp. 54-55.

Visible

Make selection
look
visible

Editing, storing the file, printing can be done in visible mode. To cancel visible display, type lookV.

See pp. 56-57.

X

select paragraph(s)
look
x
type distance in points for leading between lines of text
Esc

See pp. 50-51, 53.

Y

Select paragraph(s)
look
y
type distance in points for leading between paragraphs
ESC

See pp. 50-51, 53.

Z

Select paragraph
look
z
type distance in inches or points, measured from bottom of page
ESC

To cancel a vertical tab:

Select paragraph
look
z
\ (backslash, not /)
ESC

See Section XVI, pp. 75-78.
For use in document profile, see pp. 79-80.

0-9 (Font changes)

Make selection
look
type the number of the font desired (0-9)

To cancel a font, select the text and type either:

look
CLR

or:

look
type any other font number

See pp. 47-48.
For font changes *while typing*, see p. 47.

- (hyphen for underlining)

Make selection
look
-

See p. 19.
For underlining *while typing*, see p. 22.

↑ (superscripting)

Make selection
look
↑

See p. 46
For superscripted text *while typing*, see p. 46.

← (subscripting)

Make selection
look
←

See p. 46
For subscripted text *while typing*, see p. 46.

> (uppercase characters)

Make selection
look
>

See p. 33

< (lowercase characters)

Make selection
look
<

See p. 33

?

Make character or paragraph selection
look
?

With a character selection, information regarding the font and character look will be displayed in the system window. With a paragraph selection, information regarding margins, tabs, x and y settings, vertical tabs if any, etc. will be displayed in the system window.

? character, see p. 48.
? paragraph, see p. 54.

; (document profile property)

Select the entire document profile
look
;

To remove the document profile property:

Make selection
look
;

See p. 79.