DynaStar

Release 3.0

A High Speed Screen Editor
and Word Processing System
for the
OS-9 Operating System
on the
Radio Shack Color Computer

by

Allan G. Jost

DYNASOFT SYSTEMS LIMITED
P.O. Box 51
Windsor Junction, Nova Scotia
Canada B0N 2V0

TABLE OF CONTENTS

1/INTRODUCTION

DynaStar was written for all thos microcomputer users who, like myself, hav high speed video terminals and are tired o editors designed for 110 baud teletypes.

DynaStar is dedicated to the idea that tex editors should above all be easy to use. Th fundamental idea is that you put the curso anywhere you want on the screen, and then jus type, and what you see on your screen is wha you get on paper. It has two personalities when you are just learning, it constantl provides you with help in the form of menu showing you at all times what your option are, and when you have mastered it you can ru it in a mode that doesn't bother you wit unnecessary menus, giving you a full screen o which to do your editing. We hope the men system provided will help you to mak effective use of its features withou constantly thumbing through the manual. If w have failed, we would appreciate you comments.

The original version of DynaStar was writte in 1982 specifically for the OS-9 operatin system. DynaStar Version III is a specia adaptation of DynaStar especially for th Radio Shack Color Computer. It is designed t work in conjunction with the high resolutio graphics/text screen that comes with FH O-PAK, and incorporates horizontal scrollin giving it the ability to work with text a wide as 250 columns, making it superior to th original version which was written for muc more expensive computers.

DynaStar was designed to meet the needs o both the word- processing user and th programmer. For use as a word-processor, i contains several features which are intende to support facilities in the companio Dynaform print formatter, availabl separately. These features are only mentione briefly in this manual and are explained i detail in the DynaForm's User's Manual.

DynaStar is written entirely in Dynasof Pascal, which is a p-code implementation of a integer subset of Pascal. Many people believ that p-code is too slow for applications lik screen editors. Judge for yourself.

1.1.0 A Guided Tour

Running DynaStar

The best way to learn about DynaStar is to us it. Install it on your system disk (see th instructions in Chapter 5), make sure you ar using one of the special hires screens wit O-PAK, and type "Ds" to bring the editor int memory and start it running.

1. Run "*"
2. <B>
3. <ESPACE>
4. <date-heure>
5. TMODE -UPC
6. <CLEAR><Ø>
7. Insérer disk "Dynastar"
8. Chx /dØ/CMDS
9. Cs

If all has gone well, you should be presente with a copyright message and a menu (th "files" menu) offering several options. Fo now just type the letter "N" to start a Ne edit session. On the top line of the scree you will be asked for a file name. Pick one that is not already in your working directory, and type it in, followed by a carriage return. After a brief pause, you should be presented with a new menu (the main editing menu).

Before doing anything else, sit back and get familiar with the organization of the information at the top of the screen. The very top line should contain the name of the file you have just established, and a word at the right hand end telling you that you are in "INSERT" mode. This line is called the "status" line, and its main purpose is to constantly remind you what file you are working on, and what "modes" you are in. It is also where most operator prompts will appear when required by certain commands, and where you should look for error messages when your computer "beeps" at you.

The Ruler Line

The next five lines contain a menu of commands available to you for editing text, and below this will be a line with lots of minus signs and little triangles. If there are no triangles you are using an incorrect hires screen. This is the "ruler" line which serves the dual purpose of separating the text editing area from the menu area, and telling you where your margins and tabs are. You will notice that every 10 columns along on the ruler line the digits 1, 2, 3, etc. appear marking the locations of columns 10, 20, 30, etc. Also on the ruler line (directly above the current cursor location on the screen) you will see a little solid block. This is in fact a second cursor which moves along the ruler line in step with the real cursor. The block and the digits serve as reference points so you can always figure out what column the cursor is on (something like the scale on an ordinary typewriter). The triangles indicate the location of tabs, which can be easily set and cleared.

To learn how to edit text, you must have text to edit. Type in a few lines of anything you like just pretending the keyboard is a standard typewriter. Fill the whole screen if you like.

Cursor Commands

After you have enough on the screen to make it interesting, sit back and have a look at the menu. It will tell you very briefly about alot of single letter commands you can give to manipulate the text you have just typed. Most of the command letters have a "^" in front of them. What this tells you is that you must hold the "CLEAR" key down while typing that letter in order to convert it from a "typable" letter to an editor command. The first two lines of the menu tell you how to move the cursor around. To get the feel of it, try typing ^E ("control" E) and watch the cursor. It should move straight up. Then try typing ^S, ^D, and ^X (any order) and watch which way the cursor goes. Note that these four keys are arranged on the keyboard in a diamond pattern which reflects the cursor movement they cause. Learn to know and love this diamond. You will use it alot.

You will notice that the cursor will not move to where there is no text. This is one of the fundamental rules of this editor. The cursor can only be placed over characters which are actually in your text buffer. To get it past the last character in the buffer, you must type in more characters. To get it past the last character in a line, you must type more characters into the line. When it is sitting on an empty space at the end of a line, it is actually sitting on an invisible carriage return character. If you type more characters while it is sitting on a carriage return, it will insert the characters in the line, pushing the carriage return ahead of it.

Deleting Characters

Now look on the fourth line of the menu (look for the word "DELETE"). Place the cursor over any character on your screen and type ^G. The character should disappear and the line should close in from the right. Now press the backspace (back arrow) key. The character immediately to the left of the cursor should disappear, and the cursor should back up one position. Place the cursor over the first character of any word and type ^T. You should get the idea. Now retype the word that just disappeared and watch it being put back where it came from. For your last trick, get the cursor over the very last character of a line (it should be an invisible carriage return) and delete it with a ^G. Now, to split those two lines apart again, type a carriage return.

That should be enough to get you started. Play with the other commands on the menu to see what they do.

Supplementary Menus

You will observe on the last line of the menu that you can call up additional menus by typing ^B, ^K, ^P or ^Q. Try typing ^Q and watch the menu area. You will be presented with a whole new list of commands. To execute one of these commands you will have to type another letter. (For the second character it doesn't matter if you hold down the "CLEAR" key.) ^B, ^K, ^P and ^Q are "prefix" commands, meaning they are actually the first letter of a two-keystroke command. If you don't want to try any of these commands just now, press the space bar.

Exiting DynaStar

To wrap up your first lesson, make sure you are on the main editing menu and type ^N (the menu calls this one "doNe"). You should find yourself back on the "files" menu, which is where you started in the first place. Type the letter "D". After a few seconds, you should see a listing of the file names in your current data directory. One of them should be called "SCRATCHnn", where nn is a small number like "03". Now type "X" (for "eXit"). You should get the familiar "OS9:" prompt. Type "dir". Notice that "SCRATCH" has disappeared, but that there is now a new file there with the name you picked way back at the beginning of this exercise.

2/DESCRIPTION OF EDITING COMMANDS

All of the commands available in the DynaStar Editor are described below. They are covered a menu at a time, in approximately the order they should be learned by the beginning user.

2.1.0 Operating Modes

Insert Mode

It is first necessary to explain the two basic operating modes of DynaStar. When you first start an editing session, the editor is in "INSERT" mode, which you can verify from the right end of the status line. In this mode, any printable character you type is inserted in your text buffer at the current cursor location, "pushing" any text to the right of the cursor ahead as it goes. It is possible to operate all of the time in this mode, since text can be replaced by deleting old text and then typing in new text to replace it.

Overtype Mode

The other basic mode is "OVERTYPE". You can put the editor in this mode by typing the sequence ^QO (this notation means "hold the CLEAR key down while pressing the letter Q, then type the letter O"). The word "OVERTYPE" will appear on the end of the status line when you do this. In this mode, printable characters that you type are typed "over" and replace the characters that were there before.

OVERTYPE mode is most useful to correct simple spelling errors in text, or when working with columns of figures. The biggest problem is that you tend to forget you are in OVERTYPE mode, which sometimes leads to confusion. It is just as easy to correct isolated spelling errors from INSERT mode by first deleting the offending character and then typing in the correction, and this method usually involves fewer keystrokes than entering OVERTYPE mode typing a one character correction, and then re-entering INSERT mode. At Dynasoft w rarely use OVERTYPE for this reason. ^QO is "toggle" command: it switches back and fort between OVERTYPE and INSERT mode each time yo type it.

Borizontal Scrolling

It is possible in either "INSERT" or "OVERTYPE" mode to type past the right edge of your screen. When this happens, the entire typing area of the screen will shift left ("scroll") by eight characters so that the cursor stays on the screen. In fact the viewing area will always shift left or right as required whenever you attempt to move the cursor off the screen in either direction. Using this horizontal scrolling feature, DynaStar can edit text as wide as 250 columns.

Word Wrap Mode

There is another mode in DynaStar which is extremely useful when typing paragraphs of text. This mode is called "word wrap", and it is explained more fully a little later. When in this mode, attempts to type a word beyond the right margin (not to be confused with the right edge of the screen) will move the entire word to a new line, just as if you had typed a carriage return (i.e. pressed "ENTER") before you started the word. This means that you need never use the "ENTER" key while typing paragraphs. Word wrap mode is not active until you "turn it on": see section 2.4 for how to do this. We mention it now because it is important for you to understand the difference between automatic scroll (shifting the screen to make room to keep typing on a line) and word wrap (automatically starting a new line when a word won't fit within the right margin). It is also important to understand the difference between the right margin, which is defined by the ruler line, and the right edge of the screen, which is simply the edge of an imaginary viewing window onto a page.

2.2.0 Basic Cursor Motion Commands

Control Diamond

Most of the control keys on the main editing menu are arranged in a geometric pattern on the left side of the keyboard which reflects the function performed by each command. This pattern is centered around the basic cursor motions of up, down, left and right, which are arranged in a simple diamond pattern.

```
                      /\
                     /  \
      ______________/    \______________
     |    W       /  E  \       R      |
     |  line     / char  \   screen    |
     |   up     /   up    \    up      |
  ___|_________/___________\___________|___
 |    A      /  S    |   D  \       F      |
 |  word    / char   |  char \    word     |
 |  left    \ left   |  right/    right    |
 |___________\_______|_______/_____________|
     |    Z   \      X       /  C      |
     |  line   \   char     /screen    |
     |  down    \  down    /   down    |
     |___________\        /____________|
                  \      /
                   \    /
                    \/
```

The keys adjacent to the four basic cursor motion keys perform functions related to the basic motions. For instance, while ^S moves the cursor left one character, ^A moves it left one word. Similarly, ^D moves right one character and ^F moves right a word. On the top line, ^E moves the cursor up one character (to the line above), and ^R moves up one screenful. On the other side of E, ^W does not move the cursor, but it scrolls upward one line in the file (the screen actually scrolls down). In the same way, on the bottom line, ^X moves the cursor down one character, ^C moves down a screenful, and ^Z scrolls one line downwards.

Deletion Keys

In addition to the cursor motion keys there are several keys for deleting text. ^G deletes the character under the cursor. Backspace ("<-") or ^H deletes the character to the left of the cursor. ^T deletes the word to the right of the cursor, and ^V deletes the word to the left (useful when entering text to back up and retype a mispelled word). The sequence ^QY deletes everything on the line to the right of the cursor, and ^Y deletes the entire line.

Screen and Paragraph Movements

Prefixing any of the four basic motion keys (^E,^S,^D,^X) with a ^Q moves the cursor as far as it will go on the screen in the direction corresponding to the basic key. Prefixing ^R and ^C with a ^Q moves the cursor to the very top and bottom of the edit buffer respectively.

Response Commands

Some commands prompt for additional input if necessary on the status line of the display. Commands requiring a single- letter response continue as soon as the letter is typed, and do not care if the reply is typed in upper or lower case. Some commands require a multiple-character response such as a file name or search string. The response to these prompts can usually be up to 40 characters in length, and is terminated by pressing the ENTER key (which is not considered part of the response string). If you accidentally type an error while typing such responses you can correct it by using either the ^S, ^H, or BACK SPACE keys to back up to the offending character. Typing an "empty" response (just "ENTER") to one of these prompts will generally cancel the command.

Errors

Error conditions detected during operation of any of the editing commands will usually result in an error message on the status line, in conjunction with an audible beep from the computer. To exit from an error condition, you must press the CLEAR and BREAK keys at the same time. This combination is referred to as "ESC". Pressing any other key will result in another beep.

2.3.0 Main Edit Menu

Descriptions of the editing commands on the main editing menu follow:

^S cursor left

Moves the cursor one character to the left if possible, moving it to the end of the previous line and scrolling the screen down if either action is necessary.

^D cursor right

Moves the cursor right one character, moving to the beginning of the next line if necessary, and scrolling the screen up if the cursor is on the last line of the screen.

^A left word

Moves the cursor left to the first letter of the preceding word. For purposes of this command and others involving "word boundaries", the cursor also stops on all punctuation marks and at the beginning and end of lines.

^F right word

Moves the cursor to the beginning of the next word to the right, as described above.

^I or TAB tab right

Moves the cursor right to the next tab stop. In OVERTYPE mode, this is a simple cursor movement. In INSERT mode this command inserts blanks to get to the next tab stop, pushing any text ahead of the cursor over in the process.

^E cursor up

Moves the cursor up to the line above its current location. The screen is scrolled down if necessary to allow this. If the line above ends to the left of the cursor location, the cursor is moved left to the end of that line.

^X cursor down

Moves the cursor down to the same location, if possible, on the line below its current location, as described above.

^R up screen

Scrolls upward in the text buffer one "screenful", leaving two lines of overlap with the old display. If the cursor is in the overlap region, it remains over the same character, otherwise it will land on the last character displayed on the new screen.

^C down screen

Scrolls downward in the text buffer one screenful, as described above.

^Z scroll up

Scrolls the screen up one line, leaving the cursor over the same character if possible.

^W scroll down

Scrolls the screen down one line, leaving the cursor over the same character if possible.

^G delete character

Deletes the character under the cursor, closing in the line from the right.

BKSP ("<-")or^H delete left

Deletes the character immediately to the left of the cursor, backing up the cursor and closing in the line in the process.

^V delete word left

Deletes the characters to the left of the cursor, up to and including the first character of the current word. This command is most useful for correcting a word that has just been typed, if it would be quicker to retype the word than to back up to the character(s) in error and then space back to where you were.

^T delete word right

Deletes the characters in the current word to the right of the cursor.

^Y delete line

Deletes the entire line on which the cursor is sitting.

^J line insert

Inserts a carriage return at the cursor location, and leaves the cursor just before the CR. Any text to the right of or below the cursor is pushed down one line. If this is issued when the cursor is on the left margin it has the effect of inserting a blank line into which new text can be typed. If it is issued in the middle of a line, it splits the line, leaving the cursor on the end of the top piece.

^Q supplementary prefix

Prefix command bringing up a supplementary menu containing additional cursor movement and find/replace commands.

^B block prefix

Prefix command bringing up the block operations menu.

^K formatting prefix

Prefix command bringing up the formatting and tab setting menu.

^P Print control prefix

Prefix command for inserting print control characters for the DynaForm Print Formatter. The sequence "^P." is used to display additional menus summarizing DynaForm's "dot" commands.

^N doNe

Return to the "files" menu.

ESC macro learn or error escape

This key is used to enter or leave the "macro learn mode". Keyboard macros are fully explained in the chapter "KEYBOARD MACROS". The ESC key is also used to exit from error conditions. On the Color Computer, ESC is generated by pressing CLEAR and BREAK at the same time.

2.4.0 Supplementary Menus

In addition to the main editing menu Dynastar supplies four supplementary menus. Each menu provides you with additional commands to simplify the task of editing. You may enter any of these of menus from the main menu by typing a ^Q, ^B, ^K or ^P. Each menu is discussed below.

2.4.1 ^Q Supplemetary Menu

^Q is a prefix command which may be thought of as the first character of a two-character command sequence. To execute the command a second character must be typed after the ^Q. The second command letter can be typed with or without the CLEAR key and in upper or lower case. If the prefix was typed accidentally, it may be cancelled by pressing the space bar. Most of the ^Q commands are supplementary cursor moves.

^QE top of screen

Moves the cursor to the first character location on the screen: (i.e. to the upper left corner).

^QX bottom screen

Moves the cursor to the last character on the screen.

^QR top of buffer

Moves the cursor to the first character in the text buffer, re-displaying the screen if necessary.

^QC bottom buffer

Moves the cursor to the last character in the text buffer, re-displaying the screen if necessary.

^QS left margin

Moves the cursor to the left margin on the current line.

^QD right end

Moves the cursor to the last character on the current line.

^QP next Paragraph

Moves the cursor to the beginning of the first word of the next paragraph. A new paragraph is recognized by any of the following conditions: 1) one or more blank lines. 2) a line indented past the "wrap margin". 3) a line beginning with a period (a DynaForm "dot" command).

^QT top of block

Moves the cursor to the first character of the currently marked block, if a block is still active. Block marking is discussed under the ^B block prefix menu.

^QB bottom block

Moves the cursor to the last character of the currently marked block, if a block is active.

^QF find string

Prompts for a search string on the status line (up to 40 characters, terminated by ENTER) and searches the text buffer, from the cursor location forward, for a string matching the search string. If such a string is found, the cursor is moved to the first character of the found string, and the screen is re-displayed if necessary. If the string is not found, the message "NOT FOUND" is displayed on the status line, and pressing CLEAR-BREAK will return the cursor to its original position.

^QL repLace string

Prompts for a search string as in the "find" command, and then prompts for a second string which is to replace the next occurrence in the text buffer of a string matching the search string. It first performs a search in the same manner as the "find" command, producing the same "NOT FOUND" message if the string is not found. If a match is found, the message "Replace? (y/n)" is displayed on the status line, and the cursor is placed on the first character of the found string. At this point, if "y" or "Y" or "^Y" is typed, the found string will be deleted and replaced with the replacement string, otherwise the replace is cancelled and the cursor is left on the found string. This command can be used to delete a string by providing an "empty" replacement string (press ENTER only).

^QA Again

Repeats the last "find" or "replace" operation without re-prompting for the search and replacement strings. The same error messages may result, and confirmation is requested before any replacement is performed.

^QY delete line right

Deletes all characters in the current line to the right of the cursor.

^QO Overtype on/off

Turns OVERTYPE mode on if currently in INSERT mode; turns INSERT mode on if currently in OVERTYPE mode.

2.4.2 ^B Block Commands

The ^B prefix activates a set of commands for manipulating blocks of text. To perform a block operation it is first necessary to "mark" a block of characters by placing the cursor over the first and last characters and "marking" these characters as either "top" or "bottom". When either the top or bottom of a block has been marked, but not the other end, the message "BLOCK PARTLY MARKED" temporarily replaces the mode reminders on the right end of the status line, and when both ends of a block are marked, the message "BLOCK MARKED" appears in the same area.

It is not possible to perform normal editing (i.e. inserting or deleting) when block markers are set, and any attempt to do so will result in an error message. This is because the markers are kept as offsets into the text buffer, and these offsets would be altered by inserting or deleting characters in the buffer. Normal operation can be restored by carrying through the intended block operation, if it is a type which results in the "unmarking" of the block, or by explicitly issuing an "Unmark" command.

It is possible to place the "top" and "bottom" markers in the wrong sequence in the text buffer ("bottom" above "top"). If this is done, the "BLOCK PARTLY MARKED" message will appear, as if only one of the markers was set, and any attempt to execute a block operation will result in an error message. The condition can be corrected by re-setting either one of the markers. The ^QT and ^QB commands can be used to verify the location of block markers.

Although the "Insert from file" command is included in the ^B menu because of its similarity to other block operations, it does not require that a block be marked to operate.

^BT mark Top

"Marks" the current cursor location as the beginning of a block. The status line will show either "BLOCK MARKED" or "BLOCK PARTLY MARKED" depending on whether the other end of the block has been correctly marked. The editor will not insert or delete text while block markers are active.

^BB mark Bottom

"Marks" the current cursor location as the end of a block, as described above.

^BU Unmark block

Removes the current block markers and restores normal operation of insert and delete functions.

^BC Copy block

A copy of the text block currently marked is inserted in the text buffer at the current cursor location, leaving the original text block and its markers intact. This command is used to replicate a block of text at other locations in the buffer. The markers are left intact so that the text may be copied to additional locations if desired. If the block markers are no longer needed, they must be explicitly Unmarked.

^BM Move block

The text block currently marked is deleted from its original location and inserted at the current cursor position. The block markers are discarded.

^BK Kill block

The text block currently marked is deleted from the text buffer, and the markers are cancelled.

^BW Write block

Prompts for a file name, attempts to create a to file new output file with that name, and copies the text in the currently marked block to the output file. The file is then closed. The original text is left unaltered, and remains marked.

^BI Insert from file

Prompts for a file name, attempts to open the file for input, and reads the contents of the file into the text buffer, inserting at the current cursor location. If the contents of the file will not fit in the space available, an error message is printed and the operation aborted.

2.4.3 ^K Formatting and Tab Commands

The ^K menu contains commands relating to the automatic formatting of text and the setting and clearing of tabs.

^KC Center line

Centers the printable characters of the line containing the cursor between the left and right margins, leaving the cursor at the end of the centered string. Note that the presence of "print control" characters in a line will effect the centering of the line: if you wish to place control characters in a centered line, center it first and <u>then</u> insert the control characters.

^KW Word Wrap on/off

Turns word wrap ON if it is OFF, or OFF if it is ON. If word wrap is ON, the word "WRAP" appears on the status line. When word wrap is active, an attempt to type past the current right margin will result in an automatic movement to a new line. If the word being typed does not fit on the original line, the entire word is moved down to the next line to ensure that the original line will fit within the margins. If the wrap margin has been set (see below) to the right of the left margin, the new line is automatically indented to the wrap margin, otherwise the new line starts at the left margin. Word Wrap permits the typing of entire paragraphs without the use of the ENTER key, while ensuring that the text will fit within the currently set margins. Turning Wrap OFF also turns Justify OFF (see below).

^KJ Justify on/off

Turns Justify mode ON if it is OFF, and OFF if it is ON. Turning Justify ON also turns Word Wrap ON, and shows the words "JUSTIFY" and "WRAP" on the status line. When Justify mode is active, automatic word wrap operations will also pad lines with extra spaces between the words so that the lines will always end exactly at the current right margin, producing a "clean" rather than a "ragged" right edge. In performing the expansion of lines, spaces are never inserted to the left of the wrap margin, so that the edge under the wrap margin will also be uniform. This feature was used to produce the paragraph you are now reading. The presence of "print control" characters in a justified line will make the line appear too long on the screen, but the line will print correctly, since DynaStar correctly compensates for their presence.

^KD inDent on/off

Turns auto-indent mode ON or OFF. Auto-indent mode is designed as a convenience for programmers writing code in structured languages such as Pascal or "C". When indent mode is ON, typing a carriage return in INSERT mode will automatically indent the next line to the position of the first non-blank character on the line above, preserving the indentation of structured code. At this point, the indentation can be increased using the space or tab keys, decreased using the "<-" key, or maintained by simply typing away. Turning indent mode ON turns WRAP and JUSTIFY OFF.

^KA Adjust paragraph

Readjusts the text from the cursor location the end of the next paragraph to conform the currently set margins. This proc involves the stripping of existing carri returns and excess blanks which may have b inserted by a previous word wrap or mar justification operation. When stripp blanks, an attempt is made to leave at le one blank between words and two blanks af periods, question marks, and exclamat marks. A word wrap operation is t performed to make the text conform to current margins, including automa indentation to the wrap margin. If JUST mode is ON, the text is also justified produce a clean edge at the right margin. there is text to the left of the wrap mar on the first line of the section be readjusted, it is not altered. However, text on lines below the first is reformed the operation, regardless of its position o line. The operation is stopped at the end the current paragraph, as defined for the command, and the cursor is left on the fi character of the first non-empty line p this point. This makes it very convenient repeat the Adjust operation on the n paragraph in sequence.

^K -> or ^KI set tab

Sets a tab in the column corresponding to current cursor position. This is visua indicated by placing a triangular symbol the ruler line at the position above cursor. If the wrap margin is at t location, it is converted to a tab.

^K- or ^KM reMove tab

Clears the tab directly above the cursor the ruler line. This command can also be u to remove the wrap margin.

^KV set wrap margin

Sets the wrap margin at the cursor column displaying the letter "W" in the ruler l directly above the cursor. Word w operations automatically indent new lines this margin. Tab operations consider the w margin to be equivalent to a standard tab.

^KR set right

Prompts for a column number, waits for the margin user to enter a number (between 1 and 250), and then sets the right margin to this column, truncating or extending the ruler line as required. If the user does not enter a number, by pressing only ENTER, the current cursor position is used to define the new right margin. If the ruler line is extended by this operation, standard tabs are generated every five columns (at columns 1, 6, 11, 16, etc.). The right margin setting is used only by the word wrap and margin justification operations. Please note: If text is generated beyond a right margin of 250 while in the insert mode, the screen will turn black and there is a possibility of crashing the system. If this should accidently happen return the cursor to the left margin (^QS) immediately, there is a chance that your text will not be altered in any way.

^KK Kill all tabs

Removes ALL tabs fom the ruler line. If this is done accidentally, the fastest way to regenerate standard tabs is to move the right margin to column one, and then extend the ruler line by moving the margin back to its original column.

2.4.4 ^P Print Control Commands

The ^P prefix is used to insert print control characters in text, and to bring up a series of menus summarizing the "dot" commands for the DynaForm print formatter.

Typing a letter after the ^P prefix results in the insertion of a special character pair at the cursor location. The character pair consists of a special triangular symbol (the same on used on the ruler line to show tabs), followed by the capital form of the letter typed. Since we can not show the triangular symbol correctly in this manual, we will represent it by a "^". The DynaForm print formatter takes special action when this sequence is found in a file. The pairs "^D", "^B", and "^U" have special meaning described below; any other pair results in the sending of a standard control character to your printer in the place of the pair. This feature can be used to activate special features of certain printers such as the Epson.

Dot Commands

As with the other prefix menus, typing a space after a ^P will abort the supplementary command and return you to the main editing menu. In addition to this, however, typing a period (".") after a ^P will display special menus containing a summary of the dot commands for the DynaForm print formatter. There are approximately 5 of these special menus, and you can see them in turn by typing additional periods, or return to the main editing menu by typing any other character.

The following ^P sequences are interpreted by DynaForm to provide special effects at print time:

^PB Boldface on/off

Turns <u>Boldface</u> ON or OFF. Any text between a pair of these sequences will be printed in boldface. The method of producing the boldface effect on a printer will vary, but in general it results in multistriking, sometimes with a small shift between passes to produce a heavy effect, depending on the printer used.

^PD Double-strike on/off

Turns <u>Double-strike</u> ON or OFF. Any text between a pair of these sequences will be printed twice (with no shift), generally resulting in crisper type without the bold effect caused by ^PB.

^PU Underline on/off

Turns <u>Underline</u> ON or OFF. Text between a pair of these sequences will be underlined. Only non-blank characters will be underlined.

The effect of any of the three above commands can span more than one line, remaining in effect until another sequence turns the feature off again. The effects can also be combined: you can underline sections of boldface text, for instance, although only the <u>text</u> will be boldface in this case, but not the underlining.

You will notice if editing with JUSTIFY ON that print control characters will cause extension of lines containing them, appearing to disrupt the justified right margin. These lines will print correctly, however, because the control characters have "zero width" on a printer. DynaStar automatically adjusts its formatting to compensate for the extra width used up by these characters when displayed on the screen.

3/KEYBOARD MACROS

3.1.0 What is a Macro?

DynaStar contains a facility which permits the user to re- define any control character to mean virtually any other character or sequence of characters. This permits the complete remapping of the control functions on the main editing menu, and it permits the building of special complex commands such as global search and replace. At a simpler level, macros can be used to perform "boiler plate" operations such as inserting a copyright notice with a single keystroke.

In general, a macro is simply an expansion of the effect of a control key, sending a predefined string of characters to the editor in the place of the keystroke invoking the macro. If you define the ^L key to be a macro containing the character string "abc", then every time you subsequently type a ^L the editor will act as if you had typed "abc".

3.2.0 Defining a Macro

Macros are defined by entering "macro learn" mode, which can be done from both the "files" menu and the main editing menu by typing ESC (remember that ESC is generated by pressing CLEAR and BREAK at the same time). At this time the message "Macro learn" will appear on the status line. The next character you type should be the control character you wish to define as a macro (or else another ESC to cancel macro learn mode if you got there by accident). If you type a control character, it will be echoed on the status line in the usual form used on menus: as a capital letter preceded by a caret (^). At this point you can now type in the characters which are to be the body of the macro. You can type virtually any characters into a macro, including other control characters, which will again be displayed in the standard "caret" form. The only character which cannot be part of a macro is the ESC character. Typing an ESC terminates a macro definition, at which time it will disappear from the status line. The number of characters in a macro definition is arbitrary, but the total number of characters in all of your macros combined cannot exceed 400.

3.3.0 General Information

Special Characters

Several control characters have deliberately been left unassigned in DynaStar to make keys available for use as macros: these are ^L, ^O, and ^U. There are also four non-alphabetic keys which can generate valid control characters suitable for macro use: ^\, ^], ^^, and ^_. The character ^[is not useable because it is equivalent to ESC, and ^M should not be used because it is really the same as the ENTER key.

There are two control characters which have special meaning within macros. The first is ^U. Whenever ^U is encountered during execution of a macro, the editor accepts *one character* from the keyboard in the place of the ^U. The second special character is ^\. When a macro execution encounters a ^\, the macro restarts at the first character in the macro definition. It is sort of like a GO TO statement, and in the same way as the infamous GO TO statement, it is *very dangerous* and must be used carefully. Note that on the Color Computer, ^\ is generated by pressing SHIFT and UP-ARROW together.

Display your Macros

You can view your macros from the "files" menu with the "L" command. This command asks you whether you want to see a list of your currently defined macros (answer "?" for this option), or you want to see the *contents* of a particular macro (answer the prompt with the key corresponding to the desired macro). You can also redefine any macro by simply defining it again; in this case you should bear in mind that the previous definition space is not actually reclaimed, and if you do this too often in a single session you will overflow the macro definition area.

Examples

Macros are best explained with the aid of an example. Suppose you wanted to "move a command up" from one of the two- keystroke menus to the main menu: A good example would be the "find/replace Again" command (^QA), since there are many occasions when it would be convenient to invoke this command with a single keystroke. The following keystroke sequence would assign the ^L key to this function:

```
ESC ^L ^Q A ESC
```

After defining this macro, you can execute the ^QA command at any time from the main menu by typing ^L. Be careful, however, about typing ^L from other menus, because you might cause strange results. Macros are dangerous and should be treated with respect.

As a simpler example, let's just redefine one control key to be equivalent to another. We'll do something silly: swap the meaning of the ^S and ^D keys so they move the cursor in the opposite direction. The macro definition sequences required are:

```
ESC ^S ^D ESC
ESC ^D ^S ESC
```

The result of the above definitions will probably be total confusion, since these cursor movement commands will now work in an unnatural way, but the example serves to illustrate one more important point: macros cannot invoke macros. If a control character is contained in a macro, and that control character is itself a macro, this fact is ignored and the effect of the control character is defined by its original function as defined in this manual. If you think about it, this is the only way it would be possible to actually redefine the entire set of control commands.

Some further examples we will describe are two forms of a global search and replace macro. The first is a global replace command of the type that searches the entire edit buffer for a particular match string, and replaces each instance of the match string with something else, but subject to confirmation from the operator at each occurrence. The macro definition is as follows:

```
ESC ^U ^Q A ^U ^\ ESC
```

To use this macro you first execute a standard ^QL replace command so that the ^QA ("Again") command will repeat it. Then, instead of using ^QA to repeat the replace, type ^U. This will cause execution of a ^QA command, prompt for confirmation, accept the user's response in place of the ^U imbedded in the macro, and then loop back and do it again automatically. This brings up another question: how do you stop a looping macro? The answer is that a macro aborts whenever an error message is generated, which will happen in the above example as soon as the search reaches the end of the edit buffer and causes a "NOT FOUND" message. The other way to terminate a macro is to type ESC at a point in the macro where a ^U causes the editor to read from the keyboard. In the above example, you would answer the "replace? (y/n)" prompt with "Y" to do the replace, "N" to reject the replace, or "ESC" to abort the macro.

Macros also terminate when a cursor move out of the edit buffer is attempted.

The last example is a global replace <u>without</u> operator confirmation. The macro definition is:

```
ESC ^O ^Q A Y ^\ ESC
```

This macro is executed the same way as the previous example: by first using a standard repLace command (^QL), and then repeating it automatically by typing ^O. In this case the macro executes the ^QA ("Again") command as before, but <u>always</u> gives a "Y" response to the confirmation prompt, and then repeats the process until the end of the buffer is reached. There is <u>no way</u> to terminate this macro from the 'keyboard: it must run its course.

Words of Caution

Once a macro has been assigned to a control character, typing that control character will <u>always</u> cause execution of the macro <u>regardless</u>

of the context. This will sometimes require special care. As an example, suppose the control character ^Y has been redefined as a macro: you would then have to be careful not to answer a "y/n?" prompt with the CTRL key pressed, because inadvertent typing of a ^Y would trigger the macro instead of giving an affirmative response to the prompt.

As a final note, you will probably find macros tedious to use if "help" is enabled, because a lot of time will be wasted by the editor displaying menus for any prefix commands contained in your macros. The short answer to this complaint is that you should not attempt to use macros until you are thoroughly familiar with the operation of DynaStar, and by that time you will probably be operating with help OFF anyway.

editor so you can work on another file. If the current file is a new one, it simply writes out the contents of the text buffer to the SCRATCH file, closes it, and renames it to the name you specified on the last "N" command. If you were editing an existing file, it writes out the contents of the buffer to "SCRATCHnn", copies any unprocessed part of the original file to "SCRATCHnn", deletes your old file, and renames "SCRATCHnn" as above. It then clears the text buffer and waits for a new command.

If an "output file" error arises during operation of this or the "More" command, the SCRATCH file will be in an indeterminate state, and it would be wise to abandon it and return to the original file. The error may have resulted from a disk overflow, and it might be possible to remedy the situation by deleting unneeded files using the Shell command. However, it will not be possible to repeat the "W" command, and the only way of recovering the text in the text buffer is to use the write command from the ^B block menu.

X write and eXit

This command performs the same function as the Write command, except that when it is finished it exits to the operating system instead of remaining in the editor. It can also be used when no file is active to terminate the editor and exit to the operating system.

A Abandon file

This command is used when you have really fouled up your edit and want to go back to where you were before you started. It first prompts you to make sure you are really sure you want to throw away all the work you have just done. If the answer is "Y" (yes) it deletes the file "SCRATCHnn" and clears the edit buffer. If you were editing an old file you will still have it the way it was before you started, and'if you were creating a new file it will be gone.

E re-Enter

This command is used to re-enter the main editing menu to continue an editing session if you have temporarily entered the "files" menu to do some utility operation such as change the "help" state or examine or clean up some files. It can also be used to enter the edit mode when no file has been established to experiment. In this situation, if you decide you want to save your text, you can use the write command from the ^B block menu.

D Directory

Displays the names of the files in your current working directory.

H Help on/off

Turns HELP mode on or off. When HELP is ON, menus are displayed at the top of the screen to tell you what commands are available to you at all times. When HELP is OFF, all menu displays are suppressed except the "files" menu, so that you will have most of your screen area available for editing. H is a toggle command that alternately turns HELP ON and OFF if you execute it repeatedly. You get no indication of the state of the help flag except the presence or absence of menus. HELP defaults to ON, unless you activate DynaStar the "fast" way by passing it a filename on the command line (see section 6.0).

Shell command

This powerful (and dangerous) command prompts you for a one-line shell command and then executes it, returning to the editor when it is finished. Any edit currently in progress is undisturbed. It is possible to execute the shell command as a concurrent process by ending the command line with an ampersand ("&"). This is especially useful if you want to make a listing of a file you have just finished editing while you continue editing another. It is also possible to temporarily exit from DynaStar and enter an interactive session with shell by answering the prompt with an empty command line (typically a <return>). This will result in the standard system prompt and you can then use any commands you like provided there is enough free memory available to execute them. When you want to return to your edit session, answer an "OS9:" prompt with an end-of-file character (normally ESC).

When executing shell commands from the editor, there are several things you should be extremely careful about. It is very important that you do not tamper with the SCRATCH file if one is active, and that you do not alter any file which you are in the middle of editing. It would also be very dangerous to remove or swap the disk containing your working directory, since there are probably files open on it, and indescribable damage could result to these files or to other files if you re-entered the editor with the wrong disk mounted.

C Change directory

This command prompts you for a new working name, and then re-assigns it as your current directory for the duration of your edit session. You should not do this in the middle of a file edit, because the editor would lose track of its files. Note that it is <u>not</u> possible to use the Shell command to change the directory.

M More

This command is used when you have run out of space in the edit buffer or want to move on to a new section of the file you are working on which has not yet been read into the buffer. It writes out all of the text buffer up to but not including the line the cursor is on and then reads in as much more of your old file as is possible, setting aside approximately 1/6 of the buffer to leave room to insert text from the keyboard. This makes it possible to edit very large files in sections. Note that with both this command and the New command it is possible that the last line in the edit buffer might not have been completely read in, and it is wise in this situation to refrain from editing the incomplete line until it has been completely read in by the next More command.

ESC macro learn

Typing ESC from the "files" menu enters "macro learn" mode. This is described fully in Chapter 3: <u>Keyboard Macros</u>.

L List macros

The "L" command is used to display a list of currently defined keyboard macros, or to list the contents of a particular macro. When L is typed, a prompt is displayed on the status line. To display a particular macro, answer by typing the appropriate control character. To display a list of active macro definitions, type "?".

5.0.0 SYSTEM REQUIREMENTS and INSTALLATION

System Requirements

DynaStar 3.0 requires a 64k Radio Shack Color Computer running the OS-9 operating system, with one of the high-resolution graphics text screens from FHL O-PAK. It will not run correctly without an O-PAK hires screen, and the screen must be active before running DynaStar. Special versions of the standard hires screens "StdCs" and "bw6419" are supplied with DynaStar. These should be used to replace the corresponding screens as supplied with O-PAK. The new versions are identical to the old except that they contain two additional special characters: the triangular symbol used to mark tabs and print control sequences, and the solid block used as a second cursor on the ruler line. DynaStar will also work after a fashion with a 32 x 16 hires screen (some of the prompts will overflow onto the ruler line with a 32 character screen).

Installing DynaStar

The DynaStar distribution disk contains at least two program modules. There is a module "Ds" which is the actual editor, and a module "pinterp" which is a copy of the Dynasoft Pascal p-interpreter (the editor itself is in p-code). If the DynaForm print formatter has been purchased with the package, there will also be a module called "Df", which is also in p-code. Each of these files should be copied to your standard execution directory (usually the cmds directory of your system disk). If you already have a copy of Dynasoft Pascal on your system, you should replace your old "PINTERP" module with the one from the DynaStar disk. The editor automatically loads the interpreter when it is executed.

Additional Information

DynaStar uses both the "dir" and "rename" commands in the course of processing commands from the "files" menu. For this reason, if you should make up a special system disk for running DynaStar, the CMDS directory must have at least these two commands in it, in addition to Ds and pinterp.

Although DynaStar in principle can be run by more than one user at a time, in practice on a Color Computer there is not enough memory available to do this. This version of DynaStar is written specifically to work with the FHL O-PAK hires screen, and it will therefore not operate correctly with an external terminal. If you wish to use DynaStar with a video terminal on the RS232 port, you must obtain the standard OS-9 version. Although, the standard OS-9 version of DynaStar will operate on the color computer, Standard OS-9 and Radio Shack OS-9 text files are not compatible.

More Memory

DynaStar is called by typing a command of the form:

```
Ds [ filename ] [ #memsize ]
```

Both parameters on this command line are optional. The first is the name of a file to be edited. If this parameter is provided DynaStar will automatically load the file named and go directly into the editing mode, skipping the "files" menu. The second parameter is the usual OS-9 memory size parameter. If it is omitted, DynaStar allocates a memory area of 8K bytes, which provides about 6K of space for the text buffer, adequate for small edits only. DynaStar can use up to 32K bytes of work space, but on a 64k Color Computer with a hires screen, 15k is the practical limit. A typical command line would probably be:

```
Ds #15k
```

If too much memory is requested, it is possible that the editor would work just fine until an attempt is made to save an edit file, at which time it might fail because there is not enough memory to rename the SCRATCH file. If this happens, it is possible to recover by leaving the editor and executing the rename operation from the keyboard. Ask for a little less memory the next time.

Help on/off

If DynaStar is run with a filename provided on the command line, it starts up in "fast" mode (help turned off, skips the "files" menu). If DynaStar is run without naming the edit file on the command line, it starts up in "menu" mode (help turned on, giving constant menus, stopping first on the "files" menu). Regardless of the mode in which it starts up, help mode can be turned on and off at will from the "files" menu (see Chapter 4).

File Size

When editing large files, it is wise to be conscious of the amount of free space left on your working disk. There should be at least as much space as the size of the file you want to edit, since DynaStar creates a SCRATCH file of the same size during the course of an edit. This implies that it is not possible to edit a file larger than 1/2 of the capacity of a disk. When preparing a large document to be formatted with DynaForm, you will find it helpful to organize it as a series of smaller sub-files, and tie them together at print time using the DynaForm .FI (file insert) command.

APPENDIX I: SUMMARY OF DYNAFORM DOT COMMANDS

.BP n	Begin Page #n
.CP n	Conditional Page
.PN n	set Page Number
.PL n	set Page Length [66]
.IG	Ignore to next 'dot'
..	Comment line
.HE text	Header
.FO text	Footer
.MT n	Margin at Top [3]
.HM n	Header Margin [2]
.MB n	Margin at Bottom [8]
.FM n	Footer Margin [2]
.PO n	Page Offset [8]
.SS	set Single Space
.MS n	Multiple space [2]
.SP n	blank Space n lines
.OP n	OverPrint next n lines
.FI pathlist	File Insert
.MA xx	start Macro xx
.ME	End Macro
.XX	do macro xx
.DM text	Display Message
.IFE yy	If Even page do .yy
.IFO yy	If Odd page do .yy
.DXt text	inDeX entry (t=tag)
.XAt n	print indeX Alpha for tag t
.XNt n	print indeX Numeric for tag t
.SV name,text	Set Variable <name> to 'text'
.PV name,message	Prompt for Variable <name>
.DF pathlist	open Data File for mail-merge
.RV name1,name2,...	Read Variables <name1>, <name2>, etc.

Notes:
Document auto-repeats when data file open, until eof.
[] is default value for n, # in header/footer = page#

APPENDIX II : SUMMARY OF DYNASTAR COMMANDS

iles" menu:

	page
Abandon edit buffer without changing file	29
Change working directory	31
Directory	30
reEnter editing mode	29
Enter/Leave macro learn mode	31
Help on/off	30
List current macros	32
More: get next file section	31
edit New file	28
edit Old file	28
execute Shell command	30
Write edit buffer & update file	28
eXit editor (Writes if file active)	29

n editing menu:

Hold "CLEAR" key, type letter	
Tab right (inserts spaces in INSERT mode)	10
Delete character left	11
Cursor left word	10
Prefix: Block commands	12
Down screenful	11
Cursor right	10
Cursor up	10
Cursor right word	10
Delete character under cursor	11
Delete character left (same as BKSP)	11
Tab right (same as TAB)	10
Insert cr after cursor	12
Prefix: Format and Tab commands	12
doNe: Return to "files" menu	12
Prefix: Print control	12
Prefix: supplementary commands	12
Up screenful	11
Cursor left	10
Delete word right	11
Delete word left	11
Scroll down	11
Cursor down	10
Delete line	11
Scroll up	11
Enter/terminate macro learn mode	12

prefix:

	page
Again: repeat last find/replace	15
Cursor to Bottom of block	14
Cursor to end of text buffer	13
Cursor to right end of line	13
Cursor to top of screen	13
Find string	14
Replace string	14
Overtype mode on/off	15
Cursor to next paragraph	14
Cursor to beginning of text buffer	13
~~Cursor~~ to left margin	13
Cursor to Top of block	14
Cursor to bottom of screen	13
Delete line right	15

refix:

Mark Bottom of block	16
Copy block to cursor	16
Insert file at cursor	17
Kill block	17
Move block to cursor	16
Mark Top of block	16
Unmark block	16
Write block to file	17

refix:

Clear tab stop	19
Set tab stop	19
Adjust (reform) paragraph to margins	18
Center current line	17
auto-inDent mode on/off	18
Set tab stop	19
Justify mode on/off	18
Kill all tabs	20
Move tab stop (same as ^K-)	19
Set Right margin	20
Set wrap margin at cursor	19
Word Wrap on/off	17

efix:

Display "dot" command summary	21
Boldface on/off	21
Double strike on/off	21
Underline on/off	21

INDEX