PRODUCING BRAILLE WITH A COMPUTER;
A GUIDE FOR PEOPLE WHO FEAR COMPUTERS

by

E. Desautels

Computer Sciences Technical Report #469

March 1982

CSD-TR-469
March 1982

Producing Braille with a Computer; A Guide for People Who Fear Computers.

E. Desautels
Computer Sciences Department
University of Wisconsin-Madison
1210 W. Dayton Street
Madison, WI 53706

ABSTRACT

The needs of blind students are discussed along with the resources available at this university: a Triformation LED-120 braille printing computer terminal and a Grade 2 braille translation program which runs on a UNIVAC 1100/80 computer. The following topics are discussed: preparing text, using prepared text, using your own micro, using your own word-processor, using MACC facilities, (1) at 1210 W. Dayton, (2) anywhere else-by telephone, the ADP connection. Generating Braille, what can/cannot easily handled, common misconceptions, alternatives to Braille on paper- (1) spoken books, (2) cassette braille, (3) reading machines. Other obstacles, using computer-based information systems. The report is written for the benefit of people who are not necessarily conversant with computer technology.

Table of Contents

Acknowledgements: John Boyer wrote the braille translation program described in this report. The Triformation LED-120 braille terminal was provided by a grant from the Wisconsin Department of Vocational Rehabilitation. The author was assisted by a grant provided by the UW System program for Undergraduate Teaching Improvement.

Introduction--Needs and Resources

A number of blind students are enrolled at the University of Wisconsin-Madison. They manage to "read" their textbooks, assignments, reports, tests, etc. in a variety of ways. Some of this material is read by paid or volunteer readers, either in person or by proxy, thanks to the use of cassette recording. A few can do some of their own "reading" with the help of a tactile input device known as the Optacon. Unfortunately the reading-rate with the Optacon (for those who learn how to master it) is quite low--comparable to listening to someone read one letter at a time. Many of these students depend upon obtaining their textbooks in Braille. Much of the preparation of Braille books is done through volunteer organizations. It is usually the case that recent textbooks are not available in Braille, and since the translation from inkprint to braille is essentially an intellectual task which is very labor-intensive, brailled books often arrive too late to be of value in the semester for which they were requested.

Whatever the media used to make the printed word accessible to our blind students, it is a case of too little, too late. What resources can we bring to bear on this problem? This report is not about to solve all the problems it identifies; it simply wants to convey information on what can or cannot be done with the tools we have here today.

We have on campus (in room B109 at the Computer Sciences building, 1210 W. Dayton Street) a braille terminal, the LED-120, manufactured by Triformation. Its keyboard is very much like that of an electric typewriter. You do not have to know braille to use this machine to produce braille. Knowing braille helps, but it is not necessary. It can be used to prepare braille documents simply by typing as you would on an electric typewriter--it then produces what is known as Grade I Braille (more about this later). If you were going to prepare a long document (e.g. a fifty-page report), your intended reader would much prefer that you use another form of braille, known as Grade II braille. This form uses much less paper, and it reads much more rapidly. Unfortunately it takes a highly skilled person to mentally translate from the printed word (also referred to as ink-print) to Grade II braille--thus the need for trained volunteers.

However, if the report you want to render in Grade II braille is typed into a computer in plain English, it is possible for the computer to do a credible job of translating from your inkprint equivalent (once it has been typed in) into Grade II braille.You can then direct the computer to use the same braille terminal. The bulk of this report concerns ways and means of doing this, and when you should

or should not attempt it. No previous exposure to computers is assumed. If your experience in using the information in this report contradicts the previous statement, the author would appreciate your feedback (I was going to say "your input"--but that is computer jargon).

Preparing text

Let us set the stage by focussing on a specific task: you have in front of you a final exam which has too many detailed questions and too much factual background material to make it easy to administer by taping it on an audio cassette. The problem then becomes: how can we produce a copy of it in braille? Having determined that time does not permit a volunteer braillist to do it, how can you do it yourself?

Step 1: type it into a computer.
Step 2: have the computer translate it into grade II braille.
Step 3: have the computer type it out on the braille terminal.

So you need a computer! You don't have to own one, just borrow one.

Using MACC facilities

The people at the Madison Academic Computing Center, familiarly known as MACC (also located at 1210 W. Dayton) would be happy to let you use the UNIVAC computer they operate for all of us. All you need is an account for use of the UNIVAC computer. You can set one up by going to MACC's billing office (room 2130). After an exchange of money (or requisitions) you will be assigned a 5-digit account number (also ask them about the 10-digit identifier you should use). You can then go to room B109 and sit at one of MACC's public terminals. They too look like electric typewriters. Turn on its ON/OFF switch, type the 4 letters UWTT, and after you get a nice greeting message, type the line
@RUN Freddy,12345,8888888888 and push the key labeled RETURN. You should of course substitute your own name, your own account number (for 12345), and your own 10-digit identification (in place of 8888888888). Now that you are logged-in you can call up the special computer program called NEAT. It will help you type in a document, review it, change it, and file it away.

See appendix B for a gentle introduction to using NEAT. It

is permit you to get your document in the computer. Now that you have your document in the computer, you are ready for Step 2.

Step 2: requesting the computer to process the document you have typed in, so that it is converted into its grade II representation. Step 3: conveying the translation prepared in step 2 so that it comes out on the braille terminal.

This is an explanation of how to use the braille translation facilities on the UNIVAC 1100/80 at MACC. After login in as discussed earlier, begin by typing the statements

@use b,brl*p @asg,,x b

The UNIVAC computer will accept its commands in lower case letters as well as in upper case letters. For the first file to be brailled type the statements

@add b.triformation

if you plan to use the Triformation braille terminal we described earlier.

If you are using a Sagem braille terminal, use the following statement instead:

@add b.sagem

After a few seconds your terminal will display the message:

University of Wisconsin Braille Translator

now type the statement

@add inputfile

where inputfile is the name of the file or element you wish to braille. Before pressing carriage return adjust the paper so it is just before the start of a new page. This will take practice. You will probably fail to get it just right the first few times, but keep trying. When you press carriage return, your terminal will start producing grade 2 output immediately. When it stops printing, type

@eof

Your terminal will print one or two more lines of text, then skip to the top of a new page and print performance data on the braille program. You can avoid the need to type in the @eof statement and the interruption of the text which it entails, by placing the translator control statement

.end

at the end of the files you wish to braille. The program will print almost anything in good grade 2 with a reasonable format. More control of formatting and translation can be achieved by using the translator control statements discussed in other documentation. When using the Triformation LED-120 braille terminal, the lines per page switch should be set to 25. This is necessary to prevent the perforations between pages from creeping out of the blank space left by the translator and eventually winding up in a printed line. When using the Sagem braille terminal, the forms switch should be set to off. If you are brailling more than one file in the same run, you can call the translator directly for the second and subsequent files. Type

@b.translator

after receiving the "University of Wisconsin Braille Translator" message, type the @add inputfile statement, and proceed as above.

Using prepared text

It may be the case that the document you need in braille has already been typed into the UNIVAC computer. If so, you should review it to ensure that no extraneous material is included with it. For instance, some documents have imbedded in their text special instructions for a phototypesetting machine. You don't want those instructions coming out as part of the braille translations. You can weed these out by reverting to NEAT once more. If you get stuck in dealing with the UNIVAC computer, you can get help from a consultant, by going to room 1154, during normal business hours. You can then submit this cleaned-up document for translation into grade II braille by going directly to Step 2 above.

Using your own micro

You may have access to a microcomputer, such as an Apple II, or a TRS-80. If this micro is also equipped with
(1) communications interface and telephone modem or coupler
(2) communications software then you could use it to prepare text, and transmit it to the UNIVAC computer for translation into grade II braille, by using a telephone link between your micro and the UNIVAC computer.

You in effect can use your micro as an "intelligent terminal." Set up your micro for communication at 300 baud (see

your micro's instructions if this is new to you). If you then dial (608) 263-1108, you will connect to one of the UNIVAC computer's many telephone lines. They operate almost 24 hours a day, so if you are distant, take advantage of night and weekend rates (the computer charges are also lower then too!).

Using your own word-processor

You may have access to a word processor, such as those manufactured by A.B. Dick, Lanier, CPT, IBM, Xerox, etc. If your word processor has been equipped with optional communications hardware and the corresponding software package, then more likely than not you could prepare documents with it, revise these documents as needed, then transmit them electronically, by dialing up the MACC UNIVAC computer as it was done in an earlier section.

What if your word processor does not have the communications hardware? Almost every word processor uses some kind of a floppy disk or diskette. If you prepared a document using a Brand X word processor, you can record that document on a diskette, and transfer the diskette (by hand carrying it) to someone elses Brand X word processor and use their communication capability. For instance, on the Madison campus, you could use a non-communicating CPT word processor to prepare a document, then you could prevail upon the College of Letters and Sciences staff to use their communicating CPT to transfer your document to the UNIVAC computer. The author would be pleased to help you find communicating counterparts to whatever Brand X word processor you have access to.

The ADP connection

The Madison campus is also served by another large computer made by IBM. It is managed by Administrative Data Processing. A network of Wang word processors is connected to their IBM computer. Their IBM computer in turn is connected to MACC's UNIVAC computer. So it is possible to prepare a document on any of these Wang machines, and have that document electronically relayed to UNIVAC via ADP's IBM computer. Step-by-step instructions on doing this are given in an appendix. Once the document appears on the UNIVAC computer (it takes just a few minutes for the transfer to occur) you can review it using NEAT, then process it as described earlier, to generate grade II braille.

Generating Braille

The basis for braille lies in its use of combinations of

raised dots. As many as six dots can be arranged in a configuration of three rows with two columns. There can be 63 such distinct dot combinations (the 64th has no raised dots). A few of the codes are shown here, using periods for raised dots, and dashes for non-raised dots:

a	b	x	y	z
.-	.-	..	..	.-
--	.-	--	-.	-.
--	--	..	..	..

The digits 1-9 and 0 share the same codes as do the first ten letters of the alphabet. This requires that special codes be used to indicate when letters or figures follow. We will call the 6-dot combinations braille cells.

Grade 1 braille provides in effect a one-to-one transliteration from the inkprint character to its braille cell equivalent. We get slightly more than one braille cell per inkprint character because of the need to insert the occasional letters/figures code. When one considers that the normal line width for braille text is 40 characters, one can appreciate why some other coding scheme is needed.

Grade 2 braille uses the same braille dot cells. However, instead of transliterating character by character, some character sequences may be represented by a single braille cell. Some whole words have single-cell contractions (e.g., and, it, people) while some other whole words have two-cell contractions (e.g. "friend" transliterates as "fr"). There are many contractions for word-parts (e.g. two-cell contraction for "ness" when it ends a word). There are many such rules and many exceptions to them. The upshot is

(1) in novels and other non-technical literature, Grade 2 braille averages about 4.2 braille-cells per word, as opposed to the 5.5 characters per word in the inkprint equivalent (both averages include the word-ending space). This is clearly an improvement over grade 1. Still it leaves much to be desired. An encyclopedia which normally fits in a few feet of shelf space can take forty feet of shelf space in its Grade 2 equivalent.

(2) the rules for translating inkprint to Grade 2 braille require a great deal of training and dedication for the volunteers who undertake to do it.

There are yet other braille coding schemes in use, notably for representing musical scores, and another for mathematics. The Grade 2 computer program whose use is described in this report was prepared by a person who is expert at both computing and braille. In case you missed the acknowledgements, credit for creating this computer program should go to John Boyer.

What can/cannot easily be handled

Ordinary running text as found in a novel is very easy to represent in braille. Unfortunately that is also the kind of material that is most easily used with an audio cassette recorder. When you start transcribing textbooks, you have a lot of nonstandard material to worry about: tables, footnotes, figures, photographs, greek characters, mathematical symbols, etc.

It is commonly assumed that it must be very easy to take the computer tape that a publisher used in typesetting a book, and use that tape to generate a braille version of the book. I have been involved with a project to do just that, and it is far from easy. We obtained the computer tape used to print Blacks Law Dictionary, thanks to the cooperation extended by West Publishing, in St. Paul. The publisher even provided us with the key to the special typesetting codes that are sprinkled throughout the text as it appears on the tape. First of all, we had to write a special program to interpret and/or extract these codes. Then we discovered that the codes were not necessarily used consistently or correctly. My conclusion was that it takes almost as much time and effort, and a lot of computer expertise and computer use, to exploit such a computer tape as it would for a layperson to type it all in.

I also now have first-hand experience in seeing a textbook through every stage of processing by a publisher. The 400 page manuscript I prepared as a computer science textbook is captured on a computer typesetting tape. However all of the figures, and many of the key tables are absent from that tape. Furthermore, beyond a certain point in time, the computer tape no longer reflects accurately what appears on the final printed page. There is far more cutting and pasting done in publishing than you might expect, all of the computer gear notwithstanding.

Alternatives to Braille on paper

Spoken books were mentioned earlier. The equipment to record and play back the readings of volunteer readers is readily available. It has several virtues, among them being low cost and portability. Many blind students tape the lectures they attend. The "reading speed" while listening to a tape can be increased by using a variable-speed player. Volunteer readers are not always easy to come by, and many readers prefer not to tape material from an area they are not familiar with (e.g., attempts to find volunteer readers in Madison to tape the textbook I have written were not successful, because none of the volunteers felt at ease with the subject matter). Consequently a national group was

contacted--and they have a large backlog of requests.

Cassette braille is a fairly recent development. The basic idea is simple. Instead of recording braille dots on paper, record them electronically on cassette tapes. Then use a modified cassette reader to play back the tapes. When you play back the tapes, you don't listen to them; you feel the braille codes with your fingertips, ds they are displayed on a row of movable pins, one line of text at a time. The cassette reader is equipped with both fast forward and fast reverse, and with the ability to search for a designated section number. It is even possible to key in and record braille. These readers can have an adapter so they can monitor and record computer codes, while displaying their braille equivalents one line at a time. One cassette tape can hold the equivalent of hundreds of braille pages. There are two manufacturers of cassette-braille equipment: Triformation and TeleSensory Systems Inc. They each sell for prices in the $4,500-6,000 range, depending upon the options you choose. Unfortunately tapes prepared for use on one of these machines cannot be used on the other machine, or vice versa. This will seriously detract from their use.

The Kurzweil Reading Machine is another interesting development. You place the pages of the document you wish the machine to read aloud in the machine, just as you would if you were using a copying machine (e.g., Xerox). The machine scans the page and tapes a synthetic-voice English readout of what it sees. The success of this machine is hard to gauge even though it has been commercially available for a few years. Kurzweil is now a subsidiary of the Xerox company, and Xerox recently initiated a program to donate 100 of these $30,000 machines to colleges.

It would be very useful to be able to tap into the Kurzweil machine to intercept the letter codes it is scanning. This would provide yet another way of going from inkprint to braille with fewer keystrokes. It would also provide a way to capture preprinted text so large-print versions of it could be prepared, once again using far fewer keystrokes. Why is it so important to reduce the number of keystrokes? Having to transcribe text raises the possibility of transcription errors. As a teacher, I am well aware of the difficulties sighted students have even when using essentially error-free textbooks, notes and reference material. Providing blind students with brailled material that require transcription almost guarantees that they will have to contend with errors their teachers don't know about. Transcription errors are normally caught by careful proofreading. If it has to be done when the text is finally in braille form, it can only be done by a trained braillist. These people are already in very short supply. Thus the need for ways and means of capturing text with a minimum of transcription is

critical.

Other obstacles for blind students

Limiting this discussion to access to information in one form or another, we will disregard other serious problems blind students encounter, such as mobility problems. There is however a kind of problem which is creeping up, and it is a reflection of the way we are becoming a computerized information society. It is becoming much more common for information services to be provided via computer terminals linked to distant computers. Thus one finds terminals in the larger campus libraries which enable one to search specialized data bases, whether they be Medlars on the East coast, A.P. Sharp in Toronto, or WestLaw in St. Paul. Some of these services are mediated by professionals, so students don't have to learn any of the quirks which may be involved in using them. Others, such as the Lexis service available through a terminal at the Law School, are intended to be used directly by the budding professional.

It is difficult to come up with a general solution that would allow a blind student to directly access the latter kind of service. Each one has to treated as a special case. For instance, some of these services are provided by using custom-designed terminal keyboards. In the case of Lexis, they use a number of special-function keys whose design requires proprietary information.

Conclusions

The UW-Madison has equipment and some computer software which can simplify the preparation of documents in braille. We hope this report helps you in actually producing tests, handouts, notes, and other material in braille. The author would appreciate knowing about your successes and failures in attempting to produce braille, so that this report can be revised to be more helpful.

Appendix A: MACC Rates for Braille Equipment Use

A MACC memorandum 49TS dated December 14, 1981 establishes what the rates for use of the LED-120 braille terminal shall be. These rates are in addition to the normal rates which apply when using the UNIVAC computer.

(1) unlimited use for one student for a semester	$200
(2) use per hour	$ 8
(3) use for document printing per 1000 lines	$ 7

These rates help MACC recover the cost of supplies (continuous form braille paper is very expensive) and the cost of keeping the equipment in good working order. It is possible to arrange with MACC to have its staff operate the braille terminal for you and mail the output to you. The arrangements and costs for this special handling can be worked out by calling the Contract Operations supervisor at MACC (608)-263-2067.

The rates given above apply to UW-Madison faculty, students and staff. If you are affiliated with another part of the UW System, a rate multiplier of 1.27 applies. Users external to the UW system fall in two categories (1) nonprofit (e.g., state agencies) for whic the multiplier is 1.49, and (2) commercial users for which the multiplier is 3.0 (commercial users have to demonstrate why they can't get the work done elsewhere).

As with all rates, the above rates and multipliers are subject to change. In some cases it may be possible to obtain financial support for the use of these services. Students who receive assistance from the state's Department of Vocational Rehabilitation (DVR) may be able to have DVR defray some or all of these costs.

Appendix B: A Short Guide to the NEAT Editor

Using NEAT

With most computers, from the very smallest to the very largest, you work with "files." A file can contain a document, a computer program, a list of numbers, etc., or all of these things. Each file has a distinct name. When you are about to create a file, you make up a name for it. You do that with a @CAT command (on the UNIVAC computer, most commands start with the symbol @). The fact of making up a name for a file does not put any information in the file; we will do that with @NEAT. If there is any chance that you might need to use the file again, say tomorrow, then you have to ask that it be saved, otherwise it will automatically disappear within 24 hours; saving a file is requested with a @SAVE. The next time you log on to the UNIVAC computer, you tell it you wish to use a particular file with a @ASG command. This may sound complicated, but it will be clearer as we walk through an example step-by-step, showing exactly how all these commands are used.

Dialogue with UNIVAC	Explanation
@CAT,P ED*MYFILE.	Identify "ED*MYFILE." as a new file
@NEAT,I ED*MYFILE.DOC1	Enter NEAT editor to input text to be called DOC1, within file ED*MYFILE.
type any desired text line by line except as noted	Any line not starting with either \ or @ is taken as ordinary text
until you type a new line starting with \done	Exit by typing new line which reads \done
@PRT,S ED*MYFILE.DOC1	Displays the content of the document you just completed.
@SAVE ED*MYFILE.	Save everything associated with ED*MYFILE, including DOC1, until further notice
@FIN	Finish this session with UNIVAC
....	...
	The next day ...
@RUN FREDDY,12345,888888	log in as usual
@ASG,AX ED*MYFILE.	Assign your file so the computer will have it handy for you
@NEAT,U ED*MYFILE.DOC1	Enter NEAT with U option, to

```
                                    update (change) your old document.
...                                 Make changes as shown later
\done

@FIN                                log out
```

Whenever you type an @ command or a file name, you can use lower case letters if you prefer; they will automatically be transformed into upper case letters.

How does one change a document? We are using change in the broadest sense. If you have a long document, you might type in the first few pages today (using @NEAT,I). Then you could come back later and type the next section by using @NEAT,U. Remember--the first time you begin to create a document, use the trio of commands @CAT, @NEAT,I and @SAVE. From then on, for every subsequent run, use a @ASG,AX and a @NEAT,U if you wish to even look at the document with NEAT's help.

Appending new text (adding text to the end) is accomplished simply by typing it. NEAT assumes that each new line you type should be placed at the end, unless you force it to do something else. Also, whenever you finish typing a line of text, NEAT will sound a beeper to let you know all is well. Suppose we just started NEAT and had typed the following:

```
@NEAT,I ED*MYFILE.DOC1
Why is it so difficult
to find good ezamples?
```

Oops! Typo--how do we replace the "z" with an "x"? NEAT maintains a current-line pointer. When you begin with the I option (@NEAT,I) your new file is empty and the current line pointer points to the top of the empty file. Each time you type a line, the current line pointer moves down one line. So now it is pointing to the last line "to find good ezamples?". To change part of a line you use the S (for Substitute) command. All NEAT commands begin with the character , and they must appear at the beginning of a line. To perform a substitution, you must provide an "old part" to be found, and a "new part" which is to replace it. Here we can specify "z" for the old part, and "x" for the new part. You would type the whole command as:

```
S.z.x
```

How could you delete a sequence of characters or a whole word? Simply specify nothing as being the "new part". You could delete the word "good" by typing

```
S.good.
```

Note that only the four letters g, o, o, and d will be deleted. If you also wanted the space character in front of "good" to disappear, you should have typed

S. good.

If you noticed it too late (i.e., "good" has already been erased) then just delete the extra space. How will NEAT know which space you want deleted? There is no confusion about which line we are working on; the current line pointer has not been changed. If we have the text "to find examples?" we want to remove one space from between find and examples.There are several ways of doing this; here are two.

S.d--e.d-e or S.--.-

where each space is marked by a -, for your benefit (you should push the space bar to type a space).

Suppose we wanted to change some other line. You refer to lines other than the current line by using line numbers. You can see what the line numbers are by using the Print command P. Type

::P

to obtain a printout or display of all the lines in your file, along with their line numbers. Here we would see:

```
1       |Why is it so difficult
2       |to find examples?
```

To change any part of line 1, type a "1" followed by the appropriate command. So to delete " so", type

1S. so.

To delete any desired line, say line 2, type "2D". To replace a line, first delete it, then start typing the new text on a new line. To append text following any desired line, say after line n, type "0 and then type the new text on new lines.

Now that you know how to initially place text in a file, to change any part of any line, to delete lines, to add lines, to replace lines, you can prepare documents.

In the preceding we used upper-case letters for all of the NEAT commands (D, P, S). You can use the lower case letters if you prefer. There are many other useful features in NEAT that you should look at, particularly if you plan on preparing very lengthy documents. In that case, by all means examine the NEAT reference manual and other UNIVAC documents

available at MACC. The "1100 TUTOR" manual should also be examined if the UNIVAC way of doing things is new to you.