THE SIMPLE WORD

Issue 1 Modern Word Processing April - June 1989

An early word processor

INSIDE

From the Editor

By: Tracy Bybee

The Simple Word Newsletter is intended to promote word processing and related areas of interest pertaining to the CoCo 3. As a statement of policy, unless otherwise requested, we will print material submitted to the newsletter as received with the following exceptions. Material unsuitable due to inappropriate language, racial - ethnic - or religious slurs, unfounded negative comments or claims (try to focus on the positive) directed towards specific programs or people in the CoCo community, and content not relating to our chosen subject.

Putting together a quality newsletter is a tall order. When you bring the focus down to a single subject, it becomes inversely more difficult to keep the quality up issue after issue. As editor, I plan to keep as close to the readers as possible. I feel including a large part of the reader submitted material in each issue is the best way to achieve this goal. There are many areas in which you, the subscriber may choose to contribute. The front page line art (we have many format ideas if you have the ability but lack ideas) relating to word processing or closely related topics, a cartoon about W/P, word search or puzzle, guest column with a human interest twist towards W/P (teaching, learning, or application experiences) or printers etc, various hints - tips - tricks, or the tutorial column if you have above average expertise in a specific area, and other topics we haven't thought of yet. Think of this as your opportunity to be discovered (one of our subscribers works in movie studio) and become famous.

Our present direction is W/P and related topics and will remain the same unless popular demand dictates we add other topics. In our efforts to keep the second issue timely, people wishing to include submissions must write for a submissions guideline/worksheet as soon as possible. I look forward to hearing from you.

Publisher's Column

By: Louis Bybee

First I must say I've gained a new respect for anyone who successfully publishes anything. What an experience!

I apologize for what I hope will be the last ever, late issue of the Simple Word. What I envisioned as a routine startup regimen was anything but routine. The Simple Word Newsletter is intended to cover word processing (W/P) and related programs or subjects by any author, that are compatible with the CoCo 3.

In serving as this newsletter's publisher, my input is as an advisor to a group of youth consisting of Journalism students, and those with general computer interests, that wanted to publish a W/P newsletter that would serve as a central gathering point for people with similar ideas. Their plan is to involve others, giving a larger number the opportunity to learn and teach within the same vehicle. In keeping with this concept, we have tried to keep the cost as low as possible, just enough to cover production costs.

While these young people have invested a great deal of time designing the format and content of the Simple Word, nothing is cast in stone, so the input of readers will determine any changes in content, design, or direction. We are encouraging you to become involved, even if you feel your contribution is less than world class. You have to start at some point, and we are often our own worst critic. What you have just finished learning, no matter the level of difficulty, someone else may just be starting to learn.

There is no connection between The Simple Word and Simply Better Word Processor. I participated in the beta testing of Simply Better and enjoy many of its features. If I find a W/P I like better, I'll use it. Simply Better has agreed to subsidize a portion of the cost of The Simple Word Newsletter to the registered owners of its W/P. That is the reason for the two different price levels. The Author Dale Rickert, says anything that promotes W/P on the CoCo 3 is good for any and all CoCo 3 W/P authors.

If any readers would like to submit material to be considered for publication, please write the editor and ask for a submissions guideline/worksheet.

NEWS SHORTS

By: The Simple Word Staff

Simply Better Software announced that version 2.0 of Simply Better (a word processor) would be released in July of 89. Version 2.0 will have additional features and commands, many of which were suggested through customer feedback. Dale Rickert, the author, also informed us of two things he is working on for possible future inclusion in his word processor. The first didn't do much for me as I don't have a real need for that feature. To people with a specific parlance it will be a delight. The second really knocked my socks off. I agreed not to even hint of its nature, but I will follow its development until I can.

Word has it that one of the major software houses is evaluating the feasibility of releasing a version of their word processor for use on the CoCo 3. This inside source tells me the major hurdle at this point is concern over cost/return with the number of CoCo 3's marketed being uncertain.

I wonder what ever happened to Bob Rosen of Spectrum Projects, and Dan Nelson of VIP Software. So many of the older members of the CoCo community seem to have dropped from existence. I read the other day where the Computerware people now support the Macintosh. If you know where some of the other people or companies are, or what they are doing now, that used to support the CoCo, drop me a line and we will put it all together in an article.

If any of you hear news of interest concerning the CoCo, or word processing and related topics, send it along and if we can verify it we will print it.

Glossary – Part One

By: Louis Bybee

This glossary will be presented over two or three issues. It is intended to be a combination of word processing, printing, and desktop publishing terms to introduce areas associated with word processing that may be new to some.

Access Time: The time a computer takes to locate data in memory or mass storage device and transfer it.

Active Window: Space or area of computer screen active, or in use by software or user.

Alphaneumeric: Consisting of numbers and/or letters.

Application Program: A program written for a specific purpose such as a database, or word processing.

Archive Storage: Short or long term storage of data away from the computer.

ASCII: (American Standard Code for Informational Interchange) "askee." A code developed to standardize numerical values for characters handled by different devices.

Ascender: A portion of a letter that raises above the normal body of a group of letters as in "h."

Auxiliary Line: A text line that can be placed almost anywhere on the printed page.

(Continued on page 5)

Subscription Information

The Simple Word

Registered Simply Better Owners

United States	$ 6.50
Canada-Mexico	$ 7.50 US
Other Countries	
Surface	$10.50 US
Air	$15.50 US

Others

United States	$ 8.75
Canada-Mexico	$ 9.75 US
Other Countries	
Surface	$12.75 US
Air	$17.74 US

Above rates are for one year, four issues. Cash, check, or money order, in US funds only. Rates are subject to change. PO Box 20273, Portland, Oregon 97220.

PUBLISHER
Louis Bybee
EDITOR
Tracy Bybee
TECHNICAL EDITORS
Don Park
Bryan Bybee
ILLUSTRATOR
Tracy Simpson
PROOFREADER
Matt McNicholas

Guest Column

By: Bonnie Russell

Bonnie Russell is a very organized, energy filled housewife and mother, who uses the CoCo 3 as one tool to Home School her children.

Three or four years ago a friend of ours became interested in computers. When Pac-Man became popular, I thought it was a new movie that I had not seen. When I heard a 6-year-old talk about "popping a wheelie," I thought I was in another world — or at least experiencing a generation gap. This friend of ours — I'll call Dale — started talking about computers and using language like "hexadecimal, ASCII, bits, bytes, pixels" and many more phrases that intrigued me, but I had no understanding of what he was talking about. However, my curiosity got the better of me and I listened and tried to understand. I was more interested in making use of what he was talking about than trying to understand what he was talking about.

My first experience on a word processor was by rote. I would watch the buttons he pushed and memorize the sequences. When I was left alone at the keyboard, I would sit staring at the screen and panic, get frustrated and so angry, that I could not make it do what Dale made it do. I spent many hours on the phone asking for help — and getting help in the form of "READ THE MANUAL!" My attempts at reading the manual were frustrated by the language. It took so long to wade through the terminology and complex configurations that I would end up quitting. I put it away many times because I did not know what I was doing.

Dale came up with a "spreadsheet" program and my interest was revived. I caught on quicker, and made use of it right away. My "checkbook" turned into a beautiful work of computer art. One of my frustrations was lack of "memory" because I was trying to put too much information on one sheet. Dale said I could have used different techniques that would enable me to store much more information, and all I needed to do was "READ THE MANUAL!" I spent many hours on my "spreadsheet" and ignored the word processor.

My fear of word processors was fueled by the misconception that I could use a Typewriter faster. I didn't mind using correction fluid and tape. I was pretty good at corrections. Meanwhile, another friend of mine wrote a manuscript on the subject of "homeschooling" and she let me read it. I asked her where she had it printed. She told me she wrote it on her "computer." I said, "How did you do that?" She seemed to be knowledgeable about computers, and after discussing the many uses she employs hers for, my interest was renewed. I got out my "MANUAL" and began reading and trying to learn some of the commands. It was so strange to me I was almost afraid of it. After about two years I learned maybe half a dozen new commands and began to relax a little and have fun with my word processor.

It was at this time that Dale introduced the concept of the "DATA-BASE" program. Wow! What a super idea! My mind went wild thinking of all the things I could do with a data-base, word processor combination: recipe files, Christmas letters, family newsletters, party invitations, booklists for school, church bulletins, letters to Congressmen, class lessons, book orders, and many more. Learning the "DATA-BASE" was not easy, and again I heard "READ THE MANUAL!" I wanted to learn it NOW without READING THE MANUAL! I did succeed doing address labels one time. I haven't given up on this feature. Dale has something better now.

Last fall Dale got all excited about this new program and called us. He wanted us to try his new word processor and tell him what we thought. I knew for sure it was going to be another "READ THE MANUAL" trick. But the way he talked, it didn't sound as burdensome. He had finally designed a "tutorial" manual that is so easy, any "computer neophyte" can pick it up and run this word processor without help. Dale calls it "Simply Better," and compared to XXX XXXXXX (Editors note: name was removed) it is "SIMPLY BETTER."

With "Simply Better" I can boot my disk and type a letter faster than I can find my typewriter and take the cover off, the typewriter has since been sold. Yes I am now a full-time word processor user. No more correction fluid or correction tape. I can type rough drafts and correct or rearrange right on the monitor, and print the final copy in less time than I ever dreamed of. "Simply Better" has made my computer a fun experience as well as an educational tool for our "home school." My children love to come in and "push the buttons." When they see me using it in so many different ways, they can't help wanting to learn how to use it too.

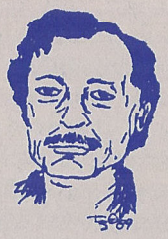
My sincere thanks go to Dale and what he has done for my computer literacy. "Simply Better" has opened a whole new world in processing information. I know there is a lot more to learn, but at least I'm not afraid to "READ THE MANUAL" now.

* * * * *

The Guest Column is open to anyone with a general interest in word processing or related subjects on the CoCo 3.

* * * * *

Author's Column



By: Dale Rickert

Space in this column is available to authors of word processing programs, or programs that relate to word processing such as spell checkers, print formatting, speech recognition, braille conversion, etc. that will run on a CoCo 3.

AUTHOR: Dale Rickert

PROGRAM: Simply Better

PROGRAM TYPE:

Word Processor

ADDRESS:

Simply Better Software
P.O. Box 20726
Portland, OR 97220

To those of you who have experienced the frustration of using a word processor that lacks the ability to do what you want, and to those of you who feel that most word processors are over-priced, and have upgrade costs that approach outrageous, this word processor was created for you.

When I began writing Simply Better (two years ago), my primary motive was to obtain a better word processor for my own use. I felt that if I couldn't buy one that would do what I wanted, I'd write one that could. In the beginning I thought I could write a top of the line word processor in less than six months. Little did I know that it would take over eighteen months of sixteen hour days, seven days a week. I could deal with the thousands of hours spent writing the program code, but the hundreds of hours spent depressed while looking for "bugs" in the software, I would like to reserve for tomorrow's day before yesterday with a little unused time left over. I found that writing a program like Simply Better consisted of alternating sessions of writing new code then trouble shooting it. I was spending a good share of my time being depressed and taking it out on those around me. (Ask my wife). Looking back, I'm not sure if I would have undertaken this project if my foresight were as clear as my hindsight. Hey, in sixty years it won't hurt a bit. The best pain relief available comes with the phone calls and letters telling me how much they like the program, and how it's the best word processor available for the CoCo 3. Having customers with good taste makes all the hours spent cooped up in a small room with a computer worthwhile.

> *"I felt that if I couldn't buy one that would do what I wanted, I'd write one that could."*

Many of you have asked me if I plan to write any other programs. At this time, I have no intentions of doing so. (Editors note: I wonder how many letters and phone calls it would take for Mr. Rickert to write a top quality database that would reach the performance limits of the CoCo 3? Show no mercy!) At most, I plan to add more features to Simply Better. I receive calls and letters daily from people suggesting how to change the program. One thing I'll always do is listen to suggestions that could lead to a better program. Version 1.1 contains changes that were suggested by Kevin Sloan, Jim DeStafeno, and others.

As to the future, I plan to continue marketing a top of the line word processor for the CoCo 3 at a price that anyone can afford. When new features are added to the program, they will be available to all registered owners at the lowest possible cost. I feel that too many software vendors treat their customers unfairly with unreasonable upgrade costs and other questionable practices that insult their pocketbooks and intelligence. Anyone agree?

Glossary . . .

(continued from page 3)

Background: To execute a task or process with a lower priority than the main operation. To print from the second window while entering text in the primary window.

Backing Up: Printing on back side of sheet.

Backup: To make a spare or working copy of a disk or software program.

Bad Break: Starting page or column with an orphan or hyphenated line.

Basis Weight: The weight in pounds of one ream (500 sheets) of the basic sheet size. Bond basic size is 17"x22".

Baud Rate: A means of measuring data transmission speed between computer and serial peripherals like printers, tape recorders, modems etc.

Bidirectional: Both directions. Often applied to a printer that prints in both directions of print head travel.

Binary: (Binary number system) This number system consists of two digits, 0 and 1. A machine language program is a Binary File, made up of 0's and 1's.

Bleed: Printed image extending off the paper edge.

Block: A section or specific area as in Text Block.

Boilerplate: See Formletter.

(Continued on page 7)

TUTORIAL COLUMN

By The Simple Word Staff

PROGRAM: Simply Better

SUBJECT: Tasks

The ability of a word processor to create, use, save, and load, tasks or macros, separates the wannabes from the top of the line programs. This task/macro ability allows more difficult or time consuming actions that would otherwise be out of the question. Simply Better isn't the only CoCo 3 word processor with task/macro capability. If there is enough interest in others, we will respond with articles on them.

A task could be as simple as a space bar (SB) or any combination of up to sixty keystrokes, in the eighty column mode. A task may also call another task or tasks from within a task. Sort of a nested task if you will. With ten keys available to assign tasks (0-9), and sixty keystrokes available with each task, you have 582 possible actions or characters with two keystrokes. If that weren't enough, you can also specify the number of repetitions of each task. Now the number of actions or characters possible from tasks is astronomical.

Any combination of keystrokes you can press at the keyboard, you can assign to a task such as loading, saving, and printing files, typing often used words or phrases, or complex text manipulation. The actions tasks can be used for are limited only by your imagination.

In this article I will cover a task with three variations, to create two column text from a standard text file or list, and a text file format to record actions, keystrokes, and setup information, necessary to keep track of multiple tasks.

I have a single column mailing list I wanted to convert into two columns to save paper. It's over seventy pages long in a single column format. I wanted page two on the listing to be column two on page one and so on down the list. My first attempt to accomplish this was to print all odd pages, then run paper through again with a wide left margin printing all even pages. It proved difficult aligning the printed lines, but did give me two columns. It satisfied me for about two seconds, then I knew only a true two columns would stop my whining and sniveling.

My mailing list consists of eleven elements, or lines per name or group. Printing at eight lines per inch gave me eighty-eight lines per page to work with. Wanting a small top and bottom margin, I settled on seven groups or seventy-seven printed lines per page. That meant the first line in the second column started with line seventy eight in the text file. I used two character combination groups to mark the first line of the first column (++), and the first line of the second column (;;), in the text file that normally would not occur in a text file.

You need to be in the insert mode to keep from overstriking characters, and set the tab at one cursor position greater than where you want the second column to start. When you tab across the screen no characters are set, and when you try to insert block moved text it would be placed immediately to the right of the last character on the line. A character or space placed after the tab forces spaces to be inserted back to the last character on the line, allowing the second column characters to be block moved to the correct position. Logically you should place the tab one position to the left to account for the extra space need above, but the two location characters at the beginning of the line (++) are deleted and moved down one line when all actions on the current line are completed. The two spaces to replace the location characters deleted, minus the space added because of the tab leaves one.

With the location character combinations properly placed, and set up taken care of (insert mode, tab set, width set wide enough), you are ready for the task keystrokes. To save space here see your manual for task entry. The following abbreviations will be used, SH for shift, SB for space bar, RA for right arrow, DA for down arrow, etc., CTRL for control, ENT for enter. Remember to press the ALT key before any arrow during task entry, except a shifted up arrow (SH-UA). The hyphen in the last example would mean to hold the shift key down while pressing the up arrow key. Pressing the ALT key before arrows avoids task entry being terminated before you are finished. Now the actual keystrokes (the spaces between the characters are for clarity only and are not to be entered):

CTRL F ;; DA F1 B DA DA
CTRL LA F1 B CTRL SH-UA
CTRL F ++ DA CTRL RA CTRL
D F2 SB CTRL M Y CTRL LA
CTRL D CTRL D DA DA ++

Now press the down arrow key without pressing the ALT key first to terminate task entry. In answer to the number query, answer 1. Always test newly entered tasks once or twice before assigning the final number of repetitions. In my case I needed seventy-seven repetitions, so after testing I increased the number asked after task entry to 77 (seventy-seven lines per column per page), and saved the task to disk with the file name REG2C2.TSK. REG for registration list, 2C for two column, 2 for task key to load task into, TSK to identify task files from text files.

Now I'll give a blow by blow description of the task keystrokes, remembering they could be manually entered from the keyboard. With the task entered and ready, setup taken care of, and cursor at the top of the text file, press CTRL 2 (or key number the task is assigned to) then Y to start the task. CTRL F ;; (find the first occurrence of two semicolons in the text file) F1 B (to begin block marking) DA DA CTRL LA (positions cursor to end block marking) F1 B (to end block marking) CTRL SH-UA (go to the beginning of the text file) CTRL F ++ DA (find the first occurrence of ++ in the text file) CTRL RA CTRL D (move to the end of current line and delete the carriage return) F2 SB (tab and insert space) CTRL M Y (insert marked block) CTRL LA CTRL

(Continued on next page)

Tutorial . . .

(Continued from page 6

D CTRL D (move to beginning of current text line and delete location characters) DA DA ++ (move down one text line and insert new location characters). One cycle is now complete and ready for the next. When all cycles are complete you'll end up with all four location characters on the same line.

Before I was able to gloat with this success, I had to ask myself why I was placing the location characters manually when such powerful tasks were available. The following task does that and calls the two column task. Use the same setup as the first task.

> ++ CTRL LA DA CTRL DA CTRL DA CTRL DA DA DA DA DA DA DA DA DA DA DA DA DA DA ;; CTRL SH-UA CTRL 2 CTRL LA CTRL D CTRL D CTRL D CTRL D

This task was saved as REG2C1.TSK. The task description is as follows: ++ (to place top of text location characters) CTRL LA DA CTRL DA CTRL DA CTRL DA DA DA DA DA DA DA DA DA DA DA DA DA DA DA (moves cursor to beginning of current line, and down to line that will be first line of second column) ;; (to place second set of location characters) CTRL SH-UA (move to top of text) CTRL 2 (calls task two) CTRL LA CTRL D CTRL D CTRL D CTRL D (deletes the end of cycle, left over location characters and leaves cursor ready for the next cycle) Before starting this task, the cursor must be placed where the two column action is to start.

The third task is one designed to produce two columns where every other line of text is to be moved to the second column. Set the tab one cursor position less than the desired start location of the second column. Be sure to set width wide enough. Place cursor where the action is to start before activating task. The task description is as follows: CTRL RA CTRL D (move to end of current line and delete CR) F2 SB DA (tab, insert space, and move next line down to second column position on current line) DA DA (move cursor one line down ready for next cycle.)

There are a few things to remember to make using these tasks result in minimum heartburn. Unless working with short text lines, you must shorten the text lines to one half the printed line width minus one half the center margin. For example if your original text line was seventy characters wide with margins set at five and seventy five, and your new center margin is to be four characters wide, you must change the original text width to thirty three using the width command. You would then place a CR at the end of each line with a short task. In the overstrike mode, so you don't end up with two carriage returns on the same line, key in the following task. CTRL RA ENT DA (move right to end of current line and insert CR then move down one line). Place cursor at start text needing carriage returns before using task. After task is completed set width wide enough to allow for two columns. If the text you wish to process into two columns has an odd number of lines, place the odd line in the left column for best appearance.

If you wish the ALT key to be part of the task and not just for encoding arrow keys, you must press the ALT key twice.

If your printer fails to recognize who is supposed to be in control, load a task file as a text file and send it to the printer. It will send it over the edge for sure!

The tasks presented here are not the only way to produce two column text. I am issuing a formal challenge for you to create and document different tasks to create two and more columns, and other tasks that do creative or unusual things. I will pass on the more useful or interesting submissions.

After creating more than one task and saving them to disk, you may later experience the same brain death I did trying to remember what they did, set up required, and the keystrokes used. To avoid this I save all tasks with the file extension.TSK, and created a text file called TASKS.TXT to store task related data. A sample of that file is on the back page. Tasks forever!

* * * * *

"The written word is more durable, and usually causes less trouble than word of mouth."

* * * * *

Glossary . . .

(Continued from page 5

Bottom Margin: The unprinted area between the last printed line and the page bottom.

Brailling Device: Electronic: A device that allows the sight impaired to determine the status of a CRT screen.

Buffer: A specific area of memory dedicated for data or text storage.

Byte: A grouping of bits or binary digits (usually seven or eight) making up one character.

Camera-ready: Material delivered to printer ready for plate making.

Carriage Return: (CR) End of text line termination. (Enter key)

Case: A definition of character type, uppercase (Capital) or lowercase.

Case: Covers of a hard bound book.

Casting-off: To find the space text copy will occupy when typeset.

Centering: Positioning a text line equidistant from left and right edge.

Character: A single unit of text. (letter, number, space, or punctuation).

Character String: A grouping of one or more characters.

Checker, Spelling-Punctuation: A computer program designed to check and/or correct a text file for possible errors.

Cicero: A European unit of measure equivalent to 4.55 millimeters, used for measuring type size.

Collate: To place sheets in proper order.

(Continued on next page)

Glossary . . .

(Continued from page 7

Column Guide: Vertical guides marking the left and right column edges.

Column Width: The horizontal measure of a column.

Command Mode: The mode of a computer program where specific instructions are executed (load, save, kill etc).

Command-driven: A computer system, or system software that responds to keystroke commands as opposed to menu choices.

Configure: The act of setting or changing default parameters for printers, screen size, screen colors etc.

Control code: A specific number combination sent to screen or printer to control specific or second copies of software.

CPI: An abbreviation for characters per horizontal inch.

Crop: To size artwork or photographs to size needed.

Crossmarks: Register marks to align multicolor printing.

Curl: Moisture or coating differences can cause paper to curl.

Cursor: Character or block indicating current position on screen where keypress will insert character.

Daisy-chain: To link or hook together sequentially.

Decimal: A base ten number system. May be used to express numerical value of an ASCII character, (32-127) control codes, (1-31 sometimes greater) and graphic codes, (128-255).

Deckle Edge: Uneven, untrimmed edge of paper.

Dedicated Word Processor: A computer designed for word processing that usually can not run other programs. They often have the WP software in a chip inside the computer.

Default Settings: Startup numerical values embedded in a computer program controlling program, screen, and printer actions.

Delimiter: A character which flags or alerts program to act upon special character sequences such as commas, periods, or slashes between characters.

Descender: A portion of a letter that drops below the normal body of a group of letters as in "y".

Desktop Publishing: A term applied to the production of printed material on a desktop computer. The most common being newsletters, memos, ad copy, banners, etc.

Directory: A list of files or programs in memory or on storage media.

Document: A form, memo, report, invoice, etc. that is designed to convey or receive information.

Double Strike: A printer action, printing character twice in the same position.

Double Wide: A printer action, printing character twice normal width. 10 CPI would be printed 5 CPI.

Driver: See Printer Driver.

Dummy: Blank or incomplete pages placed to simulate layout.

Electrophotography: Copy machines use this process.

Elite: Smallest standard typewriter type size. 12 characters per inch.

Elongated: See Double Wide.

Em: The square of a type body size.

En: One half the width of an Em.

Ergonomics: Workspace and devices planned with efficient and correct human body use in mind.

Escape: A printer control code alerting computer that following number combinations are printer control codes.

Facing Pages: Two pages that face each other when the publication is opened.

Facsimile, FAX: A telecommunications system that allows images to be sent over the phone lines. Kind of a long distance copier.

Flush Right: Forcing text to align with right margin.

Folio: 1. Page number. 2. Sheet of paper folded once.

Footer: A text line that may be enabled to repeat on selected page bottoms.

Formatting: Arranging screen or printed text to meet desired effect.

Format: 1. To prepare a diskette for data. 2. The general size, shape, margins, and appearance, etc of a publication.

Formfeed: A control code that forces printer to slew or advance to top of next form or page.

Form Letter: An incomplete letter body designed to be sent to a large number of people with the specific name, address, etc. information stored in a separate file to be merged later.

Free Sheet: Paper sheet without groundwood pulp.

Generic Characteristics: Elements that are common among different products. In word processors, control D = character delete, control V = view mode etc.

Global: The ability to act on all occurrences of specified string, such as Global Search, or Global Search and Replace.

Global Backup: To backup all files on a hard disk.

Grain: The direction most fibers form and/or lie in paper.

Greeking: A process where text and/or art is represented by an approximation of dots so an entire layout or page can be viewed at once.

Gutter: The unprinted area between the printing and the binding.

Hairline: Thinnest possible line. Usually .25 point.

Halftone: A photograph converted to dots of various density to simulate a grey scale. Printing can only reproduce light (no image) and dark (printed image).

Handshake: An interchange of data between computer and peripheral controlling data transfer.

To Be Continued . . .

HINTS & TIPS

By: Louis Bybee

PROGRAM: Simply Better

The source of items in this column unless so noted are from the staff. I would be pleased to include Hints & Tips from readers on word processing that directly or indirectly relate to the CoCo 3, or any word processor that will run on it.

Reading the manual of any software package seems like old advice but I have heard from two word processor authors that most of the questions they respond to could have been handled by the customer reading the manual before making the phone call.

As you begin using the task functions you will find an incredible amount of power available limited only by your imagination or how clever you are. Being able to save them to disk is a real time saver when you go to use the longer and more complicated ones again. See the tutorial column this issue on a suggested format to save tasks and an accompanying text file to document the task and required setup or sequence to use.

At times when five fonts seem limiting, remember you can enable font combinations you use together as one. Simply use the Config program and insert more than one printer code in the font area for on. Another method of fooling the program to enable more than one font at a time would be to set the off codes the same as the on codes for the fonts you wish to have active at the same time. Then the font would stay enabled in the printer even after you shut off the font in the text. Another interesting possibility would be to set a different code for underlining such as superscript or any other code if you could do without underlining. Remember you can have any number of modules configured different ways for special uses.

The following Hints & Tips which apply to the Simply Better W/P are from Kevin Sloan of Meridian, MS.

The Primary and Alternate windows are not only linked by the find/replace and the get block function, they also share tasks, the mask status, the verify status, tab stops, and print parameters.

If the mail merge process is aborted, the current form will be appended to the end of the document. Be sure to remove this form before saving or doing any further printing because Simply Better will treat it as an extension of the list.

A needed feature is the ability to press <BREAK> when a directory is longer than the current size of the window to allow one to go to the Command mode instead of the next page of the directory without clearing the window.

While pre-viewing text on screen, it would be much simpler if the author would allow the user to at least go back to the cursor position and resume pre-viewing with one keystroke of the <UP-ARROW> key instead of having to press <BREAK> and press <CTRL>-V again.

Before doing a global Find and Replace, one may want to locate a sequence before doing a Replace (that is press the <DOWN=ARROW> key at the Loc: prompt). That way, one can ensure that the correct text will be found.

An easy way to count the number of times an occurrence of a character or sequence of characters in one's document is to use the Find and Replace feature in Simply Better. Just enter character(s) to be found and enter the SAME character(s) to be replaced, and finally enter a number greater than the possible number of times it could have occurred.

Simply Better will respond with how many times it replaced which will be the number of occurrences in the text following the cursor. Be careful to enter the identical character sequence for the replace as for the find, or some quite drastic results may occur. Also note that if the mask is on, the case of the character(s) may be altered with this process.

While in the insert mode, anytime a change in font is made, the current line buffer is dumped to the document rendering the undo feature of the <BREAK> key lost. That is even if text is added to a line, any change in font after the insertion will place the the change and font into the document where it cannot be undone with the <BREAK> key. Deleting or overstriking the text will, therefore, have to be done.

If by some chance memory becomes filled and the error "out of memory" occurs, control key sequences and arrow key sequences will not work - seeming the program has "locked-up." A quick press of the <BREAK> key will clear the current line buffer and release the program to do saving/deleting/overstriking of the document.

Always save one's parameters (from the Config program) as a separate file when one intends to append it to the end of Simply Better. That way when future upgrades arrive or when changes need to be made on the default, the parameters will be on disk for printing or editing at the discretion of the user.

Do not change print parameters while a window is performing any kind of printing. Doing so will have an immediate effect on the printing and could result in unexpected results. The author needs to freeze the changing of print parameters while printing is being conducted.

Thank you Kevin. Any other people stumble over something interesting?

We welcome Hints & Tips from anyone, relating to any word processor that runs on the CoCo 3. They must be constructive, as we don't see anything useful in derogatory comments.

SIMPLY BETTER TASK DATA FILE

FILE NAME	KEY#	KEYSTROKES	SET UP	DESCRIPTION
REG2C1.TSK	1	++ CTRL LA DA CTRL DA CTRL DA CTRL DA DA DA DA DA DA DA DA DA DA DA DA DA ;; CTRL SH-UA CTRL2 CTRL LA CTRL D CTRL D CTRL DCTRL D	I MODE*SET TAB 1 MORE THAN WANTED	1 OF 2 FOR 2 COL LIST
REG2C2.TSK	2	CTRL F ;; DA F1B DA DA CTRL LA F1B CTRL SH-UA CTRL F ++ DA CTRL RA CTRL D F2 SB CTRL M Y CTRL LA CTRL D CTRL D DA DA ++	SEE ABOVE	2 OF 2 FOR 2 COL REG
REGNAMA.TSK		CTRL RA CTRL D DA CTRL RA CTRL D SB DA CTRL RA CTRL D SB DA CTRL LA F1B F2 F1B CTRL RA SB CTRL M Y CTRL LA DA DA CTRL X DA CTRL X DA CTRL X DA CTRL X DA CTRL X DA CTRL X DA CTRL X DA	I MODE SET TAB AT 7	LAST NAME, FIRST NAME TITLE SER# FROM REGLST
REGNAM1.TSK	1	++ CTRL LA DA CTRL DA CTRL DA DA DA DA DA DA DA DA DA DA DA ;; CTRL SH-UA CTRL 2 CTRL LA CTRL D CTRL D CTRL D CTRL D	SAME AS REG2C1	1 OF 2 FOR 2 COL NAME L,F SER#
REGNAM2.TSK	2	SAME KEYSTROKES AS REG2C2.TSK	SAME AS REG2C1	2 OF 2 FOR 2 COL NAME L,F SER#

THE SIMPLE WORD
P.O. Box 20273
Portland, OR 97220

ADDRESS CORRECTION REQUESTED