BRAILLE AUTOMATION NEWSLETTER

December 1976

edited by

J.M. Gill and L.L. Clark

Warwick Research Unit for the
Blind,
University of Warwick,
Coventry CV4 7AL
England

American Foundation for the
Blind,
15 West 16th Street,
New York 10011
USA

Contents

Editorial

In the last few months, most of the developments in braille automation have involved new hardware. This does not mean that braille translation programs require no improvement. However writing the specifications for the next generation of braille translators is no trivial task. It will be complicated by the moves in various countries to revise their contracted braille systems. Also new types of translation program will be required if best use is to be made of microprocessors for producing braille. It has been suggested that the new programs should be natural language transferrable and should be capable of modification by someone with no knowledge of programming. The editors would welcome articles outlining the specifications for this next generation of translation programs.

The proceedings of the SIGCAPH/AFB workshop have not yet been published; the delay has been caused by Bob Gildea moving from Colorado to Massachusetts. Dr. M. Berkowitz has promised to send copies, free of charge, to all those on the BAN circulation list.

The International Guide to Aids and Appliances for Blind and Visually Impaired Persons will be published in the very near future. Copies can be ordered from The Publications Department, American Foundation for the Blind, 15 West 16th Street, New York New York 10011, USA.

The Design of a Fast Braille Lineprinter

Ir A.N. Westland

I. Introduction

The Dutch Library for the Blind uses at this moment, for its production of braille books in smaller quantities, modified braille typewriters ("Perkins"), which are driven by an electric motor and controlled by punched tape. These units have a speed of about four characters/second, working on normal braille paper (weight: 180 grams/sq. metre) coming from a roll of 250 mm width. These machines however, are used at speeds for which they were not designed, so they wear fast and produce braille of not too good a quality (unequal height of the braille dots and unequal distances between them).

In order to solve these problems: the relatively low speed combined with fast wear, leading to braille of poor quality, the laboratory of Fine Mechanics of the Mechanical Engineering Department of the Delft University of Technology was invited to design a new braille-embosser. After a study of the construction of normal printers the principle of line-printing was chosen, which means that each machine-cycle produces one new line of characters. In the case in question the printer is equipped with 40 braille-cells, each consisting of six pins in the braille-configuration. All pins that are to form the braille line are selected and then embossed, after which the paper is transported over one line distance (10 mm), ready for the next printing (embossing) cycle. This printer is able to print five lines/second and is equipped with a knife to cut the paper (coming from a roll) into pages after embossing.

II. Specifications to which the braille lineprinter complies

1. Distance from one braille-cell to the next (pitch):
   6 ± 0,05 mm (0.24 ± 0.002")

2. Distance between the lines:
   10 ± 0,1 mm (0.4 ± 0.004")

3. Height of the braille dots:
   0,5 ± 0,05 mm (0.02 ± 0.002")

4. Number of characters on one line: max. 40

5. Speed: 5 lines/second

6. Paper: Width 250 mm (9.85")
   Weight 180 grams/sq. metre
   (0.004 Oz/inch^2)

   No pin feed holes necessary.

III. <u>Description of the braille lineprinter</u>

When we consider the possibilities for production of a braille dot in paper by means of a pin and a hole, there are two methods:

(i) The movement of the pin is controlled by a force; the resulting displacement of the pin is a function of the force and of the characteristics of the braille paper between pin and hole.

(ii) The movement of the pin is controlled by a form; the force exerted on the pin is a function of this (forced) displacement and of the characteristics of the braille paper.

An example of the first method is hitting the pin with the armature of an electromagnet (or by any other sort of hammer), while the second method can be demonstrated by the example of "pressing" the pin in the hole by putting pin and hole between the jaws of a vice. The displacement of the pin (and so the height of the produced braille dot) is controlled by the displacement of the jaws of the vice (the "form" of the vice).

The ideal situation in the first case is that just enough energy is supplied to the pin to produce the braille dot (to deform the paper) and no system is needed to stop the pin in its movement after the correct height of the braille dot is reached.

One property of paper however is not being homogeneous in composition, so the amount of energy required to form a correct braille dot is not constant. So it is common practice to supply more energy than strictly needed and brake the pin (for instance by means of a collision) afterwards. These collisions mean extra noise and wear.

In this printer the second method was adopted: the movement of the pins that produce the braille is controlled by the form of the parts that generate this movement.

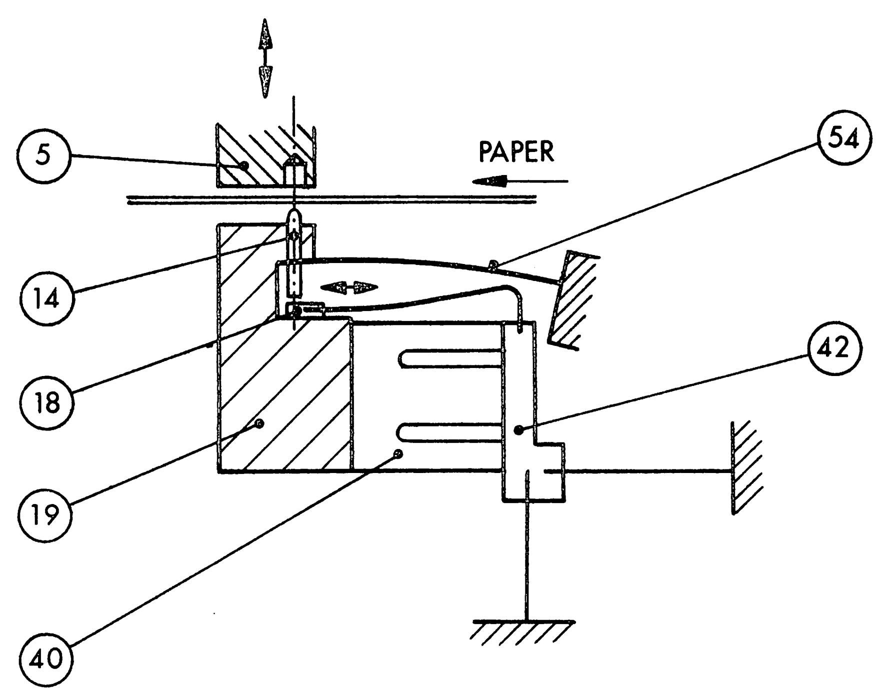

Fig. 1 The selection system

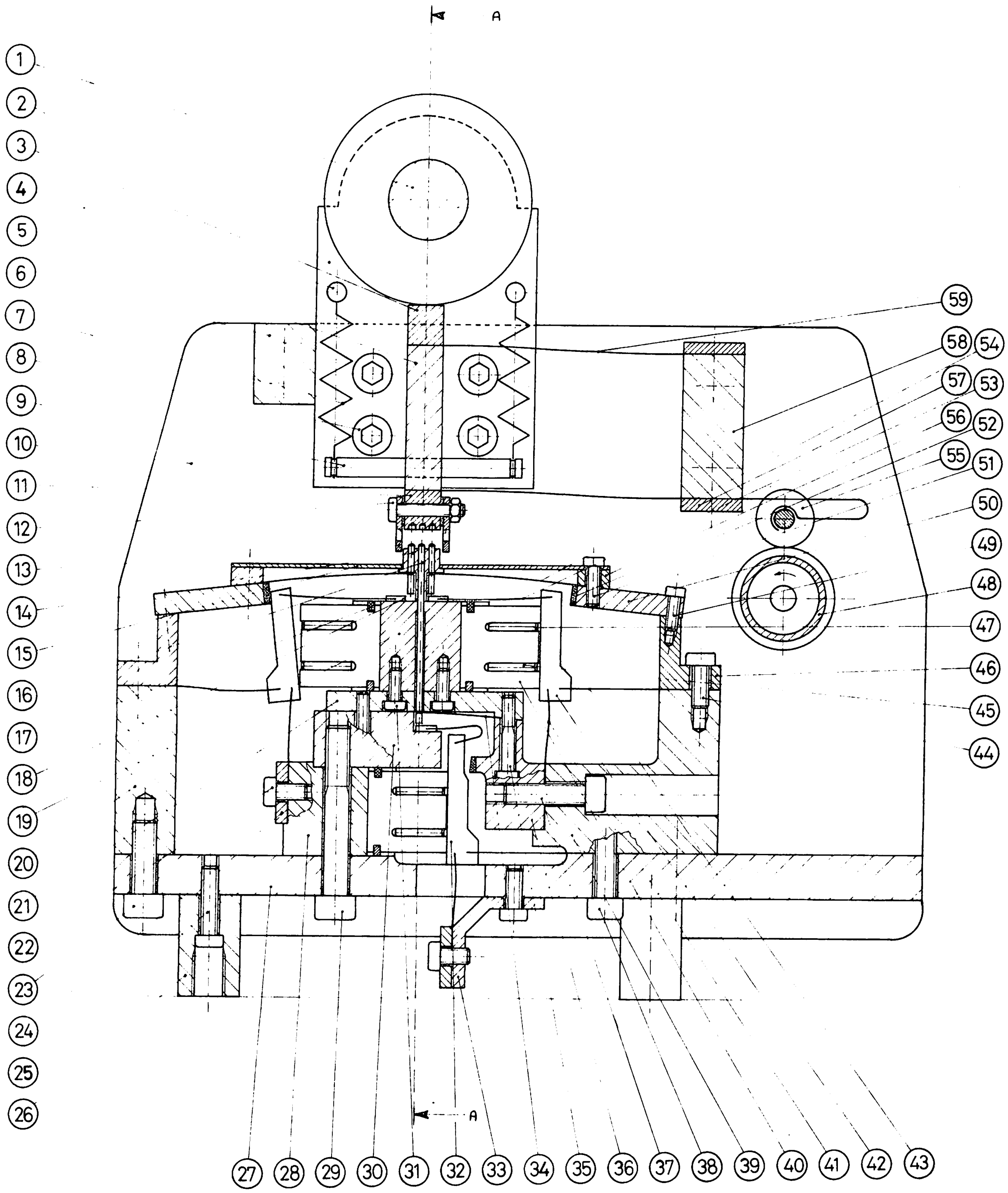

Fig. 2 The embossing mechanism

IV. The electronic control system

The electronic control system of the printer is set up for input by means of paper tape. So the input data is in series form, but for correct switching of the electromagnets, the information must be fed in parallel into those magnets of the 240 magnets, that are to produce a braille dot. So part of the control system is a 240 bits shift register for this function of buffer memory. Furthermore the system contains 240 amplifiers as members between the logic (TTL) level and the coils of the electromagnets. Other parts of the electronic control system concern the correct timing of the switching sequence for electromagnets and paper transport steppermotor, related to the position of the driving shaft (and thus the position of the upper beam).

V. Future development

The first results of test runs with this printer are satisfying regarding the attained speed and the accuracy of the dimensions of the produced braille.

This result was not reached without several dimensional variations of the leaf springs in the prototype. This was primarily necessary because certain dynamic effects caused premature breakage of some leaf springs. During this modification period it was found that the accessibility for service purposes should be improved. Certain parts that should be easily replaceable are rather inaccessible, so future development is aimed at the construction of a printer with less parts producing one line in two embossing cycles. This new printer should have 2/3 of the parts of the present one, but have the same speed of five lines of complete characters per second. This second prototype should be ready in 1977.

This project was and is carried out without any additional external financing, by integrating it in the current mechanical engineering education program.

Some Thoughts on PL/1 Braille Translators

P.W.F. Coleman

Introduction

These notes represent the core of my thinking on a structured approach to braille translators over the past five years. They are not meant to be complete in any way, for the process of thought is by no means yet complete. I am purposely soliciting others' thoughts on this problem.

The language I think in, like most programmers, is the one on which I cut my teeth, and I apologise to those not familiar with it. However, it is also relevant as the language in which my first translation program was written from which my thought has proceeded. I hope in a later Newsletter to give the fruit of my thought in a structured program in English which should be easy to transliterate into any convenient programming language.

Historical Note

Between 1968 and 1971, I produced a braille translation program for Standard English Braille written in PL/1. This was essentially an experiment in speed and program versatility, and I have good reason to believe that this is still the fastest braille translator extant.

Features of the program included a generalised input procedure (very simple in PL/1), in which the input device and its characteristics was specified at run time via job control language; compression of source text to eliminate unwanted blanks and blank lines, hyphenation on output and two extra braille grades (for technical and PL/1-orientated users). It also handled computer braille as well as Grade I literary braille. All these options could be set when invoking the program, and reset by commands embedded in the source text. These commands worked very well, but dynamic resetting of options does, of course, presuppose some degree of prior editing.

The program was not table-driven but data-driven, the rules for contracting being embedded in the actual code of the program, and not held in tabular form.

The central part of the algorithm was a series of statements which checked the current character to see if it was a letter: if it was not, then it was translated as it stood, the translation being an internal token consisting of a single EBCDIC character which was assigned to an output string OUTTEXT; the source text was held for this process in TEXT.

However, if a letter <u>was</u> detected, a similar check was made on the next character and if both were letters, the characters themselves were made the subscripts of a label array and a branch taken on its value:

```
IF¬((SUBSTR (TEXT, I, 1) < 'A'|
     SUBSTR (TEXT, I, 1) > 'Z') &
    (SUBSTR (TEXT, I+1, 1) < 'A'|
     SUBSTR (TEXT, I+1, 1) > 'Z')) THEN
   GOTO LABELS (UNSPEC (SUBSTR (TEXT, I, 1)),
                UNSPEC (SUBSTR (TEXT, I+1, 1)));
   ELSE GOTO AA;
```

A check is made specifically that neither of the current character pair is a letter and the branch made if the test is false. For those not familiar with PL/1, UNSPEC returns the internal representation in the machine of its argument, in bit form, and SUBSTR returns a substring from the first argument, the second argument giving the starting point relative to the beginning of the string and the third the length of the substring. Label AA was the point at which a one-for-one translation took place.

LABELS was declared:

```
DECLARE LABELS (191:231, 191:231) LABEL
        INITIAL ((41 * 41) AA);
```

The way the array was then set up can best be seen from an example. The letter pair AB could be the first two letters of either ABOUT or ABOVE, contracted respectively to AB and ABV. Therefore the corresponding element of LABELS would be initialised to AB - note that the bounds of each dimension of the array represent the extreme ends of the EBCDIC upper case alphabet:

```
LABELS (191, 192) = AB;
```

The code at label AB was:

```
IF SUBSTR (TEXT, I+2, 3) = 'OUT' THEN
DO;
     BRL2 = 'AB';
     GOTO I5J2;
END;
IF SUBSTR (TEXT, I+2, 3) = 'OVE' THEN
DO;
     BRL3 = 'ABV';
     GOTO I5J3;
END;
```

Contrast this with the code at label DO (remember that the word DO, contraction D, is implicit in the letter pair itself):

Here a check must be made on the surrounding context - is the letter pair preceded by a space and succeeded by a space or punctuation?

```
DO: IF SUBSTR (TEXT, I-1, 1) = ' ' THEN
       IF SUBSTR (TEXT, I+2, 1) < 'A' |
          SUBSTR (TEXT, I+2, 1) > 'Z' THEN
       DO;
          BRL1 = 'D';
          GOTO I2J1;
       END;
```

Before each such test another check was made to prevent the text pointer running off the end of the source text:

```
IF I <=ITEXT (3) THEN
```

preceded the test for both 'OUT' and 'OVE' above. SUBSCRIPTRANGE and STRINGRANGE could have been used to achieve the same thing, but that would only register that the text had been overshot; in this way overshooting is prevented.

The final branch in each section, to ImJn, resets I and J, the pointers to TEXT and OUTTEXT, and branches back to the check for the next letter pair. For example:

```
I3J2: I = I + 3;
      GOTO J2;
      . . .
J2:   IF J > 49 THEN /* END OF OUTPUT LINE */
      DO;
           VBRL = BRL2; /* TEMPORARY FOR NEW
                                           CONTRACTION */
           CALL HYPHEN; /* HYPHENATING OUTPUTTER */
      END;
      ELSE DO;
           SUBSTR (OUTTEXT, J, 2) = BRL2;
           J = J + 2;
           . . .
           GOTO NEXT;
      END;
```

All this branching was not, of course, good programming practice, and certainly made for difficulties when debugging. The obvious answer was to replace labels AB etc. by procedures, the pointer resetting being done in a parameter-driven procedure. Such wide use of procedures was standard practice in earlier braille translation programs, but would have incurred very heavy overheads in procedure prologues and epilogues on the IBM System/360 and similar machines (not half such a problem though on stack machines like the Burroughs range).

Current Thinking

For the past few years I have been working to circumvent this problem, trying to keep to a similar philosophy to the original program in order to maintain speed, but using a structured programming approach so as to simplify coding and testing. A by-product should also be a more compact program.

I have been working in PL/1, though I am currently generalising my approach as already mentioned. My current thinking is to replace the label array by a number of arrays of varying character strings. One such array or array group would contain contraction residues ('OUT', 'OVE' etc.), another array or array group having the contractions ('AB', 'ABV' etc.).

An array of integers would give the number of residues and contractions associated with each letter pair - if there were groups of residue and contraction arrays, then there would be a corresponding group of integer arrays. There would be another array or group containing bit strings, one bit for each possible rule to be checked, the bit settings for each string determining which rules were applied for a particular contraction (nothing new here, of course). The declarations might be:

```
DECLARE RESIDUES (191:231, 191:231, 10) CHARACTER (8) VARYING,
        CONTRACTIONS (191:231, 191:231, 10) CHARACTER (5)
                                                   VARYING,
        CONTRACTNUMBER (191:231, 191:231) FIXED BINARY,
        RULES (191:231, 191:231, 10) BIT (8);
```

When comparing residues and inserting contractions, two sets of values need to be known besides the above: the lengths respectively of residues and corresponding contractions. These can be calculated at the time of each comparison if storage is critical, but since this would add considerably to processor time, it might be better to calculate them once for all contractions at the beginning and to hold them in other arrays or array groups:

```
DECLARE (LENGTHR, LENGTHC) (191:231, 191:231, 10)
        FIXED BINARY;
LENGTHR (191, 192,2) = LENGTH (RESIDUES (191, 192,2));
LENGTHC (191, 192,2) = LENGTH (CONTRACTIONS (191, 192,2));
```

The values of the array elements could be set in the program by default, but could also be set by reading in tables from a file. In any event, this would be very much a table-driven program, in contrast to my previous code-driven one.

Thus, assuming a letter pair has been isolated, the following code might isolate the appropriate contraction:

```
DECLARE(SUB1, SUB2, CURRENT) FIXED BINARY;
. . .
SUB1 = UNSPEC (SUBSTR (TEXT, I, 1));
SUB2 = UNSPEC (SUBSTR (TEXT, I+1, 1));
/* SET TO INTERNAL REPRESENTATION OF LETTER PAIR */
/* CURRENT POINTS IN TURN TO CONTRACTIONS IN A GROUP */
IF SUBSTR (TEXT, I+2, LENGTHR (SUB1, SUB2, CURRENT)) =
     RESIDUES (SUB1, SUB2, CURRENT) THEN
DO;
     /* NOW CHECK EACH BIT SETTING OF THE CORRESPONDING
     RULES ELEMENT: IF TRUE APPLY THE APPROPRIATE RULE
     AND NOTE IF IT FAILS */
     . . .
     IF SUBSTR (RULES (SUB1, SUB2, CURRENT),3,1) THEN
          IF SUBSTR (TEXT, I-1, 1)¬ = ' ' THEN
               FAILED = '1'B;
```

```
    . . .
END;
IF ¬ FAILED THEN
DO;
     SUBSTR (OUTTEXT, J, LENGTHC (SUB1, SUB2, CURRENT))=
          CONTRACTIONS (SUB1, SUB2, CURRENT);
     I = I + LENGTHR (SUB1, SUB2, CURRENT);
     J = J + LENGTHC (SUB1, SUB2, CURRENT);
END;
```

The failure to find a contraction or rule match would result in a one-for-one translation, I and J being incremented by 1.

Note that the amount of executable code is minimal, being directly data-driven; also that we have again avoided using procedures; on any machine this program should generate very fast object code.

By contrast, the arrays will be large, a minimum of 300K bytes on an IBM System/370, 50K words on a Burroughs machine. This could be brought down by dealing in single letters rather than letter pairs, with residues one letter longer than otherwise. The respective figures are then around 8K bytes and 1.5K words. This would certainly make for a very compact program, at the expense of slower execution - though the speed reduction would probably be quite small, a reasonable tradeoff.

Most special cases may well be avoided by giving each contraction a priority relative to the rest, and having one smaller array for each priority level, each being in turn applied to the current letter or letter pair until a contraction and rule match is found.

However, special cases may still be treated as additional contractions, as they were in my original program. Thus the contraction WHEREVER would have a higher priority than, say, WHERE, and HERENCE than HERE.

Again, some special cases may also be avoided by alternate forward and reverse parsing of the source text word by word, as has already been done in several translators, and in standard PL/1 this is facilitated by the REVERSE function, e.g.:

```
/* ISOLATE A WORD IN THE TEXT, SETTING I TO POINT
   TO THE FIRST AND K TO THE LAST LETTER */
SUB1 = UNSPEC (SUBSTR (TEXT, K, 1));
SUB2 = UNSPEC (SUBSTR (TEXT, K-1, 1));
. . .
IF REVERSE (SUBSTR (TEXT, K-1-LENGTHR(SUB1, SUB2,
                                                  CURRENT),
   LENGTHR (SUB1, SUB2, CURRENT))) =
   REVERSE (RESIDUES (SUB1, SUB2, CURRENT)) THEN
   /* APPLY RULES RELATIVE TO (K-2-LENGTHR(SUB1,SUB2,
                                                  CURRENT))
      INSTEAD OF (I-1), AND (K +1) INSTEAD OF
           (I+2+LENGTHR (SUB1, SUB2, CURRENT))*/
```

In these notes I have only touched on the process of contraction-generation. I have not attempted to cover items such as handling numbers with or without an associated unit of measure, and the handling of punctuation including two-cell punctuation and opening and closing of quotation marks. I see the contraction problem as remaining the pre-eminent one, for this is the main process taking place in any braille translator.

I have for long been dissatisfied by the automatic recourse to trees and other complex data structures to handle this task. To me it has the flavour of the sledge-hammer cracking the nut, leading to some pretty slow translation.

It is for this reason that I am continuing work on my original lines using arrays, albeit character string rather than label, and in this respect my present efforts are just an attempt to refine and yet generalise the original concept.

This is not to say that the other methods are not valid, or may not in the end prove worthwhile. It is merely to justify my present work, to myself as much as anyone else.

Available Page Braille Embossers

J.M. Gill

This article summarises the main features of the four page braille embossers which are currently commercially-available. No attempt has been made to systematically evaluate these devices or even to verify the manufacturers' claims. However the names and addresses of known purchasers have been included so that you can obtain advice from those with experience in operating these embossers.

	Braillemboss	LED-1	LED-120	SAGEM
Maximum baud rate	150	110	1200	150
Page width (braille cells)	38 or 42	40	40	31 or 40
Quality of embossing	fair	excellent	good	excellent
Overall dimensions (mm)				
height	1555	940	1035	235
width	665	635	610	525
depth	580	635	840	525
Weight (kg)	84	114	114	25
Basic cost (US Dollars)	6500	6200	12000	3283
Keyboard included in basic cost	no	yes	yes	no
Double-sided embossing optional	no	no	no	yes
Advertised delivery (months)	0	3	3-4	6
Sales				
North America	8	5	27	0
Europe	5	0	3	7
Other Continents	1	0	0	0

Table 1 Some features of the page braille embossers

Braillemboss

Manufacturer:	Rehabilitation Engineering, Building 31-063, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139, USA.
Foreign agents:	None.
Dimensions:	665 x 580 x 1555 mm high
Weight:	84 kg
Power requirements:	115 volts, 60Hz or 240 volts, 50Hz
Speed:	up to 16 cps
Interfaces:	EIA, TTY, others optional
Input codes:	ASCII
Delivery:	Immediate
Price:	US $6500.00
Guarantee:	None but every assistance is given to users in the form of advice on trouble shooting, spare parts when available or technical drawings.
Users: 1.	Sensory Aids Evaluation and Development Center, MIT, USA.
2.	Dr. J. Morrison, Department of Transportation, Massachusetts, USA.
3.	Perkins School for the Blind, Watertown, Massachusetts, USA.
4.	Worcester Polytechnical Institute, USA.

5. Honeywell, Phoenix, Arizona, USA.

6. D. Keeping, University of Manitoba, Canada.

7. T. Hicks, Engine Division, Rolls Royce, Bristol, England.

8. J. McSpadden, Internal Revenue Service, Little Rock, Arkansas, USA.

9. Arkansas Enterprises for the Blind, USA

10. Penn State University, USA

11. Learning Resource Center, Middle Tennessee State University, USA

12. Scientific Research Foundation, Israel.

13. University of Nottingham, England.

14. Warwick Research Unit for the Blind, University of Warwick, Coventry, England.

15. University of Lulea, Sweden

16. University of Bradford, England

LED-1

Manufacturer:	Triformation Systems Inc., 3132 S.E. Jay Street, P.O. Box 2433, Stuart, Florida 33494, USA. Tel. (305) 283-4817
European agent:	None.
Dimensions:	25 x 25 x 37 inches high
Weight:	250 lbs.
Power requirement:	115 V.A.C., 60Hz, 300 Watts
Speed:	10 cps
Interfaces:	TTY, EIA, others optional
Input codes:	ASCII, Baudot, others optional
Delivery:	90 days
Price:	US $6,200.00
Guarantee:	None

Purchasers:

1. J. Covici, 2 World Trade Center, 19th Floor, New York, NY 10048, USA
2. Western Union Co., 308 West Rt. 38, Moorestown, N.J. 08057, USA
3. Dow Chem. Co., Freeport, Texas 77541, USA

4. University of West Ont., Dept. of Computer Science, Room 2036, Eng. Science Bldg., London 72, Ontario, Canada

5. WLEN Radio, 149 1/2 So. Main St., Adrian, Mich. 49221, USA

LED-120

Manufacturer:	Triformation Systems Inc., 3132 S.E. Jay Street, P.O. Box 2433, Stuart, Florida 33494, USA Tel. (305) 283-4817
European agent:	None
Dimensions:	25 x 30 x 37 inches high
Power requirement:	115 V.A.C., 60Hz (other voltages available at extra cost)
Speed:	120 cps
Interfaces:	EIA, TTY, others optional
Input codes:	ASCII, EBCDIC, BCD, Correspondence, Baudot, others optional
Delivery:	90 to 120 days
Price:	$12,000.00
Guarantee:	None

Purchasers:

1. Arkansas Enterprises F/T Blind, 2811 Fair Park Blvd., Little Rock, Arkansas 72204, USA
2. John Merz, 109 Maple Lane, Hillsmere Shores, Maryland 21404, USA
3. University of Calif., Berkley
4. General Mills, Inc., 9200 Wayzata Blvd., Minneapolis, Minn. 55426

5. State of Information, Systems Center, State of Utah, Room 104-State Office Bldg., Salt Lake City, Utah 84114, USA

6. University of Manitoba, Computer Center, 603 Engineering Bldg., Winnipeg, Manitoba, Canada R3T 2N2

7. Curtis Landtroop, General Motors Corp., 767 Fifth Avenue, 25th Floor, New York, NY 10022, USA

8. Rev. Wesley Price, Protestant Guild F/T Blind, Inc., 456 Belmont St., Watertown, Mass. 02172, USA

9. University of Louisville, Computer Services F/T Blind, Room 358-Life Science Bldg., Louisville, Kentucky 40208, USA

10. Dr. James Slagle-Code 5407, Naval Research Lab., 4555 Overlook Ave. S.W., Washington D.C. 20375, USA

11. Robert Watson, State Services F/T Blind, 1745 University Ave., St. Paul, Minn. 55104, USA

12. Mel Sauer, P.O. Box 660, c/o Kolt Radio Station, Scottsbluff, Nebraska 69361, USA

13. Internal Revenue Service, 3505 Broadway, 7th Floor, Oakland, Calif. 94611, USA

14. Internal Revenue Service, 600 Arch St., 6th Floor, Phila., PA. 19106, USA

15. Internal Revenue Service, 301 N. Lamar St., Jackson, Miss. 39202, USA

16. Volunteer Services F/T Blind, 919 Walnut Street, Phila., PA. 19107, USA

17. Learning Systems International Ltd., Canadian Gov., Systems Center, Gov. Adm. Bldg., Regina, Saskatchewan, Canada

18. Computer Science Center, University of Maryland, College Park, MD. 20740, USA

19. Nederlandsche Blindenbibliotheek, Noordwal 7, 'S-Gravenhage, The Netherlands

20. Columbus Technical Institute, Resource Center F/T Blind, Aquinas Hall, RM 107, 550 East Spring Street, Columbus, Ohio 43215, USA

21. Internal Revenue Service, 412 Main St., Wichita, Kansas 67202, USA

22. IBM Corp., Essex Junction, Vermont 05452, USA

23. Dept. of Navy, 1420 Eads St., South Arlington, VA., USA

24. Social Security Adm., Jersey City Telesensory Center, 30 Montgomery St., 7th Floor, Jersey City, N.J., USA

25. Board of Ed. Services F/T Blind, Eng. Bldg., University of Conn., Storrs, Conn., USA

26. Rockwell International, 12214 Lakewood, Downey, CA 90241, USA

27. Royal National Institute for the Blind, 224 Great Portland St., London W1N 6AA England

28. Phillip Hall, 80 Dalbury St., Apt. 210, Worcester, Mass. 01609, USA

29. Northwest Foundation F/T Blind, 3411 So. Alaska St., Seattle, Wash. 98118 USA

30. Warwick Research Unit for the Blind, University of Warwick, Coventry CV4 7AL, England.

SAGEM

Manufacturer:	Societe d'Applications Generales d'Electricite et de Mecanique (SAGEM), 6 Avenue d'Iena, 75783 Paris Cedex 16, France. Tel. 745.14.60 Telex: 610762F Cables: Telesagem - Paris
Agents:	M. Passemard, SAGEM Corp., P.O. Box 445, 35 South Main Street, Derry, New Hampshire 03038, USA (Tel. (603) 432-2013 and Telex 940478). SAGEM B, 51 Rue d'Arlon, BP 12, 1040 Brussels, Belgium. V. Thompson, 11 Poulton Avenue, Sutton, Surrey, England (Tel. 01-644 6402) SEBS Nederland, Kanaalweg 25-27, BP 174, 3130 Capelle A/D Ijssel, The Netherlands. Hans Puttgen, Grev Turegatan 73, 11438 Stockholm, Sweden. SAGEM have agents in many other countries - please contact Paris office for names and addresses.
Prices:	TEM 8BR Electronic modular terminal, send receive, 8 levels for embossing "Braille" characters (embossing of 31 characters per line, paper width 240 mm) Unit price, ex-works ... FF. 20.900,00

Options

- punching block with logic cards ... FF. 1.860,00
- tape reader with logic cards ... FF. 1.240,00
- both sides embossing ... FF. 1.350,00
- embossing of 40 characters per line, paper width 310 mm according to data processing requirements ... FF. 800,00

REM 8BR

Electronic modular terminal, receive only, 8 levels, for embossing "Braille" characters (embossing of 31 characters per line)

Unit price, ex-works ... FF.16.830,00

Options

- built-in punching block with logic cards ... FF. 1.860,00
- both sides embossing ... FF. 1.350,00
- embossing of 40 characters per line, paper width 310 mm according to data processing requirements ... FF. 800,00

(Note US $1 is about 5 French Francs)

Dimensions: 525 x 525 x 235 mm high (c. 20 x 20 x 9.25 inches high) for REM 8BR. The TEM 8BR is 630 mm deep.

Power requirements: 120 VA at 115/127/220/240 volts, 50/60Hz

Speed: 110 or 150 bauds

Code:	CCITT No. 5, ASCII, others optional
Interfaces:	EIA, TTY, others optional
Delivery:	TEM 8BR - 8 months REM 8BR - 6 months
Maintenance:	A technician can be trained free of charge in Paris (does not include travel or hotel expenses). The maintenance manuals are supplied free of charge in various languages including French and English.
Guarantee:	The equipment is guaranteed for a period of 12 months against any defect of design and manufacture, from the date of receipt of the equipment. During this period, SAGEM will supply or repair, free of charge, any part recognised as faulty further to unusual wear, provided its use has conformed with the manufacturer's standards.
Purchasers:	Only started marketing the equipment in Autumn 1976 and there are a considerable number of enquiries from potential purchasers. Sales so far:
1.	Conservatoire National des Arts et Metiers, 292 Rue Saint Martin, 75141 Paris Cedex 03, France.
2.	Ecole Nationale Superieure des Telecommunications, 46 Rue Barrault, 75013 Paris, France.
3.	Association pour la Sauvegarde des Enfants Invalides, 22 Rue de la Croix Baragnon, 31073 Toulouse Cedex, France.

4. Association Valentin Hauy, 3 A 9 Rue Duroc, 75007 Paris, France.

5. Ministere des Finances, Chef du Bureau M3, 9 Rue Croix des Petits Champs, 75001 Paris, France.

6. Amicale des Standardistes Aveugles de France, 16 Rue de la Folie Regnault, 75011 Paris, France.

7. Universite Paul Sabatier (Ver Informatique), 116 Route de Narbonne, 31077 Toulouse Cedex, France.

New Braille System in England

The annual report for 1975/76 of the Royal National Institute for the Blind says: "The report of an ad hoc committee of experts on braille production, particularly by computerised methods, has been received and adopted. A search for new premises is in progress to house what it is anticipated will be the most modern braille production centre in the world, more than doubling the RNIB's present output, thus extending the range of subjects and titles to meet the ever-increasing demand of braille readers, particularly students and those engaged in professional occupations."

RNIB are advertising for a programmer to work on this system. The advertisement also says "An early start will be made on an improved version of the Fortran text translation program (possibly using CORAL 66 or Algol)."

Compositor's Tapes in Czechoslovakia

The Czechs, who use uncontracted braille, plan to start using compositors' tapes for driving a stereotype machine. Ing. K. Vrana has developed a mini-computer program to automatically hyphenate words at the end of a line; this is technically feasible since Czech is a phonetic language. The system should be operational by the end of 1976.

Further details from Ing. K. Vrana, Vyvojove Dilny Cs. Akademie Ved, Husova 4, Praha 1, 110.00 Czechoslovakia.

Publications and Reports

Brown A. "Braille Remote Computer Terminal".

Computer Centre, Monash University, Australia, 1975, 31 pp.

The report first establishes the design specifications for a braille terminal and then discusses the problems of physical realization. The system decided upon incorporated an IBM Braille Electric Typewriter and a PDP 11/10 minicomputer.

Browne J. "Information Needs of Blind Lawyers".

Warwick Research Unit for the Blind, Sept. 1976, 37 pp.

The report includes a brief survey on the availability of legal material in error-free computer-compatible form. The most significant finding, in this respect, was that Her Majesty's Stationery Office (HMSO) make available legal statutes on digital magnetic tape.

Charpentier J.M. "Etude d'un Editeur Braille".

Conservatoire National des Arts et Metiers, Paris, November 1976, 25 pp.

This report, in French, gives further details of the project described in Braille Automation Newsletter, August 1976, pp. 85-89. Copies obtainable from Laboratoire de recherche pour la reinsertion professionnelle des handicapes, Conservatoire National des Arts et Metiers, 292 Rue Saint-Martin, 75141 Paris Cedex 03, France.

de Jong E.H. "A Braille Translator".

Massachusetts Institute of Technology, U.S.A.
January 1976, 67 pp.

The main text documents a generalised minicomputer program written in ANSI Fortran for ink print to braille translation. The program produces a crude approximation to grade II braille and translates at about 10 words per minute.

Küppers H.J. "Braille Printing Techniques: Final Report".

Stiftung Rehabilitation, Heidelberg, 1976, 26 pp.

This report covers text editing for braille production, the use of compositor's tapes, digitally-controlled stereotyper and systems for producing embossed graphics. Copies, in German or English, are available from Dr.-Ing. H.-J. Küppers, Forschungszentrum fur Rehabilitation und Pravention, 6900 Heidelberg 1, Postfach 10 14 09, German Federal Republic.

Loeber, N.C. "Modified Standard Office Equipment for Braille Translation and Embossing".

IBM Corporation, Systems Communication Division, Los Gatos, California, U.S.A., May 1976, 12 pp.

This report describes the interfacing of a braille page embosser to a word-processing typewriter. The 'translation' is to a new braille code developed by the author.

Michel M. "Computer-produced Braille Translation of Serials: A Demonstration, Feasibility Study, and Implications for Librarianship".

City University of New York, 1976, 18 pp.

Copies available from M. Michel, 4205 17th Avenue, Brooklyn, New York 11204, U.S.A.

The report concentrates on the problems of producing single copies of articles in journals in contracted braille. This report is in the form of a proposal for a research project.

Snelders J.A.H. and Spanjersberg H.A. "Braille Apparatus Based on Microprocessors".

Switching Techniques and Data Processing Laboratory, Department of Electrical Engineering, Delft University of Technology, Mekelweg 4, Delft, The Netherlands, November 1976, 25 pp. Also published in Dutch in the Review of the Dutch Electronics and Radio Association (N.E.R.G.), Vol. 41, No. 4, 1976.

The paper describes a microprocessor-based braille system consisting of an input program, a code converter, a correction program and an output program to drive a braille page embosser.

"DJ's Plea, IBM Creative Development Effort Spark IBMers' Search for Tools for Blind".

IBM News, Vol. 6, No. 11, 24 Sept. 1976, 2 pp.

A single cell braille display has been interfaced to a word-processing typewriter. The article describes, in journalistic phraseology, the development of this device. A dozen systems are now being built for evaluation. Further details from M. Pandich,

Communications Manager, IBM Corporation, Box 12195, Research Triangle Park, North Carolina 27709, U.S.A.

"Rylec i uklady scalone".

Polska (Nasza Ojczyzna), No. 8, 1976, p. 9, 11.

The article describes the page braille embosser developed by Mgr inz Wojciech Zawistowski. The embosser operates at speeds up to 12 characters per second and it is claimed that it will cost not more than $160.

Further information can be obtained from Mgr inz W. Zawistowski, 00-901 Warszawa, Computation Centre, P.O. Box 22, Palace of Culture, Poland.