

日本版 UBC 点字標準化と支援ソフトウェアに関する研究

(10680295)

平成 10 年度～平成 11 年度科学研究費補助金（基盤研究(C)(1)）
研究成果報告書

平成 12 年 3 月

研究代表者 大 武 信 之
(筑波技術短期大学 教育方法開発センター 助教授)

はしがき

日本点字が制定され、すでに百年がたつ。時代の変遷と共に、点字の定義も変更・修正され、近年ではほぼ10年おきに改定されている。点字に対し、一般に印刷された文字を墨字と呼ぶが、墨字で印刷される漢字は、点字において「読み」を仮名表記するため、漢字が持つ象形文字としての意味が点字で表すことが困難である。過去において、漢字を点字で表現する方法が考案されたが、現在ではほとんど使われていない。これは、高度情報化機器の普及により、視覚障害者でも、墨字文書の読み書きが出来るようになったからである。取り分けパソコンと音声合成装置の発達により、全盲でも第三者の助けを借りることなく、自由に漢字仮名交じり文書を書けるようになった。高度情報化機器の発達により、点字を使用しなくても、文字情報の交換が出来るように、視覚障害者の環境は大きく変わったが、点字の担う役割は変わっていない。つまり、一般社会においてペーパーレス化が進んでも、一般印刷物が無くならないのと同様に、点字が無くなることはない。従って、時代に合せた点字の定義も、日本点字委員会により、十年単位に改定が成されてきた。しかし近年は、大きな変更もないため、現状の日本点字規則が定着している。

日本点字規則に大きな変更がないのは、現状の規則に問題がない訳ではない。文脈に依存した曖昧な記述や、数式や化学式、楽譜など墨字をうまく表現できない部分を多大に残している。点字が6点表示であるため、点字パターンと、墨字の象形文字とのギャップを埋めるには無理がある。現在の日本点字規則は、過去の経緯と歴史を考慮すれば、多少使いづらい面もあるが、これを維持・継続使用していこうという認識が持たれている。ただし、現状の問題点を解消しないで良いと考えている者は少なくない。しかし、問題解消には大きな変更をしないことには、解決がなされないため、教育現場での混乱、既点字習得者の再学習など様々な問題があるため、二の足を踏んでいる状態にある。点字の抱える同様の問題は諸外国にもあり、同じ英語圏においても、英国式・米国式と違いがあり不自由さが存在する。

80年代後半から英語圏において、英語点字の持つ問題点を解消し、新たな英語点字を制定するという動きが出てきた。これは Unified English Braille Code と呼ばれる新点字体系（通称 UBC）を作成するものである。UBC の目的は、これまでの問題を解消し、英語圏での統一を図り、新規に作成する体系であるから、コンピュータ処理を考慮した定義にしようとするものである。UBC に対し、90 年に入り日本でもこれを積極的に評価し、導入する動きが生まれ、日本版 UBC の活動が始まった。本研究では、UBC 普及とそれに関わる調査研究、および UBC を対象とした支援ソフトウェアである日本語自動点訳システムの研究開発を研究の範囲とした。

研究組織

研究代表者：大 武 信 之（筑波技術短期大学 教育方法開発センター 助教授）
研究分担者：石 川 准（静岡県立大学 国際関係学部 教授）

研究経費

平成10年度	2,000 千円
平成11年度	800 千円
計	2,800 千円

研究発表

- (1) 曾根, 佐藤, 原, 大武: UJBC 文書の墨字翻訳システム II, 電子情報通信学会, 信学技報 ET98-93, pp.1-8, Dec., 1998.
- (2) 十川, 佐藤, 原, 大武: 墨字点字双方向変換統合システムの開発, 電子情報通信学会, 信学技報 ET98-94, pp.9-16, Dec., 1998.
- (3) Ohtake,N., Hara,S., et.al.: A Check system of Japanese \LaTeX document for the blind, Advanced Research in Computers and Communications in Education, Vol.II, IOS Press, Nov., 1999. (ISBN1-58603-027-2)
- (5) 楠, 佐藤, 原, 大武: 音声を用いた視覚障害者用科学技術文書作成システム, 電子情報通信学会, 信学技報 ET99-76, pp.97-104, Dec., 1999.
- (6) 宇田川, 新村, 川満, 楠, 渡部, 佐藤, 原, 大武: XML を用いた点字教材とその応用, 電子情報通信学会, 信学技報 ET99-82, pp.143-150, Dec., 1999.

目次

I	統一点字規則	1
1	UBC	1
1.1	背景	1
1.2	誕生と現状	1
1.3	最初の評価	1
1.4	波及	2
1.5	第2回評価	2
1.6	最終評価	2
1.7	英語圏外への波及	2
1.8	第2委員会	3
2	ガイドライン	3
2.1	指針	3
2.2	委員会の任務	3
2.3	任務の解釈	4
2.4	作業基盤	4
2.5	その他一般原則	4
2.6	明示規則	4
2.6.1	読みの規則	5
2.6.2	点訳の規則	5
2.6.3	デザインの規則	5
3	UJBC	5
3.1	日本版UBC	5
3.2	開発ソフトウェア	6
3.3	関連研究	6
4	成果論文	6
II	日本語英語自動点訳	54
5	研究目的	54
6	基本操作	54
6.1	対象：点字編集ができる	54
6.2	対象：点字編集はできないが、かな編集ならできる	57
6.3	対象：原文が読めない、あるいは大至急点訳したい	59
7	機能解説	61
7.1	作業1～作業5	61
7.2	オプション・メニュー	61
7.3	入力ファイルの指定（点訳したいファイルの指定）	61
7.4	出力ファイルの指定	62
7.5	点訳に関する設定	62

7.6	レイアウト	63
7.7	点字印刷	64
7.8	プリンタ設定	64
7.9	オプション設定	65
7.10	作業開始	66
7.11	連続処理	66
7.12	状況	66
7.13	内容	66
7.14	補足	67
8	機能リファレンス	67
8.1	入力ファイル形式	67
8.2	出力ファイル形式	68
8.3	点字表記	69
8.4	レイアウト	70
8.5	プリンタ設定	70
8.6	印刷形式	70
8.7	印刷設定	71
8.8	前処理	71
8.9	ドルコマンド	72
8.10	英語点訳についての補足	74
8.11	囲み記号	75
8.12	その他	75
9	ユーザー辞書	76
9.1	ユーザー辞書とは	76
9.2	登録作業の手順	76
9.3	品詞コード	76
9.4	エディタでのユーザー辞書作成	77
A	EXTRA インストール方法	78
A.1	EXTRA for Windows のインストールプログラム	78
A.2	プリンタドライバ	78
A.3	点字フォント	81

Part I

統一点字規則

1 UBC

1.1 背景

現在、世界の英語を話す国々では、いくつかの点字コード体系が使われている。技術文書ではないものに使われる標準英語点字コードは、通常文書コードと呼ばれているが、それに加え数学、コンピュータ、化学、音楽(楽譜)等のコードが存在する。これらのコードの各々が、設計された各分野では妥当なものとされているが、他分野では衝突や矛盾が起きている。ある点字パターンが、ある分野と他分野では意味が違い、また別の分野ではさらに違う意味になることもある。加えて文書コードが設計されたのは、盲人があまり教育を受けなかった時代のことであったため、現代の非技術的文書の中にも頻繁に出てくるような、多くの簡単な記号も含まれていなかった。さらに事態を複雑にしているのは、英語圏の国々で、ある特定の分野用として使われているコードが、程度の差こそあれ、やはり一致していない点にある。例えば、アメリカで公式に使われている文書用コードは、英国のものと似かよってはいるが違っている点がある。両国ではまったく違う数学コードを使い、さらにオーストラリアの数学コードはその二つのいずれともいくつかの点で違っている。結果、アメリカの点字利用者は英国で点訳された数学のテキストを読むためには、まず英国の数学点字コードを覚えなければならない。数少ない点字図書を共有することが重要であるにもかかわらず、同一英語圏にありながら、一つの英語使用国の点字図書を、他の国では有効に使うことが出来ないということになる。

1.2 誕生と現状

1991年にT.V. クランマーとエイブラハム・ネメス¹は、北米点字委員会(BANA: Braille Authority of North America)に、英語圏で使われている点字コードの統一計画を提出した。同年春のBANA会議においてこの提案は注目され、同年秋のBANA会議では、両氏の統一英語点字(UBC: Unified Braille Code)の詳しい設計と評価の計画案が提出された。計画は委員会に採用され、このプロジェクトの個々の側面を担当するため、いくつかの委員会が設立された。その最初の委員会、第1委員会は現在英語圏で使われている全ての点字コードの比較を行ない、第2委員会はUBCの設計に当り、ほぼ完成している。その他委員会は、Grade II点字で認められている略字を査定する委員会、点字書の書式を提案する委員会、英語文脈中の外国語の書き方の規則を提案する委員会、UBCプロジェクト参加各国の点字委員会に最終的な採否をはかる前に行う評価のための委員会などがある。なおこれらの委員会は、第2委員会の仕事が終了後に活動する形がとられた。

1.3 最初の評価

UBCの最初の草案はアメリカとカナダの点字の読み手、教師そして点訳者の評価を受けた。これは詳細なものではないが、点字の世界で受け入れられる最終案を生み出すために必要な情報を回収した。予想通り、評価は好意的なものばかりではなく、統一点字コード不要論や、割り付けられた記号のいくつかに反対などもあった。数式を伴う文章を読んだ点字読者の中には、現在広く使われている下がり数字の代わりに高位の数字を使うことに疑問を抱く人もおり、技術系ではない文書だけを読む年輩の読者は、現行点字で十分で、今の文書用コードを読みこなす彼らの経験を無にする全く異なるコードを導入する理由はないという意見もあった。これらの不満はあるものの、最初の草案に対する評価はおおむね好意的であった。

¹数式点字体系の1つであるネメス・コードの草案者。

1.4 波及

国際英語点字協議金 (ICEB: International Council on English Braille) は、UBC が国際的に意味を持つものとして関心を抱き、UBC が英語圏全体の統一コードとして採用されればという願望があったため、この種の計画が英語圏全体が参加した国際組織のもとで遂行されることに意義があると考えはじめていた。そのため 1993 年 7 月オーストラリアでの ICEB の会議において、BANA はこのプロジェクトを ICEB に移管することを提案した。ICEB はこの提案を受け入れ、BANA はプロジェクトに対する強い関心は持ち続けているものの、活動の主体ではなくなり、UBC は ICEB の管轄となった。この再編成の結果、いくつかの変更があり、英語圏全体で使われる UBC の発展が今やはっきりした目標となった。ICEB の参加各国がプロジェクトを推し進める委員会に代表を送ることになり、メンバーは ICEB 参加の全ての国の代表からなった。UBC の設計に責任を持つ第 2 委員会は、当初からの 4 名の委員に、現在は英国、オーストラリア、ニュージーランド、南アフリカが加わっている。カナダからの委員は参加していないが、第 2 委員会が報告を提出する上位の委員会の委員長として第 2 委員会の仕事の詳細を把握しているため、カナダの利害も正当に反映されている。アイルランド共和国には代表を出すようにと働きかけているが、今のところまだ参加がない。なお、フォーラムには第 2 委員会のメンバーだけではなくオブザーバーも参加し、同じように委員が交換するメッセージを受け取ることができ、第 2 委員会の委員長が提供する期間中はフォーラムあてに質問を出し議論に参加することもできる。

1.5 第 2 回評価

現在の状況は、コードはほぼ完成している。化学で必要な記号を与えるための作業はまだ残っているが、コンピュータ表記のための記号を全て供給する仕事は残りわずかである。なお数式用記号はほとんど定義されており、今や自然科学、社会科学のほとんどの文書は UBC で正確に表現することができる。まだ仕事は残っているが、UBC 最終案は固まりつつあり、点字の読者が出会うほとんどの文書は UBC で扱うことができるようになる。第 2 回の評価が終った段階で、第 1 回同様、評価は点字の読者、点字の教師、点訳者により行われた。第 2 回評価は、アメリカとカナダに加え、英国、オーストラリア、ニュージーランド、南アフリカでも行われた。評価の手順は、第 2 次の試案の検討から始まり、変更される記号と、現在使われているコードにはない記号の例が挙げられている新しいコードで書かれた文書を渡し、これを評価する。これには、現在使われていて新コードでは廃止されるコードの例もいくつか含まれている。また高校の数学・物理程度の数式の含まれた例もあるが、高等数学で扱う高度に技術的な文書は含まれていない。理由は評価をする人の多くは高等数学の式の評価は不可能で、後日適正な資格を持つ人々による評価が行われるべきとされたからである。

1.6 最終評価

新しいコードは点訳しやすいか、教えやすいか、簡単に学べるか、読みやすいかを探るため 3 回目の評価段階が必要となる。点訳者によって新しいコードで点訳されたテキストを使い、子供と成人に UBC を教えることになる。この評価の段階が全て終了し、評価の結果がおおむね好意的なものであれば、参加各国の点字委員会に UBC を出版用公式コードとして採用するように要請することになる。コードが全ての国で採用されれば、英語圏全体が単一の統一点字コードを使い、点字読者は英語圏のどこの国で点訳されたものでも読むことができるようになる。

1.7 英語圏外への波及

UBC の作業が開始された時点で UBC は、英語圏での共通コード作りが目標であった。他言語圏で UBC に興味を抱くことは考えてはいなかった。特に、日本が統一英語点字の技術的分野の記述の部分を採用する動きを示していることは、UBC 作業部会にとっては驚きであるようである。

1.8 第2委員会

第2委員会は、もともと ICEB の研究プロジェクトである統一英語点字コード (UBC) への第一着手として、基本文書コード (English Code) の拡張のための基本原則を完成する任務をもって発足したものである。1993 年になり、同委員会の任務に、UBC コードの数学、コンピュータ・サイエンス、その他専門分野への拡張を実施する作業が付け加えられ、成果についても記載されている。1992 年 11 月の BANA の最終報告書、1994 年の ICEB の中間報告書を含む従来の同委員会の全ての報告書の集大成であると言える。UBC の正式規則として決定されてからでも、その全てもしくは一部について、各国の国内点字委員会の承認を得ることが必要である。これは、承認手続きが完全に終わって初めて UBC コードが正式な共通公式コードとして認定されたものになるためである。

この報告書は ICEB の統一点字コード研究プロジェクトの一つの課題である第2基本コードの拡張を遂行するために構成された委員会が発表したものである。統一コード研究プロジェクトの目標は、数学、コンピュータ・サイエンス、その他の科学、工学関係や普通の英文に使用される際の共通の点字コードの開発である。統一コードはその大部分において現在主として使用されている一般の文書である英語点字 (EB) システムからかけ離れたものであってはならない。第2委員会は、後続する各委員会の仕事を通じ、コードが互いに矛盾せず、曖昧な部分を残さないように、コードを拡張する過程の基礎的な方法論を決定すること、さらに、数学やコンピュータ・プログラミング等の技術分野にまで実際に拡張するものである。

2 ガイドライン

2.1 指針

統一点字コード研究プロジェクトのいくつかの技術的ガイドラインは第2委員会の仕事に関連している。したがってそれらのガイドラインを読者の便宜のため以下に再録する。統一コードのガイドラインは6項目あり、次の通りである。

- (a) 点字 1 マスに 6 つの点を使用する。
- (b) Grade II 点字の略字に大きな変更を加えることなく、Grade I, Grade II 点字を取り入れる。
- (c) 点字の読み手の初心者と習熟者の両者に共に使いやすいこと。
- (d) 読みとりやすさを損なうことなしに、最大限のコンピュータ処理が可能であるようにする。そのため墨字から点字、点字から墨字へ相互変換が可能であり、また墨字の各記号を表す曖昧でない点字を使用する。
- (e) 教科書、数学、コンピュータ、その他技術的なコード (楽譜記号を除く) に留意する。
- (f) 作成に当たっては、既に発表されたあらゆる英文点字コードを考慮する。

2.2 委員会の任務

前節ガイドラインは、さらに以下の技術的な細目と共に委員会の任務に加えられた。

- (g) 様々な科学的・技術的分野の記号体系をカバーできるように、基礎的な文書コードの拡張のための一般的方法を開発する。
- (h) 用語の定義をする。
- (i) 記号 (図形記号と指示符) の領域を決定する。
- (j) 拡張によって認められた新しい記号は、曖昧でなく、しかも点字と墨字に同じ一般的表現形式を容認するものであることを保証すること。
- (k) 基本的なコードの記号は、点字と墨字が対応する形にするため以外には変更されないことを保証すること。

1993年7月、このリストに、数学、コンピュータ・プログラミング、その他専門諸分野へのコード拡張を実行する課題が付け加えられた。

2.3 任務の解釈

前節ガイドラインは、個別に絶対的基準として理解されるのではなく、全体として理解されるべきであることは、最初から明白であった。そうでなければ定められた課題は互いに矛盾し調整不可能なものとなったことであろう。例えば、現行の英語点字(アメリカ式用法)では文字グループ(“gg”)と括弧は直接に文字で囲まれている場合には区別が付かないので、曖昧さを取り除くためには、略字システムか基本的な記号の用法の変更が明らかに必要であった。特にガイドライン(b)は、新しい略字を追加したり、あるいは現存している189の略字の定義を変更することを禁止しおり、十分な理由があれば、使用規則に変更を加えたり、いくつかの略字を廃止したりすることを妨げるものではない。さらに、正確さとコンピュータ対応性が我々の任務の核心をなしているが、それが墨字への隷属を意味するとは考えていない。現在実施されているものと同じく、本質的な情報は点字に訳し、単なる装飾に過ぎない墨字の書式やスタイルは無視する。例えばあるレベルの見出しが、全て下線が引かれ太字で装飾的に書かれていても、点訳に際してはそれを無視し、大文字、スペースなど何らかの情報を伝えるべきものは残す。その上で、そのレベルの見出しに対応する点字書式の規則に従って点訳するということである。

2.4 作業基盤

1993年7月の委員会任務と委員数拡大以降、委員会作業の一般的ベースは、1992年11月に出された最初の委員会報告が出発点である。しかし、報告で提出された全ての提言は必要に応じ変更可能とされ、実際に幾つかの変更が加えられた。

2.5 その他一般原則

委員会は、現在の英語点字の読者のためのコードを創作するにつき、そのシステムの伝統に敬意を払いつつ、国際的に認められている点字規則と、英語と姉妹関係にある言語の点字規則に合致しない記号割付や規則を避けるように努めてきた。また、記憶の手がかりとして英語に過度に依存している記号は定義しないよう努力されてきた。例えば、英単語の知識の有無に依存するような点字記号“s”を、四角形(square)に使うことは避けられている。

2.6 明示規則

検討された規則として、現在の英語点字の規則は、主として墨字がどのように点字に変換されるかという形で述べられている。つまり現在の規則は、点訳者またはコンピュータ・プログラムが墨字から点字に向かう方向で対処するように作られている。これは情報の伝達がほとんどその方向で行われていることを考えれば、無理もないことである。しかしながら、当委員会は当初から点字の規則としての三つの形態を考えてきた。そのうち我々の目的から考えるとまず読みの規則とデザインの規則から点訳の規則が引き出されるのが最善であって、この逆はありえないと考える。

- (1) 読みの規則
- (2) 点訳の規則
- (3) デザインの規則

上記3つの規則に関し、以下に詳しく述べる。

2.6.1 読みの規則

委員会は、点字コードは、まず読み手の立場になって考えられるべきであるとされていた。読み手はどんな記号が、与えられた点字テキストに表現されているかを正確に理解しなければならない。この視点から、初めから点字で文書を書く人のことを考慮しても、変わることはない。即ち記号を理解するだけでなく、正確に管理できなければならない。普通の文章、特に教育関係や専門的な文脈においては、対応する墨字がどのように綴られ、また句読点をつけるかという重要な点を、正確に決定する能力が要求される。技術分野では正確な記号の表現はより一層重要となる。これは文献を有効に利用するためには、不可欠のものである。そのような配慮から引き出される結論は、UBCは実行上可能なかき力、大雑把な言い方で点字から墨字という方向で曖昧さをなくすべきである。このことは人が行なうにせよ、コンピュータが処理するにせよ、必ずしも実際の変換作業のことを指すのではなく、点字から墨字記号に移る際の正確度について論じている。実際、墨字にしても点字にしても、並行的にまた同一の方法で表現されれば、いずれも抽象的記号の提示をするシステムであることに変わりがない。点字で抽象的記号や連続性と関連性が、少なくとも墨字と同様の正確さで表現できるかという点が、重要な関心事である。逆の方向の変換の場合には、ある墨字の文書に対して点字への変換の方法がただ一つというのがもちろん望ましいが、一つだけでなくそれほど重要な問題にはならない。例えば、墨字のABCは点字では以下の2通りの書き方がある。

1. 大文字符 大文字符 A B C
2. 大文字符 A 大文字符 B 大文字符 C

その時に応じてどちらの方法が適切であるかということはあるが、それは点訳規則の問題になる。しかしながら、どの点字表記に対しても、ただ一つの記号列で正確に確定されることが、読みの規則にとっては最も大事なことなのである。

2.6.2 点訳の規則

上記のように書いても、現在大部分の点字を作り出している点訳の規則の重要性を、軽視するものではない。点字コードの定義は、読みの規則の立場から行われるというのが基本である。その観点から、最も基本的かつ重要な点訳規則とは、純粋に装飾的な部分は除き、読んだときに墨字原本と全く同じになるものを作ることにある。

2.6.3 デザインの規則

UBCが長く方向性を保ち首尾一貫したものにするため、コードの多様な側面をデザイン行なう。すなわち読みの規則と点訳の規則をまとめ、コード自体がその規則に従うことが必要である。そのようなデザインの規則を引き出すことが、非常に重要なことである。それゆえ、まず主にデザインの規則と、出発点として基本的であると思われる幾つかの読みの規則と点訳の規則からUBCは始まっている。

3 UJBC

3.1 日本版 UBC

日本点字委員会は、日本点字表記法(1990年版)の発行を終えた後、点字科学記号専門委員会を発足させ、点字数学記号、点字理科記号、情報処理用点字記号の体系的統一を目指し検討を始めた。点字科学記号専門委員会では、現行表記の問題点を洗い出し、既存記号間の統一と共に、墨字の諸記号と点字記号との相互変換が可能な記号体系を確立することを目標としていた。その後、墨字の数学や科学記号の確認を行うと共に、科学技術文書用組版ソフトとして知られる \LaTeX や、アメリカのネメスコード体系をも検討対象とした。また同時期に、北米点字委員会によるUBC(統一英語点字コード)制定の動きが分かり、第2委員会の報告書の内容が極めて卓越していたため、UBCを点字科学記号の検討対象とした。一方、UBCは英語圏全体の課題へと発展し、

1992 年 10 月の北米点字委員会の報告を受け、ICEB が統一コード拡張委員会を設け、イギリス、オーストラリア、ニュージーランド、南アフリカの委員を加えて検討し、1995 年 3 月に、その最終報告書がまとめあげられた。

3.2 開発ソフトウェア

本研究を始めるにあたり、下調的な調査を数年前から行なってきた。また本格的な調査の開始と共に、UJBC に関する最初の成果が、本研究を申請した年に行われた以下のものである。

- 野辺, 藤山, 佐藤, 原, 大武: UJBC 文書の墨字翻訳システム
電子情報通信学会, 信学技報 ET97-81, pp.1-8, 1997.

上記研究は本研究の申請年度年に行われたものであるため、ここには掲載しないが、本研究の成果として UJBC に関するものを以下 (第 4 章) に掲載する。

UBC は、点字を読む者と点字を書く者の双方に、使い勝手の良い点字を提供することを、UBC 定義の基本としている。従って、UJBC も同じ視点に立っているため、読み手と書き手の双方が使えるものでなくてはならない。日本語文書の平易なものは、容易に UJBC の良し悪しの判断はつくが、今回提示された数式と化学式の定義に関する部分は、簡単に定義の矛盾や曖昧性を見出すことはできない。本来であれば、UBC で提示された Grade II, 数式、化学式の全てを網羅すべきであるが、本研究メンバーおよび関連メンバーの中に、化学点字に詳しい者がいないため、本研究の範囲は化学式を除く、Grade II と数式を対象とした。

特に数式に関しては、高等教育 (大学教育) 以上で使用される数式までを範疇とし、全盲学生にも UJBC に関する意見を求めた。しかしながら、新規に定義された UBC の数式点字をマスターすることは容易でないため、全盲学生には、数式を含む文書を $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ で記述して頂き、作成された文書を UJBC に変換する点訳ソフトと、UJBC 文書を $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ に逆変換する墨訳ソフトの双方を利用した。全盲学生による $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文書に際しては、音声ガイドを必要とするため、そのチェック・システムも、本研究で行われた。なおその詳細に関しては、以下の論文に詳細を載せた (第 4 章)。

3.3 関連研究

著作権法により、著作者・版權者に無断で複製物を作成することは、違法行為であることは熟知されている。ただし著作権法 37 条では、点字による複製が認められている。ただし、ここで言う点字は凹凸のある形になったもののみを指すということが、文化庁著作権課から明示されている。したがって、点字データとして保存されている電子メディア媒体に保存されているものは、たとえ暗号化されたものであっても違反であることも口頭で注意されている。しかし現在、ボランティアに代表されるパソコン点訳は、電子メディア媒体にデータを保存し、必要な時に点字文書として印刷 (打ち出し) するもので、点字に関する著作権法 37 条は、時代の要請にそぐわないものになってきた。これら時代の変化に伴い、1999 年 12 月に著作権審議会から、2001 年 1 月をめぐりに、点字データの電子媒体の配布を認める法改正に関わる答申が出された。

2001 年 1 月から、点字データの電子媒体の配布が認められれば、現在問題となっている点字データの電子配信に関する問題点が解消するため、インターネットを利用した点字著作物も配布が容易になる。本研究の申請段階では、点字に関する著作権法 37 条の改正は予期していなかったため、計画には入れられなかったが、1 年後の改正が望まれることから、本研究の関連として、点字データの配信に関する関連研究をまとめた。点字データの電子配信にあたる詳細は、以下の掲載論文 (第 4 章) の最後にある XML を用いた研究がこれにあたる。

4 成果論文

この部分は以下の雑誌掲載論文で構成されています。

Title<JP>	UJBC 文書の墨字翻訳システム II
Title<EN>	Automatic translation system from UJBC into printed characters II
Author(s)	曾根佐枝子 佐藤浩史 原俊介 大武信之
Citation	信学技報 ET Vol.98 No.496 pp.1-8 1998 年 12 月
Abstract<JP>	<p>現在、点訳ソフトウェアがいくつか開発されており、墨字から点字への変換は以前に比べ容易になっている。これに対し、点字から墨字へ変換するソフトウェアはほとんどなく、視覚障害者が情報発信をする上で大きな障害となっている。我々は以前に統一日本語点字記号 (UJBC) の規則に基づき、点字文書を墨字に翻訳するシステムを開発し、数式処理部では点字の数式表現を LATEX コマンドに変換する基本的な部分を完成させた。本研究では前研究を発展させ、UJBC で定義された全ての数式を処理をできるようにし、より多くの人が利用できる墨字翻訳システムを作成した。</p>
Abstract<EN>	<p>Using braille translation softwares, printed characters can be translated into braille. On the other hand, there is no softwares to transform braille into printed characters. This causes serious difficulty for the visually impaired to send information to the sighted. Previously, we developed a system which translates UJBC (Unified Japanese Braille Code) documents into printed characters automatically. That system possessed a sub-system which transforms braille mathematical expressions into the LATEX commands. That sub-system, however, could process only the basic LATEX command. We have improved that sub-system in order to correspond all the mathematical expressions that are defined in the UJBC.</p>
Rights	©社団法人電子情報通信学会
License	10GB0037

CiNii(国立情報学研究所 論文データベース・サービス)では論文の全文または一部が公開されています。詳しくは下記のリンク先をご覧ください。
<http://ci.nii.ac.jp/naid/110003192475>



この部分は以下の雑誌掲載論文で構成されています。

Title<JP>	墨字点字双方向変換統合システムの開発		
Title<EN>	Development of an Integrated System for the Translation between Printed Characters and Braille.		
Author(s)	十川奈美子 佐藤浩史 原俊介 大武信之		
Citation	信学技報 ET Vol.98 No. 496 pp.9-16 1998 年 12 月		
Abstract<JP>	<p>これまでに我々は、統一日本語点字記号に基づいた点訳及び翻訳システムを開発してきた。しかし、それらのシステムはそれぞれ特有の利用目的に応じて個々に開発されているため、点字へ変換する場合、複数のシステムを併せて利用する必要がある。その際、システム間のデータの受け渡しや、各システムの仕様に合わせるための手作業が必要となり、使用者にとっては利用しにくい点がある。本研究では、これまでのシステムを見直し、墨字から点字へ、点字から墨字への双方向変換の自動化を可能にし、視覚障害者を含めた利用者がスムーズにそれらのシステムを使用できるための制御・統合システムを開発した。</p>		
Abstract<EN>	<p>We have developed several softwares which translate Japanese, English and LATEX texts into Unified Japanese Braille Code(UJBC). And also a software which translate UJBC into printed characters. These system are, however, developed independently for the specific input text. Therefore, it causes a lot of inconvenience for users, when they try to combine the systems to translate a mixed text which is composed of Japanese, English and LATEX texts into UJBC. In this research, we have developed an integrated system which translates the mixed texts into UJBC and also UJBC into printed characters.</p>		
Rights	©社団法人電子情報通信学会	License	10GB0038

CiNii(国立情報学研究所 論文データベース・サービス)では論文の全文または一部が公開されています。詳しくは下記のリンク先をご覧ください。
<http://ci.nii.ac.jp/naid/110003192476>



この部分は以下の雑誌掲載論文で構成されています。

Title<JP>	視覚障害者用 LATEX 文書作成システム		
Title<EN>	A processing system of the LATEX documents for the visually impaired		
Author(s)	岡田久美 楠佳奈子 佐藤浩史 原俊介 大武信之		
Citation	信学技報 ET Vol.98 No. 496 pp.17-24 1998 年 12 月		
Abstract<JP>	<p>現在、点訳ソフト、音声ソフト等が多数開発され、視覚障害者は以前に比べると幅広い情報を収集できるようになった。しかし、視覚障害者の情報発信、特に数式等を含む文書を書く場合にはまだ多くの問題が存在する。我々は視覚障害者自身が数式を含む文書を作成できる「視覚障害者用 LATEX 文書作成システム」を開発した。このシステムは、視覚障害者の作成した LATEX 文書の誤りおよび完成した文書を音声で読み上げて、視覚障害者自身が容易に修正できる機能を持つ。</p>		
Abstract<EN>	<p>At present, with the help of braille-translation and reading-out systems, the visually impaired are able to get various informations. On the other hand, there exist lots of difficulties for them to produce manuscript, especially when it includes mathematical expressions. We have developed "a processing system of the LATEX documents for the visually impaired" which help them to write the LATEX text. This system has functions to announce LATEX errors by synthetic voice so that they can correct errors easily. The completed text can also be read out by this system.</p>		
Rights	©社団法人電子情報通信学会	License	10GB0039

CiNii(国立情報学研究所 論文データベース・サービス)では論文の全文または一部が公開されています。詳しくは下記のリンク先をご覧ください。

<http://ci.nii.ac.jp/naid/110003192477>



A check system of Japanese LATEX document for the blind

Copyright	IOS press
Source	Advanced Research in Computers and Communications in Education Vol.2, Nov.1999

A check system of Japanese \LaTeX document for the blind

Kanako Kusunoki*, Hiroshi Sato*,
Kaori Hotate**, Kazue Yamano**, Kumi Okada**,
Nobuyuki Ohtake[†] and Shunsuke Hara[†]

**Graduate School of Humanities and Sciences, Ochanomizu University,
2-1-1 Ohtsuka, Bunkyo-ku, Tokyo 112-8610, Japan
Tel 03-5978-5398 kusu@ns.is.ocha.ac.jp sato@is.ocha.ac.jp*

***Department of Information Science, Ochanomizu University,
2-1-1 Ohtsuka, Bunkyo-ku, Tokyo 112-8610, Japan*

*[†]Tsukuba College of Technology, 4-12 Kasuga, Tsukuba-si, Ibaraki 305-0821, Japan
Tel 0298-58-9525 ohtake@k.tsukuba-tech.ac.jp hara@k.tsukuba-tech.ac.jp*

\LaTeX is a very powerful tool to typeset scientific articles. Using \LaTeX , the blind students can write scientific articles including complex mathematical formulae. Usage of \LaTeX is, however, not very easy for them. Therefore, in this research, a system is developed which helps the blind to make \LaTeX documents. This system checks \LaTeX commands of input file both in text and mathematical environments. In case errors exist, the cause of errors is informed by synthetic voice software on sound device and the system prompts the user to modify the commands in error. When the check and analysis of \LaTeX commands are finished, the system produces an output file for reading. The user can confirm the contents of completed text and mathematical formulae by listening to the output file. By employing this system, the blind can accomplish \LaTeX document containing mathematical expressions without assist of the sighted. Programme of the system is coded using C and compiled by Microsoft 32-bit C/C++ Standard Compiler Version 10.00. Macro commands for the sound device are coded with TX-C.

Keywords: Computer Aided Learning System, the Blind, \LaTeX , Mathematical Formula, Reading

1 Introduction

To date, not a few systems have been developed which transform electronic documents into Braille or spoken output. However, the applicability of these systems is rather limited to the literal documents. For the blind students learning science and engineering at university, shortage of Braille text books is one of the serious problems. The main reason for this shortage is due to the complexity of Braille rules in mathematical formulae(math formulae). Not enough people can help to produce scientific texts in Braille.

To get over these situations, we have developed a system MathBraille[1] which transforms J \LaTeX (extended \LaTeX [2] that contains fonts of Chinese characters and Japanese syllabaries for typesetting) documents into Braille. A reading system of \LaTeX documents; MathRead[3] is also constructed, which produces output file for reading from

input \LaTeX documents. These systems are helping the blind students studying science.

Another problem for the blind students is difficulty in presenting their thoughts to the sighted. For example, to write scientific reports including math formulae in ink print is not an easy task for the blind, even he/she has a good understanding of the subject.

\LaTeX is a very powerful tool to typeset scientific articles. Using \LaTeX , the blind students can write scientific articles including complex math formulae. Usage of \LaTeX is, however, not very easy for them. When errors exist in their \LaTeX documents, they have to check log file with synthetic sound device. Log file usually contains too much information to listen and they are given in English. Most of Japanese synthesized voice software on sound device read alphabets only character by character. Furthermore, there is no way to confirm the math formulae they have written except to listen sequence of raw \LaTeX commands in alphabets.

In this paper, we discuss a system MathCheck[4], which is developed to help the blind to make \LaTeX document including math formulae. This system checks \LaTeX commands of input file. In case errors exist, the cause of errors is informed by synthetic sound device. When the check and analysis of \LaTeX commands are finished, MathCheck produces an output file for reading, where a sequence of \LaTeX commands is replaced to a sentence. The user can confirm the contents of completed text and math formulae by listening to the output file.

2 Outline of the system

In MathCheck, one character in the input file is read and written down to the file for reading unless it is one of the special characters defined in \LaTeX . If it is the first character for \LaTeX commands in text mode, the string of the command is read and processed. When the character or the string is one of the delimiters for math formula(\$, \$\$ etc), the whole sequence of the formula is read and stored in an array for analysis. So long as the command read is analyzable, the system replaces \LaTeX command to a set of words through a translation table and write it in an output file. The method of analysis is the same as that developed in MathBraille[1].

Programme is coded using the programming language C and compiled by Microsoft 32-bit C/C++ Standard Compiler Version 10.00. MathCheck works on personal computers(PCs) operated by Windows95/98.

2.1 Sound device

Recent multimedia PCs are equipped with audio device by default. Several Japanese text-to-speech softwares are developed which can be used in the PCs operated by Windows95/98. Examples are 95Reader[5] and ProTALKER 97[6]. They can speak very audible Japanese without external devices. Since the sound interfaces are open to the public, these softwares can be used in many application softwares including text editors. In this system, WZ editor[7] and 95Reader are adopted. Macro commands for the sound device are coded by TX-C[8].

2.2 Work flow

Fig.1 shows a flowchart of MathCheck system. Utilizing this system, users can complete \LaTeX text free from errors as follows:

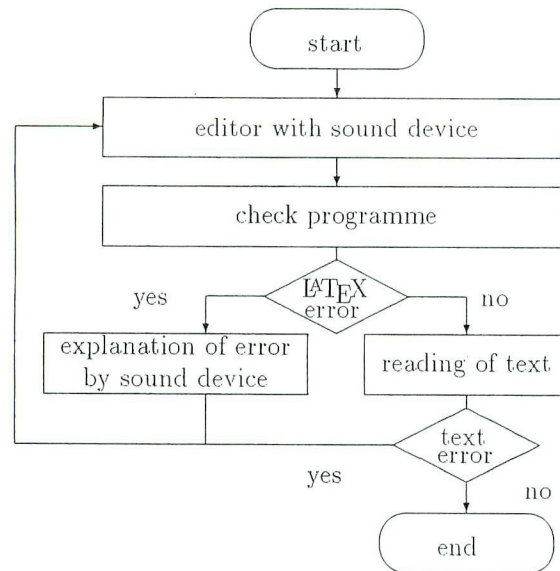


Figure 1: System flowchart.

- step (1) Users prepare a \LaTeX source file in WZ editor with the help of synthetic sound device.
- step (2) Users check the source file by inputting it into the check programme.
- step (3) In case \LaTeX errors exist in the source file, the line number where the errors exist and the cause of errors are informed by synthetic sound device. When the users need to reconfirm the errors, the error messages can be repeated. Furthermore, all the error messages are saved in a history file.
- step (4) After confirming the error messages, users go back to step (1) to correct errors. In the editor, users move the cursor to the lines where the errors exist and amend the errors. Steps (1)-(4) are repeated until no \LaTeX error exists in the source file. It is possible, if desired, to ignore error message and proceed analysis.
- step (5) When no \LaTeX error exists in the input file, the check programme produces an output file for reading. Users can confirm the contents of completed text and math formulae by listening to this file. When faults are found in the text, users go back to step (1) to modify them.

Processing the file thus checked by \LaTeX , the blind is able to produce \LaTeX document in printed characters without help of the sighted.

3 Reading of math formula

In many European languages, the order of mathematical symbols in math formula is, in general, in consistent with the order of reading. Whereas, the composition of Japanese

is quite different from European languages, and therefore, in Japanese, the order of mathematical symbols is not in accord with that of reading. Besides, rules for reading math formula are not well defined in Japan. Furthermore, there are lots of ambiguities in reading math formula which are given in two dimensional form. For example, in Japanese, equations

$$a + \frac{c}{b} + d \quad \text{and} \quad \frac{c + d}{a + b}$$

are read in the same way ($a + b$ *bun-no* $c + d$). Nevertheless, we have to establish rules for reading math formula without ambiguity, even the way of reading is not very common to Japanese. We have tentatively adopted following rules for reading math formula[3]:

- Math formula is read, not as a part of Japanese text but as a sequence of mathematical symbols. They are read, in many cases, from left to right.
- The way of reading for one mathematical symbol which may corresponds to more than two \LaTeX commands is, in principle, fixed to one.
- To avoid ambiguity in reading math formula, extra "*hajime*(start)" and "*owari*(end)" are added if necessary.

By these rules, all the math formulae are read out without ambiguity.

4 Processing of \LaTeX text

In the construction of \LaTeX document, it is recommended for the blind to utilize a prepared master file(root file) which includes style file and preambles. When the master file is ready, the blind user just need to write document files separately and input them to the master file. Our system analyzes \LaTeX text within the "document environment", that is, the contexts between the lines `\begin{document}` and `\end{document}`. Therefore, master file is necessary to process the completed input file by \LaTeX and to typeset articles. Definition of customized commands(`\newcommand`) must be also in the master file. For the customized commands, MathCheck gives error message in the course of analysis. As has been mentioned earlier, this message can be ignored.

The \LaTeX commands may be classified into two categories. One is the commands in text mode and the other, that in math mode.

4.1 Commands in text mode

Commands in text mode are processed one by one. Input command is converted to output data through the translation table for reading. New line commands such as `\newline` and `\\` are replaced to a long pause. When the command does not exist in the table, error message and line number where the command exists are announced by synthetic sound device. The message can be repeated if necessary. Examples are given in Table 1 . The last row is an example of error. The command `\alpha` is usable only in math mode.

input data	L ^A T _E X output	Japanese reading	remarks
<code>\^a</code>	\hat{a}	⁽¹⁾ <i>hatto a</i>	European accents in text mode.
<code>{\bf abc}</code>	abc	⁽²⁾ <i>futomoji hajime abc futo-moji owari</i>	When argument is more than 2 characters, "hajime(start)" and "owari(end)" are attached.
<code>\alpha</code>		⁽³⁾ <i>komando \alpha wa bunchū yō no hyō ni arimasen</i>	Error message. Not written to the output file

⁽¹⁾hat a

⁽²⁾bold start abc end of bold

⁽³⁾(line xx:) command `\alpha` does not exist in the table for text mode.

Table 1: Examples of L^AT_EX command in text mode.

4.2 Commands in math mode

In L^AT_EX, inline math formulae are delimited by `$`, and displayed math formulae by `$$`(or by `\(`, `\)` and `\[`, `\]`, respectively). These two math formulae are not distinguished in reading.

In MathCheck, commands in math mode are classified into 5 categories; subscript, superscript, commands with arguments 0, 1 and 2. Most of the commands are converted into words for reading through the translation table. Exceptions are subscript, superscript, large operators, `\frac` and several other commands. They are processed in the independent functions. For the errors in math formulae, the line number where the math formula begins is given.

There are math formulae which may be incorrect as mathematical descriptions, though they are written correctly in L^AT_EX format and no error message is given when processed by L^AT_EX. For such cases, the system gives warning messages.

4.2.1 Subscript and superscript

Subscript is usually read as "shita-tsuki"(subscript) and superscript as "ue-tsuki"(superscript). Therefore, for example, x^2 is read as "x ue-tsuki 2" instead of "x ji-jō(the second power of 2)". Exceptions are those attached to the large operators

input data	L ^A T _E X output	Japanese reading	remarks
<code>\$\$\int_a^b f(x)dx\$\$</code>	$\int_a^b f(x)dx$	⁽¹⁾ \int a kara b made f(x)dx	"_a^b" is read as "a kara b made" in this case.
<code>\$\$\lim_{n \rightarrow \infty} a_n\$\$</code>	$\lim_{n \rightarrow \infty} a_n$	⁽²⁾ <i>rimitto n tsū ∞ a shita-tsuki n</i>	"_" attached to <code>\lim</code> is not read. <code>\to</code> is read as "tsū" in this case.
<code>\$x^{a^b}\$</code>		⁽³⁾ <i>ni-jū ue-tsuki fu erā</i>	Error message. Not written to the output file.

⁽¹⁾integral from a to b f(x)dx

⁽²⁾limit n to infinity a subscript n

⁽³⁾(math mode in line xx:) double superscript error.

Table 2: Examples of L^AT_EX command in math mode I.

such as `\int` and `\sum`. Examples are given in Table 2 . The last row in this table corresponds to an example of error.

4.2.2 Commands with arguments(0-2)

In Table 3, examples of commands with arguments(0-2) are given together with their readings. Numbers in the first column correspond to the number of arguments. Commands with one argument are divided into two groups according to the way of reading. The 7-th row is an example of error and the last row, that of warning.

	input data	ℒ _T EX output	Japanese reading	remarks
0	<code>\$a \times b\$</code>	$a \times b$	(1) <i>a kakeru b</i>	
1	<code>\$\bar{a}\$</code>	\bar{a}	(2) <i>a bā</i>	Argument is read first.
1	<code>\$\sqrt{a}\$</code>	\sqrt{a}	(3) <i>rūto a</i>	Command is read first.
1	<code>\$\sqrt{a+b}\$</code>	$\sqrt{a+b}$	(4) <i>rūto hajime a+b rūto owari</i>	When argument is more than 2 characters, " <i>hajime(start)</i> " and " <i>owari(end)</i> " are attached.
2	<code>\$a+\frac{b}{c}+d\$</code>	$a + \frac{b}{c} + d$	(5) <i>a+b ōbā c+d</i>	/ is read as " <i>ōbā(over)</i> ", when both arguments are one character. Also for <i>df/dx</i> and <i>∂f/∂x</i> etc.
2	<code>\$\frac{a+b}{c+d}\$</code>	$\frac{a+b}{c+d}$	(6) <i>bunsū hajime a+b surasshu c+d bunsū owari</i>	/ is read as " <i>surasshu(slash)</i> ", when arguments in fraction are more than 2 characters.
0	<code>\$\lamda_0\$</code>		(7) <i>sūshiki chū no ko- mando \lamda wa hyō ni arimasen</i>	Error message. Not written to the output file.
0	<code>\$(a+\beta^2\$</code>	$(a + \beta^2$	(8) <i>shō kakko no kazu ga icchi shimasen</i>	Warning. Not written to the output file.

(1) *a times b.*

(2) *a bar.*

(3) *root a.*

(4) *root start a+b end of root.*

(5) *a+b over c + d.*

(6) *fraction start a+b slash c+d end of fraction.*

(7) (math mode in line xx:) command `\lamda` does not exist in the translation table.

(8) (math mode in line xx:) number of parentheses does not match.

Table 3: Examples of ℒ_TEX command in math mode II.

5 Conclusions

We have developed a system MathCheck which helps the blind to prepare ℒ_TEX document.

This system checks ℒ_TEX commands both in text and math modes. When the ℒ_TEX errors exist in the document, the line number where the errors exist and the cause of errors are given by the synthetic voice. By these error messages, the blind users can modify the ℒ_TEX commands and complete ℒ_TEX document free from errors.

We have established rules to read out math formulae unambiguously in Japanese. Following these rules, our system produces an output file for reading. The users can confirm the contents of the completed document by listening to the output file. Thus the blind can construct a correct \LaTeX document unaided. This system will be very useful to the blind students studying science.

References

- [1] N.Miyazaki, M.Higuchi, H.Sato, S.Hara and N.Ohtake, "Automatic braille transformation system of mathematical expressions "(in Japanese except abstract), Technical Report of the Institute of Electronics, Information and Communication Engineers(IEICE), vol. 94, no. 480, pp. 9–16 (1995)
- S.Hara, N.Ohtake, M.Higuchi, N.Miyazaki, M.Higuchi, K.Kusunoki and H.Sato, "MathBraille; a system to transform \LaTeX document into Braille", submitted to ACM SIGCAPH Newsletters.
- [2] L. Lamport, " \LaTeX ; A Document Preparation System", Addison-Wesley, Reading, Mas. (1986)
- [3] K.Hotate, K.Yamano, H.Sato, S.Hara and N.Ohtake, "Reading system of mathematical expressions for the visually impaired"(in Japanese except abstract), Technical Report of the Institute of Electronics, Information and Communication Engineers(IEICE), vol. 97, no. 464, pp. 17–24 (1997)
- [4] K.Okada, K.Kusunoki, H.Sato, S.Hara and N.Ohtake, "A processing system of the \LaTeX documents for the visually impaired"(in Japanese except abstract), Technical Report of the Institute of Electronics, Information and Communication Engineers(IEICE), vol. 98, no. 496, pp. 17–24 (1998)
- K.Kusunoki, K.Okada, H.Sato, S.Hara and N.Ohtake, "A \LaTeX documents processing system for Windows adapted to the visually impaired"(in Japanese), Abstracts of contributed papers No.4, the 58th Annual Conference, Information Processing Society of Japan, pp. 271–272 (1999)
- [5] T.Watanabe, S.Okada and T.Ifukube, "Development of a GUI Screen Reader for Blind Persons", Systems and Computers in Japan, vol. 29, no.13, pp. 18–27 (1998)
- [6] "ProTALKER 97 Version 2.0 Manual", IBM Corp. (1997)
- [7] "WZ EDITOR Version 3.0 Manual", VILLAGE CENTER INC. (1997)
- [8] Y.Mikone, "WZ Ver.2.0 Macro Manual"(in Japanese), VILLAGE CENTER INC. (1996)

この部分は以下の雑誌掲載論文で構成されています。

Title<JP>	音声を用いた視覚障害者用科学技術文書作成システム		
Title<EN>	A processing system of scientific documents for the blind using sound device		
Author(s)	楠佳奈子 佐藤浩史 原俊介 大武信之		
Citation	信学技報 ET Vol.99 No. 500 pp.97-104 1999 年 12 月		
Abstract<JP>	<p>情報処理機器の発達により、視覚障害者の情報収集は比較的容易になってきたが、情報発信、特に数式等を含む科学技術文書の作成にはまだ困難が伴い、晴眼者の助力が必要になることが多い。我々は視覚障害者自身で数式を含む科学技術文書の作成を可能とするため、Windows 上で動作する LATEX 文書作成支援システムを開発した。このシステムは Windows 画面読み上げソフトウェアを用いて音声出力を行い、必要があれば点字ディスプレイへの出力も可能である。このシステムを用いることで視覚障害者は LATEX 文書の修正および作成した文書の確認が容易に行える。</p>		
Abstract<EN>	<p>With the advancement of computer technology, giving and receiving of various information are gradually becoming easy for the blind people. Nevertheless, there exist lots of difficulties for them to present a scientific document, especially when it includes mathematical expressions. LATEX is a system used commonly to produce scientific documents. Therefore we have developed "a processing system of scientific documents in LATEX format" which helps the blind to write the scientific text. This system is supported by sound device, and also gives Braille output through Braille display equipment. Using this system, the blind can correct LATEX errors and confirm the prepared document easily.</p>		
Rights	©社団法人電子情報通信学会	License	10GB0040

CiNii(国立情報学研究所 論文データベース・サービス)では論文の全文または一部が公開されています。詳しくは下記のリンク先をご覧ください。
<http://ci.nii.ac.jp/naid/110003226844>



この部分は以下の雑誌掲載論文で構成されています。

Title<JP>	XML を用いた点字教材とその応用		
Title<EN>	The Braille texts in XML and their applications		
Author(s)	宇田川明子 新村由紀子 川満良子 楠佳奈子 渡部亜矢子 佐藤浩史 原俊介 大武信之		
Citation	信学技報 ET Vol.99 No. 500 pp.143-150 1999 年 12 月		
Abstract<JP>	<p>これまでに、我々は統一日本語点字記号に基づいた点訳及び翻訳システムを開発してきた。しかし、墨字文書と点字文書が個々のファイルに納められていた。各文書が分離して存在していたことにより、視覚障害者が原文の墨字文書を参照することは困難である。そこで、我々は墨字ファイルと点字ファイルを統合した XML 文書を作成した。作成された統合点字文書は、墨字情報も納められているため、晴眼者にも利用できる。このことはこの文書の電子図書としての利用を考える上で大きな意味をなす。</p>		
Abstract<EN>	<p>So far, We have developed several softwares which translate . Japanese, English and LATEX texts into Unified Japanese Braille Code. In these systems, the file for printed characters and that for North American Braille Computer Code have existed separately. Therefore, it has been difficult for the visually impaired to refer to the original files in printed characters. In this research, we have made XML document integrated those two files. The Integrated documents have informations of printed characters, and the sighted people can also use them. This has an important meaning that the files can be used as the electronic library.</p>		
Rights	©社団法人電子情報通信学会	License	10GB0041

CiNii(国立情報学研究所 論文データベース・サービス)では論文の全文
または一部が公開されています。詳しくは下記のリンク先をご覧ください。
<http://ci.nii.ac.jp/naid/110003226850>



Part II

日本語英語自動点訳

5 研究目的

日本語の点訳を行なうには日本点字規則を習得すると共に、ある程度の経験も必要とされる。従って、全く点字を知らない者にとっては、点訳作業は容易なものではない。たとえ点訳の熟練者にとっても、短期間に大量の点訳依頼にはこたえることができない。従って、点字に対する習熟度の違いがあっても、日本語墨字を自動的に点訳するソフトウェアが存在すれば、様々な階層の点字利用者やボランティアが、点訳作業の軽減に役立たせることができる。

これまでボランティア等により行われてパソコン点訳は、おもにMS-DOS環境におけるものであった。以前の情報機器は、高価なものであり、また専門家にしか扱えないものであったが、安価なものになるにつれ一般社会への浸透と共に、点字編集・印刷等を目的とした強力な道具となった。これまでのMS-DOS環境における作業は、コマンド入力によるアプリケーション・ソフトウェアの利用であったが、GUI (Graphical User Interface) を重視したWindows環境の基本ソフトウェアがパソコンの主流となり、本研究ではWindows95/98/NT環境で動作する、日本語英語自動点訳システムの開発を目指した。

6 基本操作

この章では初めての使用者に、比較的使用頻度が高いと思われる変換手順を、以下を対象に、例題形式で順に解説する。

- 「点字編集ができる」
- 「点字編集はできないが、かな編集ならできる」
- 「原文が読めない、あるいは大至急点訳したい」

6.1 対象：点字編集ができる

事前準備

元ファイルの末尾が「sample1.txt」のように、「(改行無しの) テキストのみ」の「.txt(テキスト形式ファイル)」かどうか確認する。

注：他にはHTMLファイル（末尾が「.htm」「.html」）も扱える。

注：Wordファイル“拡張子「.doc」”または一太郎ファイル“拡張子「.jtd」”等、上記の形式のファイル以外を点訳したい場合は、それぞれのワープロソフトを使い、「名前をつけて保存」の画面で「ファイルの種類」にテキストを指定してから「保存」をクリックし、元のファイルをテキストファイルとして保存し直す。

例) test.doc の場合、以下のようになる。

元ファイルのファイル名 test、拡張子 .doc、新しく作ったテキスト形式ファイルのファイル名 test、拡張子 .txt。

漢字かな英数の文書ファイルを、点字印刷する。

手順1) 中間ファイルとして、点字ファイルを作成する。→ 例題 1-1

手順2) その中間ファイルを点字印刷する。→ 例題 1-2

<例題 1-1> 漢字かな英数の文書ファイル (sample1.txt) を、BES 形式のファイルにする。

1 EXTRA for Windowsを開く。

まず、Windowsのスタートメニューを開き、「EXTRA for Windows」の上で一回クリックして、EXTRA for Windowsをスタートさせる。

2 読み込むファイルの形式を指定。→「入力形式」

右の下向き矢印をクリックするとリストが表示されるので、BESに変換したい元ファイルとして、「漢字かな英数」を選ぶ。

3 読み込むファイルがどれかを指定。→「ファイル選択」

「入力ファイル」の右にある、「ファイル選択」をクリックして「入力ファイル選択」画面を開け、読み込むファイルが存在するドライブ、フォルダー等を開き、ファイルを指定する。(ファイルを選ぶ時は目的のファイル名の上でクリックし、「ファイル名」の欄に、選んだファイルの名前が入ったか確認する。)
「開く」をクリックしてEXTRA for Windows本体画面に戻ると「ディレクトリ」には、選んだファイルが存在しているドライブおよびフォルダーが、「入力ファイル」にはファイル名が表示されているはずである。ここで「内容」をクリックすると通常、ファイルの種類に適したソフトウェアが開き、指定したファイルの内容を見たり編集したりすることができる。

注：ファイルに適したソフトウェアとは、ファイルの拡張子に対してあらかじめ関連付けされているソフトウェアのことである。これはWindows95/98の場合、エクスプローラーのオプション上で設定されている。

次は、新しくできた点字ファイル（BES形式）を中間ファイルとして保存するための設定。

4 出力結果をどういう形にするか指定。→「出力形式」

今回はBES形式の点字ファイルにしたいので、「出力形式」の行の右端下向き矢印をクリックし、リストから「BES」を選ぶ。

5 出力ファイルを指定。→「ファイル選択」

「出力ファイル」の右にある、「ファイル選択」をクリックして「出力ファイル選択」画面を開けると、入力ファイルと同じ場所が表示され、入力ファイル名を元にした名前（sample1.bes）が自動的に割り当てられる。ここで、出力ファイルの名前を新しく付けたいときや、出力ファイルの保存される場所を入力ファイルと異なる場所に変更したいときは、どこに保存するかドライブ、フォルダー等を選び、出力ファイルの名前を入力し、「開く」をクリックし確定する。その際、拡張子無しでファイル名のみを入力すると、出力形式に対応した拡張子が自動的に付加される。

例) ここでは漢字かな英数の文書ファイル（sample1.txt）を、BES形式にするのが目的なので、「出力形式」：BES、元ファイル名：sample1.txtとすると、

a) 出力ファイル名：sampleA（と入力した場合）

→ 割り当てられた出力ファイル名：sampleA.besと正しく設定される。また、拡張子を省略せずにファイル名全体を入力する場合は、拡張子を間違えないように注意する必要がある。

b) 出力ファイル名：sampleB.bes（と入力した場合）

→ 割り当てられた出力ファイル名：sampleB.bes（適切）

c) 出力ファイル名：sampleC.bse（と誤入力した場合）

→ 割り当てられた出力ファイル名：sampleC.bse.bes（不適切）

注：c)の場合、誤った拡張子に、「出力形式」での指定が追加され、不適切な名前になってしまう。そのため、通常はファイル名のみを入力を勧める。

6 点訳の種類を指定。→「点訳」

「点訳」の右にある下向き矢印をクリックするとリストが表示される。

ここでは、一般的な「日本 英2」を選ぶ。

- 7 点字にする際の、一ページあたりの行数と、一行あたりの点字の数を指定。→「行」「マス」
必要があれば、右の上下矢印をクリックして行数あるいはマス数を選ぶか、直接入力する。
(初期値は22行、32マスになっている。)
- 8 ページ番号をつける。→「ページ付け」
出力ファイルにページ番号をつけるため、「ページ付け」の左のマスにチェックマークが付いているか確認する。もしないようなら、左のマスをクリックすることでチェックマークをつける。
- 9 変換作業を行う。→「作業開始」
「作業開始」スイッチをクリックする。
- 10 変換作業の状況を確認する。→「状況」
「作業開始」スイッチの右にある「状況」が、“作業処理中”から“作業終了”になるまで待つ。
- 11 結果を確認する。→「内容」
(もしIBMの点字編集プログラム(BES)がインストールされていれば、)終了後、出力ファイル名の横の「内容」をクリックすると、点字編集プログラムが開くので、新しくできたファイルの内容を確認する。なお、BES形式ファイル(拡張子「.BES」)とBE形式ファイル(拡張子「.BET」)以外の点字ファイル形式を指定した場合や、点字編集用ソフトウェアがインストールされていない場合は、EXTRA for Windows内蔵の点字表示用ソフトウェアが開く。
- 12 新しくできたファイルを修正する。
「11」で誤変換が発見されたら修正し、その後保存する。ただし、IBMの点字編集プログラム(BES)がインストールされていない時は、別途利用の点訳ソフトウェアを使って修正する必要がある。

<例題1-2> 例題1-1で作成したBESファイル(sample1.BES)を点字印刷する。

- 1 読み込むファイルの形式を指定。→「入力形式」
右の下向き矢印をクリックするとリストが表示されるので、点字印刷したい元ファイルとして、「BES」を選ぶ。
- 2 読み込むファイルがどれかを指定。→「ファイル選択」
「入力ファイル」の右にある、「ファイル選択」をクリックして「入力ファイル選択」画面を開け、読み込むファイルが存在するドライブ、フォルダー等を開き、ファイルを指定する。(ファイルを選ぶ時は目的のファイル名の上でクリックし、「ファイル名」の欄に、選んだファイルの名前が入ったか確認する。)
「開く」をクリックしてEXTRA for Windows本体画面に戻ると「ディレクトリ」には、選んだファイルが存在しているドライブおよびフォルダーが、「入力ファイル」にはファイル名が表示されているはずである。

次は、点字印刷のための設定。今回は「例題1-1」のレイアウトをそのまま使用することにして、「レイアウト変更」「オプション変更」は行わない。

- 3 出力結果をどういう形にするか指定。→「出力形式」
点字印刷したいので、「出力形式」の行の右端下向き矢印をクリックし、「プリンタ1同時 ET (IL)」を選ぶ。
- 4 点字印刷の部数やページを指定。→「印刷部数」「開始ページ」「最終ページ」
「印刷部数」はプリンターから同一内容の出力結果をいくつほしいか、「開始ページ」「最終ページ」はプリントしたいのは何ページから何ページなのかを、右の上下矢印をクリックして数を選ぶか、直接入力する。ただし、開始ページが1ページ目でない時は、7章4の5)を参照し、「ページ初期値」を開始ページに合わせて修正する。

5 点字印刷を行う。→「作業開始」

「作業開始」スイッチをクリックする。

6 変換作業の状況を確認する。→「状況」

「作業開始」スイッチの右にある「状況」が“作業処理中”から“作業終了”になるまで待つ。

＜補足＞ 点字印刷した際の左右の空白調整をしたい場合は、7章4の3)を参照しインデント調整後、再度上記5、6を行う。

以上で、点字印刷は終了である。

6.2 対象：点字編集はできないが、かな編集ならできる

＜例題2-1＞ 漢字かな英数の文書ファイル (sample2.txt) を、ひらがな分かち書きにする。

1 EXTRA for Windowsを開く。

まず、Windowsのスタートメニューを開き、「EXTRA for Windows」の上で一回クリックして、EXTRA for Windowsをスタートさせる。

2 読み込むファイルの形式を指定。→「入力形式」

右の下向き矢印をクリックするとリストが表示されるので、ひらがな分かち書きに変換したい元ファイルとして、「漢字かな英数」を選ぶ。

3 読み込むファイルがどれかを指定。→「ファイル選択」

「入力ファイル」の右にある、「ファイル選択」をクリックして「入力ファイル選択」画面を開け、読み込むファイルが存在するドライブ、フォルダー等を開き、ファイルを指定する。(ファイルを選ぶ時は目的のファイル名の上でクリックし、「ファイル名」の欄に、選んだファイルの名前が入ったか確認する。)
「開く」をクリックしてEXTRA for Windows本体画面に戻ると「ディレクトリ」には、選んだファイルが存在しているドライブおよびフォルダーが、「入力ファイル」にはファイル名が表示されているはずである。ここで「内容」をクリックすると通常、ファイルの種類に適したソフトウェアが開き、指定したファイルの内容を見たり編集したりすることができる。

注：ファイルに適したソフトウェアとは、ファイルの拡張子に対してあらかじめ関連付けされているソフトウェアのことである。これはWindows95/98の場合、エクスプローラーのオプション上で設定されている。

次は、ひらがな分かち書きに変換されて、新しくできるファイルに関する設定。

4 出力結果をどういう形にするか指定。→「出力形式」

今回はひらがな分かち書きにしたいので、「出力形式」の行の右端下向き矢印をクリックし、リストから「ひらがな英数」を選ぶ。

5 出力結果ファイルを指定。→「ファイル選択」

「出力ファイル」の右にある、「ファイル選択」をクリックして「出力ファイル選択」画面を開けると、入力ファイルと同じ場所が表示され、入力ファイル名を元にした名前 (sample2.hra) が自動的に割り当てられる。ここで、出力ファイルの名前を新しく付けたいときや、出力ファイルの保存される場所を入力ファイルと異なる場所に変更したいときは、どこに保存するかドライブ、フォルダー等を選び、出力ファイルの名前を入力し、「開く」をクリックし確定する。その際、拡張子無しでファイル名のみを入力すると、出力形式に対応した拡張子が自動的に付加される。

例) ここでは漢字かな英数の文書ファイル (sample2.txt) を、ひらがな分かち書き形式にするのが目的なので、「出力形式」：ひらがな英数、元ファイル名：sample2.txt とすると、

- a) 出力ファイル名 : sampleA (と入力した場合)
→ 割り当てられた出力ファイル名: sampleA.hra と正しく設定される。また、拡張子を省略せずにファイル名全体を入力する場合は、拡張子を間違えないように注意する必要がある。
- b) 出力ファイル名: sampleB.hra (と入力した場合) → 割り当てられた出力ファイル名: sampleB.hra (適切)
- c) 出力ファイル名 : sampleC.kta (と誤入力した場合) → 割り当てられた出力ファイル名 : sampleC.kta.hra (不適切)

注: c) の場合、誤った拡張子に、「出力形式」での指定が追加され、不適切な名前になってしまう。そのため、通常はファイル名のみの入力を勧める。

6 変換作業を行う。→「作業開始」

「作業開始」スイッチをクリックする。

7 変換作業の状況を確認する。→「状況」

「作業開始」スイッチの右にある「状況」が“作業処理中”から“作業終了”になるまで待つ。

8 結果を確認する。→「内容」

終了後、出力ファイル名の横の「内容」をクリックすると通常、ファイルの種類に適したソフトウェアが開き、指定したファイルの内容を見たり編集したりすることができる。

注: ファイルに適したソフトウェアとは、ファイルの拡張子に対してあらかじめ関連付けがされているソフトウェアのことである。これは Windows95/98 の場合、エクスプローラーのオプション上で設定されている。

9 新しくできたファイルを修正する。

「8」で誤変換が発見されたら修正し、その後保存する。もし、「8」で開かれたソフトウェアが編集にしたいソフトウェアと違っていた上、Windows95/98のエクスプローラでの設定変更がわからなかった場合は、一旦「8」で開いたファイルを閉じる。その後、希望の編集ソフトウェアを開き、編集する。

＜例題 2-2＞ 例題 2-1 で作成したひらがな分かち書きファイル (sample2.hra) を点字印刷する。

1 読み込むファイルの形式を指定。→「入力形式」

右の下向き矢印をクリックするとリストが表示されるので、点字印刷したい元ファイルとして、「ひらがな英数」を選ぶ。

2 読み込むファイルがどれかを指定。→「ファイル選択」

「入力ファイル」の右にある、「ファイル選択」をクリックして「入力ファイル選択」画面を開け、読み込むファイルが存在するドライブ、フォルダー等を開き、ファイルを指定する。(ファイルを選ぶ時は目的のファイル名の上でクリックし、「ファイル名」の欄に、選んだファイルの名前が入ったか確認する。)
「開く」をクリックして EXTRA for Windows 本体画面に戻ると「ディレクトリ」には、選んだファイルが存在しているドライブおよびフォルダーが、「入力ファイル」にはファイル名が表示されているはずである。ここで「内容」をクリックすると通常、ファイルの種類に適したソフトウェアが開き、指定したファイルの内容を見たり編集したりすることができる。

注: ファイルに適したソフトウェアとは、ファイルの拡張子に対してあらかじめ関連付けがされているもののことである。これは Windows95/98 の場合、エクスプローラーのオプション上で設定されている。

次は、点字印刷のための設定。

- 3 出力結果をどういう形にするか指定。→「出力形式」
点字印刷したいので、「出力形式」の行の右端下向き矢印をクリックしリストから、「プリンタ1同時 ET (IL)」を選ぶ。
- 4 点訳の種類を指定。→「点訳」
「点訳」の右にある下向き矢印をクリックするとリストが表示される。ここでは、一般的な「日本 英2」を選ぶことにする。
- 5 点字にする際の、1ページあたりの行数と、1行あたりの点字の数を指定。→「行」「マス」
必要があれば、右の上下矢印をクリックして行数あるいはマス数を選ぶか、直接入力する。
(初期値は22行、32マスになっている。)
- 6 ページ番号をつける。→「ページ付け」
出力ファイルにページ番号をつけるため、「ページ付け」の左のマスにチェックマークが付いているか確認する。もしないようなら、左のマスをクリックすることでチェックマークをつける。
- 7 点字印刷の部数やページを指定。→「印刷部数」「開始ページ」「最終ページ」
「印刷部数」はプリンターから同一内容の出力結果をいくつほしいか、「開始ページ」「最終ページ」はプリントしたいのは何ページから何ページなのかを、右の上下矢印をクリックして数を選ぶか、直接入力する。ただし、開始ページが1ページ目でない時は、7章4の5)を参照し、「ページ初期値」を開始ページに合わせて修正する。
- 8 点字印刷を行う。→「作業開始」
「作業開始」スイッチをクリックする。
- 9 変換作業の状況を確認する。→「状況」
「作業開始」スイッチの右にある「状況」が“作業処理中”から“作業終了”になるまで待つ。

＜補足＞ 点字印刷した際の左右の空白調整の必要が生じた場合は、7章4の3)を参照しインデント調整後、再度上記8、9を行う。

以上で点字印刷終了。

6.3 対象：原文が読めない、あるいは大至急点訳したい

＜例題3＞ 漢字かな英数ファイル(sample3.txt)を点字印刷する。

- 1 読み込むファイルの形式を指定。→「入力形式」
右の下向き矢印をクリックするとリストが表示されるので、点字印刷したい元ファイルとして、「漢字かな英数」を選ぶ。
- 2 読み込むファイルがどれかを指定。→「ファイル選択」
「入力ファイル」の右にある、「ファイル選択」をクリックして「入力ファイル選択」画面を開け、読み込むファイルが存在するドライブ、フォルダー等を開き、ファイルを指定する。(ファイルを選ぶ時は目的のファイル名の上でクリックし、「ファイル名」の欄に、選んだファイルの名前が入ったか確認する。)
「開く」をクリックしてEXTRA for Windows本体画面に戻ると「ディレクトリ」には、選んだファイルが存在しているドライブおよびフォルダーが、「入力ファイル」にはファイル名が表示されているはずである。ここで「内容」をクリックすると、通常ファイルの種類に適したソフトウェアが開き、指定したファイルの内容を見たり編集したりすることができる。

注：ファイルに適したソフトウェアとは、ファイルの拡張子により対してあらかじめ関連付けがされているもののことである。これはWindows95/98の場合、エクスプローラーのオプション上で設定されている。

次は、点字印刷のための設定。

3 出力結果をどういう形にするか指定。→「出力形式」

点字印刷したいので、「出力形式」の行の右端下向き矢印をクリックしリストから、「プリンタ1同時 ET (IL)」を選ぶ。

4 点訳の種類を指定。→「点訳」

「点訳」の右にある下向き矢印をクリックするとリストが表示される。

ここでは、一般的な「日本 英2」を選ぶことにする。

5 点字にする際の、一ページあたりの行数と、一行あたりの点字の数を指定。→「行」「マス」

必要があれば、右の上下矢印をクリックして行数あるいはマス数を選ぶか、直接入力する。

(初期値は22行、32マスになっている。)

6 ページ番号をつける。→「ページ付け」

出力ファイルにページ番号をつけるため、「ページ付け」の左のマスにチェックマークが付いているか確認する。もしないようなら、左のマスをクリックすることでチェックマークをつける。

7 点字印刷の部数やページを指定。→「印刷部数」「開始ページ」「最終ページ」

「印刷部数」はプリンターから同一内容の出力結果をいくつほしいか、「開始ページ」「最終ページ」はプリントしたいのは何ページから何ページなのかを、右の上下矢印をクリックして数を選ぶか、直接入力する。ただし、開始ページが1ページ目でない時は、7章4の5)を参照し、「ページ初期値」を開始ページに合わせて修正する。

8 点字印刷を行う。→「作業開始」

「作業開始」スイッチをクリックする。

9 変換作業の状況を確認。→「状況」

「作業開始」スイッチの右にある「状況」が“作業処理中”から“作業終了”になるまで待つ。

<補足> 点字印刷した際の左右の空白調整の必要が生じた場合は、7章4の3)を参照しインデント調整後、再度上記8、9を行う。

以上で点字印刷終了。

7 機能解説

この章では、6章では割愛した EXTRA for Windows の使い方などを解説する。

7.1 作業1～作業5

EXTRA for Windows では、作業セットを5個用意している。最後に使用した時の設定が次回に活かされるので、それぞれの作業セットを用途に応じて使い分けられる。

注：初期設定に戻したい場合は、1～5の各作業ごとに、オプション・メニューの「J 項目設定」から「K 設定クリア」をクリックする。

7.2 オプション・メニュー

以下の「3」以降で説明している「入力形式」を選んだり「ファイル選択」を行う等の諸作業は、オプション・メニューの「J 項目設定」から選んでも同様の処理が可能である。特に音声ユーザーは、このメニューによる操作選択を行う。

7.3 入力ファイルの指定（点訳したいファイルの指定）

1) 「入力形式」を選ぶ。

右の下向き矢印をクリックするとリストが表示される。ファイル選択の際は下記の通り、元ファイルの後についている拡張子によって識別する。

例) test.kta の場合、以下になる。(ファイル名 test、拡張子.kta)

入力形式	対応拡張子	入力形式	対応拡張子
漢字かな英数	.txt	コータクン	.nab
ひらがな英数	.hra	BE	.bet
カタカナ英数	.kta	JETTY	.jty
標準点字	.brl	BES	.bes
BASE	.bse	ブレイルスター3	.bs
ブレイルスター2	.ble		

2) 読み込むファイルを指定。

「入力ファイル」の右にある、「ファイル選択」をクリックして「入力ファイル選択」画面を開け、読み込むファイルが存在するドライブ、フォルダー等を開き、ファイルを指定する。(ファイルを選ぶ時は目的のファイル名の上でクリックし、「ファイル名」の欄に、選んだファイルの名前が入ったか確認する。)ただし、表示されるのは「入力形式」で指定した形式のファイルのみで、例えば点字系を指定しているときは、「ひらがな漢字」等のテキストファイルは表示されない。「開く」をクリックして EXTRA for Windows 本体画面に戻ると「ディレクトリ」には、選んだファイルが存在しているドライブおよびフォルダーが、「入力ファイル」にはファイル名が表示されているはずである。ここで「内容」をクリックすると通常、ファイルの種類に適したソフトウェアが開き、指定したファイルの内容を見たり編集したりすることができる。

注：ファイルに適したソフトウェアとは、ファイルの拡張子に対してあらかじめ関連付けされているソフトウェアのことである。これは Windows95/98 の場合、エクスプローラーのオプション上で設定されている。

7.4 出力ファイルの指定

- 1) 「出力形式」をリストから選ぶ。

右の下向き矢印をクリックすると以下の内容がリスト表示される。

入力形式	対応拡張子	入力形式	対応拡張子
ひらがな英数	.hra	コータクン	.nab
カタカナ英数	.kta	BE	.bet
標準点字	.brl	JETTY	.jty
BASE	.bse	BES	.bes
ブレイルスター2	.ble	ブレイルスター3	.bs

- 2) 出力結果ファイルを指定。→「ファイル選択」

「出力ファイル」の右にある、「ファイル選択」をクリックして「出力ファイル選択」画面を開けると、入力ファイルと同じ場所が表示され、入力ファイル名を元にした名前（sample1.bes）が自動的に割り当てられる。ここで、出力ファイルの名前を新しく付けたい、あるいは出力ファイルの保存される場所を入力ファイルと異なる場所に変更したいときは、どこに保存するかドライブ、フォルダー等を選び、出力ファイルの名前を入力し、「開く」をクリックし確定する。その際、拡張子無しでファイル名のみを入力すると、出力形式に準じた拡張子が自動的に付加される。

例) 「出力形式」: BES、元ファイル名: sample1.txt とすると、

- a) 出力ファイル名: sampleA (と入力した場合)

→ 割り当てられた出力ファイル名: sampleA.bes と正しく設定される。また、拡張子を省略せずにファイル名全体を入力する場合は、拡張子を間違えないように注意する必要がある。

- b) 出力ファイル名: sampleB.bes (と入力した場合)

→ 割り当てられた出力ファイル名: sampleB.bes (適切)

- c) 出力ファイル名: sampleC.bse (と誤入力した場合)

→ 割り当てられた出力ファイル名: sampleC.bse.bes (不適切)

注: c) の場合、誤った拡張子に、「出力形式」での指定が追加され、不適切な名前になる。そのため、通常はファイル名のみを入力をお勧めする。

7.5 点訳に関する設定

- 1) 「点訳」(初期値:「日本 英2」)

入力が点字ファイルのときは(点訳済なので)この指定は不可ですが、漢字かな英数、ひらがな英数、カタカナ英数のときは、指定しなければならない。「点訳」の右にある下向き矢印をクリックすると点訳の種類が表示される。詳しくは8章の3「点字表記」を参照。

- a) [日本 英1]

日本語・英語1級。日本語を主言語、英文を補助言語として扱い点訳する。英文はつねに外国語引用符で囲む。英文の点訳は英語1級点字となる。

- b) [日本 英2]

日本語・英語2級。日本語を主言語、英文を補助言語として扱い点訳する。英文はつねに外国語引用符で囲む。英文の点訳は英語2級点字となる。

- c) [英1 日本]

英語1級・日本語。英文を主言語、日本語を補助言語として扱い点訳する。日本語と英文が混在している段落(改行コードで区切られる論理行)は、英文を外国語引用符で囲むが、1段落に日本の文字が1つも含まれていない場合には、外国語引用符を省略する。英文の点訳は英語1級点字となる。

d) [英2 日本]

英語2級・日本語。英文を主言語、日本語を補助言語として扱い点訳する。日本語と英文が混在している段落（改行コードで区切られる論理行）は、英文を外国語引用符で囲むが、1段落に日本の文字が1つも含まれていない場合には、外国語引用符を省略する。英文の点訳は英語2級点字となる。

e) [日本語・情報処理]

日本語とプログラム言語からなる文書を点訳する。デフォルトは日本語である。プログラム言語は情報処理点字表記で点訳する。

f) [情報処理・日本語]

日本語とプログラム言語からなる文書を点訳する。デフォルトはプログラム言語である。プログラム言語は情報処理点字表記で点訳する。

g) [日本語・北米情報処理]

日本語とプログラム言語からなる文書を点訳する。デフォルトは日本語である。プログラム言語は北米情報処理点字表記で点訳する。

h) [北米情報処理・日本語]

日本語とプログラム言語からなる文書を点訳する。デフォルトはプログラム言語である。プログラム言語は北米情報処理点字表記で点訳する。

7.6 レイアウト

1) 「レイアウト変更」

入力が標準点字以外の点字ファイルは既にレイアウト済だが、この機能を使用すれば既存のレイアウトを変更することができる。なおレイアウトとは「行」「マス」「ページ付け（含む、ページ初期値）」「インデント」の値のことである。すなわち、「入力形式」が漢字かな英数、ひらがな英数、カタカナ英数、標準点字のときは必要ないが、それ以外のときはチェックマークをつけることにより、「行」「マス」「ページ付け（含む、ページ初期値）」「インデント」の変更ができる。

注：「入力形式」がBASE、コータクンの場合には、データ構造上の問題（論理行と物理行の区別がない。）により、レイアウト変更が正しく行われない。したがって、レイアウト変更は勧めない。

2) 「行」「マス（1行あたりの点字の数）」（初期値：22行、32マス、範囲：10 ≤ n ≤ 99）

入力が漢字かな英数、ひらがな英数、カタカナ英数、標準点字のときは、出力される点字ファイルに「行」「マス」の設定をする。また、入力が標準点字以外の点字ファイルでは「行」「マス」は設定済だが、「レイアウト変更」をクリックしチェックマークをつければ、既存の設定を変更することができる。右の上下矢印をクリックして行数あるいはマス数を選ぶか、直接入力する。詳しくは、8章4の1), 2)を参照。

3) 「インデント」（初期値：0、範囲：0 ≤ n ≤ 一行文字数-5）

入力が漢字かな英数、ひらがな英数、カタカナ英数、標準点字のファイルのとき、プリントする用紙の左側を空けたい場合、つまり出力結果を右に寄せたい場合に設定する。また、入力が標準点字以外の点字ファイルでは「インデント」は設定済ですが、「レイアウト変更」をクリックしチェックマークをつければ、既存の設定を変更することができる。適切と思われる数値を「インデント」の右の上下矢印をクリックして選ぶか、直接入力する。

例）インデント：2とした場合 → 点字で2文字分、右に寄る。

注：インデント指定をすると、その分1行のマス数が減る。

4) 「ページ付け」（初期値：「付加する」）

入力が漢字かな英数、ひらがな英数、カタカナ英数のときは、出力される点字ファイルに「ページ付け」

を行う。出力ファイルにページ番号をつけるため、「ページ付け」の左のマスにチェックマークが付いているか確認し、もしないようなら、左のマスをクリックすることでチェックマークをつける。また、入力標準点字以外の点字ファイルは「レイアウト変更」をクリックしチェックマークをつければ、既存の設定を変更することができる。

注：「ページ付け」のチェックをしない場合、空のページ行を作る。

- 5) 「ページ初期値」（初期値：1、範囲： $-99 \leq n \leq 999$ 、ただし0は除く）

4) の「ページ付け」をチェックした場合、初期値を決める。ページの開始番号を「ページ初期値」の右の上下矢印をクリックして数を選ぶか、直接入力する。

注：負の数を指定した場合には、枚数分だけページを付けずに処理をするという指定になる。また、ページ番号に0は指定できない。

例) 「-1」の場合、1枚番号のない用紙を作り、2枚目から番号「1」がページとして付けられていく。「-5」であれば、5枚ページ番号なしのページが出力され、6枚目から番号「1」が付けられていく。

7.7 点字印刷

- 1) 「印刷部数」（初期値：1、範囲： $1 \leq n \leq 999$ ）

「印刷部数」は出力形式で指定したプリンターから、同一内容の出力結果をいくつほしいかを設定する。それぞれ右の上下矢印をクリックして数を選ぶか、直接入力する。

- 2) 「開始ページ」「最終ページ」（初期値：1、範囲： $1 \leq n \leq 999$ ）

「開始ページ」「最終ページ」は、それぞれ何ページから何ページまで印刷したいかを設定する。それぞれ右の上下矢印をクリックして数を選ぶか、直接入力する。

注：「開始ページ」が1ページ目からでないときは、「ページ初期値」を開始ページに合わせて修正する。そうしなかった場合、実際が何ページ目であっても、“1”から順次ページ番号がふられていく。

注：「開始ページ」「最終ページ」は、以下のようなときに使用する機能である。

- ・ 出力途中に紙切れや紙詰まりのようなトラブルがあった場合、どこから出力し直したいかを指定。
- ・ 入力ファイルが点字でそのまま出力する場合、ひらがな英数等のテキストファイルと違い、ページ番号はわかっているので必要ページのみ指定。

7.8 プリンタ設定

「プリンタ設定(一覧)」の画面上で設定する。

- 1) 「機種」

接続している点字プリンターの機種を、「機種」の右端の下向き矢印をクリックし、リストから選ぶ。

- 2) 「用紙長」（初期値：自由、範囲： $10 \leq n \leq 15$ ）

プリンタによっては用紙長の設定が無効な場合がある。詳しくは各プリンタのマニュアルで確認。

- 3) 「ドライバ」

使用のパソコンにインストールされているドライバが、「機種」の右端の下向き矢印をクリックするとリスト表示されるので、そこから適切なものを選ぶ。

- 4) 「方法」

「方法」の右端の下向き矢印をクリックし、印刷方法をリストから選ぶ。

- a) 「片面（印刷）」全てのプリンタに対して指定することができる。（初期値：「しない」）

- b) 「表・裏面 (印刷)」片面打ちプリンタで、表ページつまり奇数ページをまず印刷し、紙を裏面に裏返して偶数ページを印刷する。(初期値:「しない」)

注: 全ての片面印刷プリンタでこの方法が使えるわけではない。

作業の流れは以下のようになる。EXTRA for Windowsの実行 → 表面 (奇数ページ) の印刷 → 一時停止 → 裏面 (偶数ページ) 印刷 → 変換作業終了一時停止の所で、用紙の表、裏の入れ替えを行う。用紙をセットしたら、どれかのキー (リターンキーやスペースバー) を押すと、処理が再開され、裏面印刷に入る。この項目の選択可能な点字プリンタは以下の通りである。

- バーサポイント 1D-J, ニュー ESA721, BT3000, TP32, ブレイロコメット

- c) 「表面 (印刷)」 (初期値:「しない」)

片面打ち点字プリンタで表、裏、分けることで両面インターライン方式の印字が可能なもので、表面 (奇数ページ) だけを印刷する。b) の「表・裏面」の前半のプロセスだけを行うものです。この項目の選択可能な点字プリンタは以下の通りである。

- バーサポイント 1D-J, ニュー ESA721, BT3000, TP32, ブレイロコメット

- d) 「裏面 (印刷)」 (初期値:「しない」)

片面打ち点字プリンタで表、裏、分けることで両面インターライン方式の印字が可能なもので、裏面 (偶数ページ) だけを印刷する。[表・裏面] の後半のプロセスだけを行うものである。この項目の選択可能な点字プリンタは以下の通りである。

- バーサポイント 1D-J, ニュー ESA721, BT3000, TP32, ブレイロコメット

- e) 「両面同時 (印刷)」 (初期値:プリンタによる)

両面印刷機能を持つ点字プリンタで両面印刷を行うことを指定する。指定可能な点字プリンタは以下の通りである。

- NABCC (generic), ET, ブレイルエベレスト

7.9 オプション設定

「オプション設定」をクリックしてリストを出し、必要なものの左ボックスをクリックすることでチェックマークをつける。

- 1) 「ドルコマンド」

習熟者向けの機能。ドルコマンドとは点字表記やページレイアウトの設定を細かく指定するためのテキストファイル埋め込みコマンドである。ドルコマンドは、英数文字を半角ドルマーク「\$」と半角のコロン「:」で囲んで記述する。詳しくは8.9 [ドルコマンド] を参照。

- 2) 「折り返し削除」

1行を一定の長さで折りたたむことをフォールド (fold) という。ワープロ専用機で作成したファイルを、MS-DOSのテキストファイルに変換した場合やOCRで印刷文書をテキスト化した場合に、このようなフォーマットになっていることがある。原文がフォールドされていると、点訳やかな訳が乱れる原因になる。「折り返し削除」を使うと、自動的にフォールドされたファイルであるかどうかを判別し、もしそうなら論理行と物理行を区別したうえで、不要な改行、行頭のインデントを削除する。

- 3) 「改行削除」 (初期値:「しない」)

原文が、論理的な区切りの他に、文章の途中でレイアウトの都合により改行されていると、点訳やかな訳は乱れる。「改行削除」は全ての改行を削除して、こうした問題に対処する。本来は改行のうち、論理的な区切りとしての改行は残し、レイアウトのための改行は削除して点訳すべきですが (それを行おうとするのが「折り返し削除」)、判別は困難である。

注：「改行削除」は全ての改行コードを削除するために、論理的な区切りとしての改行コードも削除することに注意する。

4) 「改ページ削除」 (初期値:「しない」)

改ページコードを全て削除する。原文中には論理的な区切りや、レイアウト上の都合により、改ページコードが入っていることがある。本来はこれらのうち、論理的な区切りとしての改ページは残し、レイアウトのための改ページは削除して点訳すべきですが、判別は困難である。

注：「改ページ削除」は全ての改ページコードを削除するため、論理的な区切りとしての改ページコードも削除することに注意する。

5) 「全角スペース削除」 (初期値:「しない」)

余分な全角スペースが文字と文字の間に入っていると、点訳が正しく行われない。この機能を使うと全角スペースをあらかじめ削除したうえで変換作業を行う。

注：段落先頭の空白は削除しない。

6) 「半角スペース (・タブ) 削除」 (初期値:「しない」)

余分なスペースが文字と文字の間に入っていると点訳が正しく行われない。「半角スペース削除」は、全角文字の間の半角スペース・タブを削除する。また、アルファベットなどの半角文字の間の半角スペース・タブは一つを残して削除する。

注：段落先頭の空白とタブは削除しない。

例)	御 見 積 書	→	御見積書
	納 期	→	納期

7.10 作業開始

変換作業を開始する。EXTRA for Windows 画面上の作業1から5のうち、一つの作業のみ設定し実行を行うなら、そのまま「作業開始」をクリックする。

7.11 連続処理

作業1から作業5のうちのいくつかを、事前に設定した上で一括処理したいときに設定する。一括処理の対象としたい作業の画面内で、「連続処理」の右にあるチェックボックスをそれぞれの作業画面毎にチェックする。その後一括処理対象となっているいずれかの作業画面上から、「作業開始」をクリックする。

7.12 状況

「作業開始」スイッチの右にある「状況」が、“作業処理中”から“作業終了”になるまで待つ。

7.13 内容

もしIBMの点字編集プログラム(BES)がインストールされていれば、終了後、出力ファイル名の横の「内容」をクリックすると、点字編集プログラムが開くので、新しくできたファイルの内容を確認する。なお、BES形式ファイル(拡張子「.BES」)とBE形式ファイル(拡張子「.BET」)以外の点字ファイル形式を指定した、あるいは点字編集用ソフトウェアがインストールされていない場合は、EXTRA for Windows内蔵の点字表示用ソフトウェアが開く。

7.14 補足

- 1) 新しくできたファイルを修正する。

「11」で誤変換が発見されたら修正し、その後保存する。ただし、IBMの点字編集プログラム(BES)がインストールされていない時は、別途ご利用の点訳ソフトウェアを使って修正する必要がある。

8 機能リファレンス

8.1 入力ファイル形式

- 1) [漢字かな英数]

入力ファイルが通常の日本語文書、英語文書、あるいはそれらの混在したテキストファイルであれば、これを指定する。

- 2) [ひらがな英数]

入力ファイルがEXTRA for Windowsの「hra ファイル（全角ひらがなと英数からなる分かち書きファイル）」の場合に指定する。

注：半角カタカナ英数出力はできない。

- 3) [半角カタカナ英数]

入力ファイルがEXTRA for Windowsのkta ファイル（半角カタカナと英数からなる分かち書きファイル）の場合に指定する。

注：ひらがな英数での出力はできない。

- 4) [標準点字]

入力ファイルが標準点字ファイルである時に選択する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

- 5) [BASE]

点訳エディタ「BASE」で作成されたデータを入力ファイルとして利用する場合に選択する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

- 6) [ブレイルスター2]

点訳エディタ「ブレイルスター2」で作成されたファイルを入力ファイルとして用いる場合に指定する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

- 7) [コータクン]

点訳エディタ「コータクン」で作成されたファイルを入力ファイルとして用いる場合に指定する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

- 8) [BE]

点訳エディタ「BE」で作成されたファイルを入力ファイルとして用いる場合に指定する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

9) [JETTY]

点訳エディタ「JETTY」で作成されたファイルを入力ファイルとして用いる場合に指定する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

10) [BES]

点訳エディタ「BES」で作成されたファイルを入力ファイルとして用いる場合に指定する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

11) [ブレイルスター3]

点訳エディタ「ブレイルスター3」で作成されたファイルを入力ファイルとして用いる場合に指定する。

注：・ひらがな英数、半角カタカナ英数での出力はできない。

- ・点字表記は変更できない。
- ・前処理オプションは指定できない。

8.2 出力ファイル形式

EXTRA for Windows の出力ファイルの種類または出力先を選択する。

1) [ひらがな英数]

EXTRA for Windows の変換結果を分かち書きされた全角ひらがなと英数からなるファイルとして出力する。変換データには、点字のルールに則った分かち書きが施され、分かち書きのスペースには、全角スペースを用いる。

注：レイアウト指定はできない。

2) [半角カタカナ英数]

EXTRA for Windows の変換結果を分かち書きの半角カタカナと英数からなるファイルとして出力する。変換データには、点字のルールに則った分かち書きが施され、分かち書きのスペースには、半角スペースを用いる。

注：レイアウト指定はできない。

3) [標準点字]

EXTRA for Windows の変換結果を標準点字コードである北米情報処理点字コード（NABCC）形式のファイルに出力する。行、ページ等のレイアウト付けは行わない。

注：レイアウト指定はできない。

4) [BASE]

EXTRA for Windows の変換結果を「BASE」形式の点字ファイルとして出力する。

5) [ブレイルスター2]

EXTRA for Windows の変換結果を「ブレイルスター2」形式の点字ファイルとして出力する。

6) [コータクン]

EXTRA for Windows の変換結果を「コータクン」形式の点字ファイルとして出力する。

- 7) [BE]
EXTRA for Windowsの変換結果を「BE」形式の点字ファイルとして出力する。
- 8) [JETTY]
EXTRA for Windowsの変換結果を「ジェットイ」形式の点字ファイルとして出力する。
- 9) [BES]
EXTRA for Windowsの変換結果を「BES」形式の点字ファイルとして出力する。
- 10) [ブレイルスター3]
EXTRA for Windowsの変換結果を「ブレイルスター3」形式の点字ファイルとして出力する。

8.3 点字表記

「日本語英語自動点訳プログラム EXTRA for Windows」は、通常の日本語に加えて、英語や情報処理文書の点訳をサポートしている。英語の点訳には、1級英語点字、2級英語点字が利用されており、その実情に合わせた点訳モードをサポートする。また、情報処理文書の点訳は、情報処理用点字という特別の点字体系を用いる必要がある。コンピュータプログラムのマニュアル類、プログラミング言語のテキスト類、プログラムのソースリストの点訳を行うため EXTRA for Windows は4種類の情報処理点字表記をサポートしている。

- 1) [日本 英1]
日本語・英語1級。日本語を主言語、英文を補助言語として扱い点訳する。英文はつねに外国語引用符で囲む。英文の点訳は英語1級点字となる。
- 2) [日本 英2]
日本語・英語2級。日本語を主言語、英文を補助言語として扱い点訳する。英文はつねに外国語引用符で囲む。英文の点訳は英語2級点字となる。
- 3) [英1 日本]
英語1級・日本語。英文を主言語、日本語を補助言語として扱い点訳する。日本語と英文が混在している段落（改行コードで区切られる論理行）は、英文を外国語引用符で囲むが、1段落に日本の文字が1つも含まれていない場合には、外国語引用符を省略する。英文の点訳は英語1級点字となる。
- 4) [英2 日本]
英語2級・日本語。英文を主言語、日本語を補助言語として扱い点訳する。日本語と英文が混在している段落（改行コードで区切られる論理行）は、英文を外国語引用符で囲むが、1段落に日本の文字が1つも含まれていない場合には、外国語引用符を省略する。英文の点訳は英語2級点字となる。
- 5) [日本語・情報処理]
日本語とプログラム言語からなる文書を点訳する。デフォルトは日本語である。プログラム言語は情報処理点字表記で点訳する。
- 6) [情報処理・日本語]
日本語とプログラム言語からなる文書を点訳する。デフォルトはプログラム言語である。プログラム言語は情報処理点字表記で点訳する。
- 7) [日本語・北米情報処理]
日本語とプログラム言語からなる文書を点訳する。デフォルトは日本語である。プログラム言語は北米情報処理点字表記で点訳する。
- 8) [北米情報処理・日本語]
日本語とプログラム言語からなる文書を点訳する。デフォルトはプログラム言語である。プログラム言語は北米情報処理点字表記で点訳する。

8.4 レイアウト

点字として出力する際のレイアウトの指定。

- 1) [マス (1行文字数指定)] (初期値: 32、範囲: $10 \leq n \leq 99$)

点字のレイアウトのうち、1行に入れる点字の文字数(マス数)を決める。8×10インチ(B5サイズと同等)の点字用紙に打ち出す目安としては、欧米仕様の点字プリンタの場合は28、日本仕様の点字プリンタの場合はデフォルトの32である。

- 2) [1ページ行数] (初期値: 22、範囲: $10 \leq n \leq 99$)

点字のレイアウトのうち、1ページに入れる点字の行数を決める。ページ行も行数に含まれる。8×10インチ(B5サイズと同等)の点字用紙に打ち出す目安としては、片面打ちの場合および両面同時打ち出し(インターポイント式)の場合は、デフォルトの22、両面インターライン式では18である。

注: EXTRA for Windowsで指定する1ページ行数には、ページ番号行が含まれる。

- 3) [ページ付け] (初期値: 「付加する」)

ページ番号を付けるか付けないかを指定する。点字では、原則として用紙の右上にページ番号を付けることになっている。「ページ付け」のチェックをしない場合、空のページ行を作る。

- 4) [ページ番号初期値] (初期値: 1、範囲: $-99 \leq n \leq 999$)

3) で「ページ付け」をチェックした場合の番号の初期値を決める。範囲には負の数から正の数まで指定できる。負の数を指定した場合には、枚数分だけページを付けずに処理をするという指定になる。

注: ページ番号に0は指定できない。

例) 「-1」の場合、1枚番号のない用紙を作り、2枚目から番号「1」がページとして付けられていく。

「-5」であれば、5枚ページ番号なしのページが出力され、6枚目から番号「1」が付けられていく。

- 5) [インデント] (初期値: 0、範囲: $0 \leq n \leq 1$ 行文字数-5)

左端につねに一定の余白をあけたいときに使う。点字印刷の左端マージンとしても利用できる。

注: インデント指定をすると、その分1行のマス数が減ってしまう。

- 6) [レイアウト]

レイアウトを持つ点字ファイル(点字エディタファイル)のレイアウトを解除してあらためて新しいレイアウトを与える。デフォルトでは入力点字ファイルのレイアウトは解除しない。出力ファイルがひらがな英数、半角カタカナ英数のときは無効である。

8.5 プリンタ設定

EXTRA for Windowsの変換結果を出力する点字プリンタ、点字ディスプレイを選ぶ。各装置の種類と、EXTRA for Windowsメニューで指定する[印刷形式]とには、密接な関係があるので、点字出力装置の機能をあらかじめ知っておく必要がある。

8.6 印刷形式

- 1) [印刷部数] (初期値: 1、範囲: $1 \leq n \leq 999$)

指定した印刷部数になるまで、変換作業を行い、プリンタに出力する。

- 2) [開始ページ] (初期値: 1、範囲: $1 \leq n \leq 999$)

印刷させたいページ範囲の内の開始ページを指定する。途中から印刷する際には数字を具体的に指定することができる。

3) [終了ページ] (初期値:999、範囲: $1 \leq n \leq 999$)

印刷させたい範囲の内、最後のページを指定する。初期値は最大数となっていて、この場合「最後のページまで」印刷する。

8.7 印刷設定

以下は、印刷設定の中の「方法」から指定する。

1) [片面印刷] (初期値:プリンタにより異なる)

片面印刷モードを指定する。全てのプリンタに指定することができる。

2) [表・裏面印刷] (初期値:[しない])

片面打ちプリンタで、表、裏、分けて印刷することで、インターライン式両面印刷の可能なものに指定できる。作業の流れは、EXTRA for Windowsの実行 → 表面(奇数ページ)の印刷 → 一時停止 → 裏面(偶数ページ)印刷 → 変換作業終了となる。一時停止の所で、用紙の裏、表の入れ替えを行う。用紙をセットしたら、どれかのキー(リターンキーやスペースバー)を押すと、処理が再開され、裏面印刷に入る。この項目の選択可能な点字プリンタは以下の通り。

- バーサポイント 1D-J, ニュー ESA721, BT3000, TP32, プレイロコメット

3) [表面印刷] (初期値:[しない])

片面打ち点字プリンタで表、裏、分けることで両面インターライン方式の印字が可能なもので、表面(奇数ページ)印刷する。[両面印刷]の始めのプロセスだけを行うものである。この項目の選択可能な点字プリンタは以下の通り。

- バーサポイント 1D-J, ニュー ESA721, BT3000, TP32, プレイロコメット

4) [裏面印刷] (初期値:[しない])

片面打ち点字プリンタで表、裏、分けることで両面インターライン方式の印字が可能なもので、裏面(偶数ページ)印刷する。[両面印刷]の後半のプロセスだけを行うものである。この項目の選択可能な点字プリンタは以下の通り。

- バーサポイント 1D-J, ニュー ESA721, BT3000, TP32, プレイロコメット

5) [両面同時印刷] (初期値:プリンタによる)

両面印刷機能を持つ点字プリンタで両面印刷を行うことを指定する。指定可能な点字プリンタは以下の通り。

- NABCC (generic), ET, プレイルエベレスト

8.8 前処理

かな訳、点訳する入力ファイルの前処理を行う。変換作業を疎外するような漢字かな英数ファイルの特殊なレイアウトへの対策を行う。

1) [ドルコマンド] (初期値:[無効])

ドルコマンドとは点字表記やページレイアウトの設定を細かく指定するためのテキストファイル埋め込みコマンドである。ドルコマンドは、半角英数文字を半角ドルマーク「\$」と「:」で囲んで記述する。ドルコマンドには以下の3種類がある。詳しくは“8.9 [ドルコマンド]”を参照。

- 「点字表記」に関するコマンド

- 「レイアウト」に関するコマンド
- 「前処理」に関するコマンド

ドルコマンドは入力ファイル形式が漢字かな英数、ひらがな英数、半角カタカナ英数の場合に有効である。

2) [折り返し削除] (初期値:「しない」)

1行を一定の長さで折りたたむことをフォールド (fold) という。ワープロ専用機で作成したファイルをMS-DOSのテキストファイルに変換した場合やOCRで印刷文書をテキスト化した場合に、このようなフォーマットになっていることがある。「折り返し削除」を使うと、自動的にフォールドされたファイルであるかどうかを判別し、もしそうなら論理行と物理行を区別したうえで、不要な改行、行頭のインデントを削除する。

3) [改行削除] (初期値:「しない」)

このような場合に[改行削除]を指定すると不必要な改行が削除されて単語が分断されることがなくなる。全ての改行を削除する。[折り返し削除]で改行が削除されるのは、フォールドされているファイルに対してだけであるが、[改行削除]は1行の長さが一定でない場合にも有効である。

4) [改ページ削除] (初期値:「しない」)

OCRソフトで読み取ったファイルでは、ページの末尾に改ページコードが入っていることがある。これがあるとページの切れ目の変換結果が悪くなってしまう。このようなときに[改ページ削除]を指定する。なおこの機能を用いると、章の区切りなど、意味のある改ページコードまで削除してしまうので注意する。

5) [全角スペース削除] (初期値:「しない」)

テキストファイルには、タイトルなどの体裁を整えるために、余分に「スペース」や「」が入っていることがある。これをそのまま変換すると変換がうまくいかない。この機能を使うと全角スペースをあらかじめ削除したうえで変換作業を行う。

注: 段落先頭の空白は削除されない。

6) [半角スペース(・タブ)削除] (初期値:「しない」)

テキストファイルには、タイトルなどの体裁を整えるために、余分に「スペース」や「タブ」が入っていることがある。これをそのまま変換すると変換がうまくいかない。この機能を使うと半角スペースとタブをあらかじめ削除したうえで変換作業を行う。

注: 段落先頭の空白とタブは削除しません。また、半角英字中の半角スペースとタブは1マス残すように処理する。

例)	御 見 積 書	→	御見積書
	納 期	→	納期

8.9 ドルコマンド

ドルコマンドとは点字表記やページレイアウトの設定を細かく指定するためのテキストファイル埋め込みコマンドである。以下の形式で記述する。

\$ <コマンド> :

ドルコマンドには[点字表記]に関するコマンド、[レイアウト]に関するコマンド、[前処理]に関するコマンドの3種類がある。ドルコマンドは本文中の任意の場所に書くことができる。[点字表記]に関するコマンドと[前処理]に関するコマンドの有効範囲は文字単位である。すなわち、ドルコマンドの直後(：の次の文字)から、排他的な関係にある別のドルコマンドの直前まで有効となる。一方、[レイアウト]に関するド

ルコマンドの有効範囲は行単位である。[レイアウト] に関するドルコマンドを行の途中で埋め込んでも有効範囲はその行の先頭からとなる。したがって、[レイアウト] に関するドルコマンドはつねに行頭に書くようにする。[レイアウト] に関するドルコマンドは排他的な関係にある別のドルコマンドがくる直前の行まで有効である。

1) [点字表記] に関するドルコマンド

\$コマンド	機 能		有効範囲	排他的関係
	主 言 語	副 言 語		
\$EK1:	日本語	英語1級	文字単位	\$E?:
\$EK2:	日本語	英語2級	文字単位	\$E?:
\$EG1:	英語1級	日本語	文字単位	\$E?:
\$EG2:	英語2級	日本語	文字単位	\$E?:
\$EJ1:	日本語	情報処理	文字単位	\$E?:
\$EJ2:	情報処理	日本語	文字単位	\$E?:
\$EN1:	日本語	北米情報処理	文字単位	\$E?:
\$EN2:	北米情報処理	日本語	文字単位	\$E?:
\$EM:	数 式	_____	文字単位	\$E?:
\$EX:	無変換	_____	文字単位	\$E?:

a) \$EM: 数式

以下の文字列を数式とみて変換する。

0～9	数字	/	分数記号
.	小数点	(開き小かっこ
,	桁区切り)	閉じ小かっこ
+	加算	{	開き中かっこ
-	減算	}	閉じ中かっこ
	乗算	[開き大かっこ
//	除算]	閉じ大かっこ
=	等号	@	ルート記号
[小なり	^	上つき添字
]	大なり	-	下つき添字

b) \$EX: 無変換

点字パターンを直接入力ファイルに埋め込むときに指定する。以下はアスキーコードに対応するNABCCコードとして出力される。

2) [レイアウト] に関するドルコマンド

\$コマンド	機 能	有効範囲	排他的関係
\$FLn:	1 行 n 文字	行単位	なし
\$FIn:	左端マージン n 文字	行単位	なし
\$FTn:	タイトル行指定	行単位	\$FR: \$FD: \$@FTn: \$@FR:
\$FR:	右寄せ行指定	行単位	\$FTn: \$FD: \$@FTn: \$@FR:
\$FD:	本文行指定	行単位	\$FTn: \$FR: \$@FTn: \$@FR:
\$@FTn:	タイトル行指定	1 行	なし
\$@FR:	右寄せ行指定	1 行	なし

a) \$FTn: タイトル行指定

以下の各行を、n 文字下げのタイトル行として処理する。2 行以上に跨るときは 2 行目以下は、さらに 2 マス下げされる。\$FD: もしくは \$FR: の指定があるまでタイトル行として処理する。

b) \$FR: 右寄せ行指定

\$FD: もしくは \$FTn: の指定があるまで右寄せ行として処理する。

c) \$FD: 本文行指定

\$FR\$ および \$FTn\$ を解除する。

d) \$@FTn:

この指定のある行だけがタイトル行として処理される。次行からは前行と同じ行モード（本文行または右寄せ行）に戻る。

e) \$@FR:

この指定のある行だけが右寄せ行として処理される。次行からは前行と同じ行モード（本文行またはタイトル行）に戻る。

例) \$@ft10:GUI用スクリーンリーダーの現状と課題

\$@ft12:ー北米と欧州の取り組みを中心に

\$@fr:石川准

\$@ft4:1. はじめに

パーソナルコンピュータおよびエンジニアリング・ワークステーションの利用環境は、近年グラフィカル・ユーザー・インターフェイス（GUI）へと急速に移行しつつある。

3) [前処理] に関するドルコマンド

\$コマンド	機 能	有効範囲	排他的関係
\$AZ:	全角空白削除	文字単位	\$AZX:
\$AZX:	\$AZ\$を解除する	文字単位	\$AZ:
\$AH:	1 マス以上の半角スペース・タブ削除	文字単位	\$AHX:
\$AHX:	\$AH\$を解除する	字単位	\$AH:

各ドルコマンドの機能は、[3 点字表記]、[4 レイアウト]、[7 前処理] についての先の説明を参照する。ここでは、まだ触れていないものだけ説明する。

8.10 英語点訳についての補足

ここでは、EXTRA for Windowsが英語点訳を行う際に、テキストファイル中の特殊記号をどのように処理するかを説明する。

1) 「” (ダブルコーテーション)」、 「' (アポストロフィ)」、 「‘ (アクサングラフ)」 の区別

EXTRA for Windows は、半角の「”」を二重引用符 (ダブル・コーテーション)、 「'」をアポストロフィー、 「‘」を引用符 (シングル・コーテーション) として処理する。また、開きと閉じの区別は前後関係で判断している。

2) イタリックス、アクセント符、文字符号

英語点訳では、イタリックス、アクセント符、文字符号も重要である。EXTRA for Windows は、入力英文テキスト中の「_」をイタリックス、「^」をアクセント符、「|」を文字符号として解釈する。点字のイタリックス符号は 4・6 の点である。「_」が英文中にあるとイタリックス用マークとみて 4・6 の点に変換する。イタリックスを付けたい単語の直前に「_」を付ける。また、ダブル・イタリックスを付けたい場合には、「_」を 2 つ書く。

英文中にフランス語やドイツ語等のアクセントの付いた文字を書きたい場合には、アクセント符を付けた文字の直前に「^」を書く。EXTRA for Windowsは、これを点字のアクセント記号（4の点）に変換する。英文中に1文字のアルファベットを書きたい場合には、点字ではその文字が単語ではなく文字であることを示すために、その文字の前に文字符号（5・6の点）を付ける。EXTRA for Windowsは自動的にこの処理を行うが、「a」と「I」では自動的には文字符号は付かない。より厳密な点訳をめざす場合には、これらが単語ではなく文字として使われるときは、文字の直前に「|」を書く。EXTRA for Windowsはこれを5・6の点に変換する。英文中で単独のギリシア文字を書く場合には、そのギリシア文字の前に2の点を書く。EXTRA for Windowsでは、「,」を2の点に変換するので、ギリシア文字を表現したい場合には、この「,」をアルファベットに前置する。

例) The widespread use of systems like MS-Windows, X-Windows, or OS/2 at the office work place makes clear that blind computer users need access to graphical user interfaces (GUIs) if they should not loose their ability to communicate with and through computers. One of the great advantages of a GUI for sighted users is direct manipulation. Instead of typing commands with numerous parameters, users move visualized objects on the screen using a mouse. ‘Alternative methods’ of direct manipulation need to be developed for blind users.

8.11 囲み記号

点字には独特の囲み記号がある。それらと墨字コードの対応は決められていない。EXTRA for Windowsは次のような対応付けを独自規則として用いる。

墨 字	点字 (NABCC)	墨 字	点字 (NABCC)
「あああ」	-アアア-	[あああ]	;7アアア72
<あああ>	;’アアア,2	【あああ】	”-アアア-1
『あああ』	;-アアア-2	[あああ]	”-アアア-1
(あああ)	7アアア7	《あああ》	;-アアア-2
{あああ}	”7アアア71		

8.12 その他

EXTRA for Windowsは、改行コードまでの間に漢字やかな文字が1つもないときには、自動的にその行全体を英文と認識する。主言語が日本語の場合には、行全体が外国語引用符で囲まれ、主言語が英語の場合には、外国語引用符は省略される。

- EXTRA for Windowsは入力ファイル中の制御コードを自動的に削除する。ただし改行コード、タブコード、改ページコードは、原則として、レイアウト情報として受け取る。
- タブコードは、スペース2つに置き換えられる。
- スペースコードが8個（全角スペースコードでは4個）以上連続する場合には、スペース8個分に縮められる。またそのような行はタイトル行とみなされ処理される。

9 ユーザー辞書

9.1 ユーザー辞書とは

EXTRA for Windowsは約15万語の辞書と独自開発の自然言語処理変換アルゴリズムにより日常のほとんどの文書をほぼ正確に点訳することができる。しかし、様々な分野の専門用語の全てを網羅しているわけではない。どうしても多少の誤変換は避けられない。

よく現れる単語が誤変換される場合は、その単語をユーザー辞書に登録することで変換精度を上げ、校正作業を軽減することができる。なお、EXTRA for Windowsではシステム辞書よりユーザー辞書が優先されるので、ユーザー辞書は強力に働く。以下の説明を理解したうえで注意深く登録作業を行う。

9.2 登録作業の手順

ユーザー辞書への単語登録を行うには「ユーザー辞書管理」をクリックする。

- 1) [追加] 「追加」を選び、「単語」（変換対象文字列）「読み」「品詞コード」を入力する。
なお、「単語」に半角文字を含めることや、「読み」に漢字を含めることはできない。
また「品詞コード」は後述の説明を参考に選択する。
- 2) [編集] 「編集」を選ぶと、登録された内容を修正・変更することができる。
- 3) [削除] 「削除」を選び、削除する登録を決めると、その登録はユーザー辞書から削除される。
- 4) [検索] 「検索」を選んで文字列を入力すると、ユーザー辞書に登録されている単語のなかから該当するものを探す。
- 5) [更新] 「更新」を選ぶとユーザー辞書が更新される。
- 6) [終了] 「終了」を選ぶとユーザー辞書管理を終了させることができる。

9.3 品詞コード

「読み」と「品詞コード」の入力は、以下を参考に行う。

人名	良範	よしのり	ま行五段動詞語幹	鉄	はさ
人姓	亜久里	あぐり	ら行五段動詞語幹	霽れ上が	はれあが
	豆生田	あみうだ	わ行五段動詞語幹	闊ざあ	せめざあ
	麻生	あそー (注1)	が行五段動詞語幹	喘	あえ
名詞	人型機械	ひとがたきかい (注2)	ば行五段動詞語幹	學	まな
固有名詞	阿賀野川	あがのがわ	一段動詞語幹	飛び出	とびで
	亜細亜	あじあ	さ行サ変動詞語幹	和はく	わはく
か行五段動詞語幹	咽びな	むせびな (注3)	ざ行サ変動詞語幹	論	ろん
さ行五段動詞語幹	印	しる	形容詞語幹	恋し	こいし
た行五段動詞語幹	引きはな	ひきはな		わびし	わびし (注5)
な行五段動詞語幹	死	し	形容動詞語幹	和やか	なごやか
は行五段動詞語幹		(注4)	その他	しかしながら	しかしながら

注意：読みは半角仮名か全角ひらがなで入力する。登録後は全て半角かなに変換される。

注1：読みが「アソウ」でなく「アソー」であることに注意。点訳規則参照。

注2：読みに入れる区切りには「^ゝ」またはスペースを使う。

注3：語幹だけを登録。

注4：現代語には存在しない。

注5：ひらがなで始まる単語の場合は読みの先頭に区切り記号をいれないと前の文字列に区切りなしで連結されるので注意する。

9.4 エディタでのユーザー辞書作成

\program files\jun\extra\user.org をエディタで開いて「単語」「読み」「品詞コード」を直接入力することもできる。ただし user.org 自体は辞書のソース・ファイルにすぎず、これだけではユーザー辞書は更新されない。EXTRA for Windows を起動して「ユーザー辞書管理」を開いて「更新」を実行する必要がある。エディタで入力する際は以下の規則に従う。USER.ORG は 1 単語 1 行で、「登録語」、「読み」、「品詞コード」で構成されている。「登録語」と「読み」の間および「読み」と「品詞コード」の間は「半角スペース」を入れ、点字のマスあけに相当する部分には、ティルダ「~」を入れる。単語に半角文字は含めない。また読みには全角文字は含めない。

	単語 半角カタカナの読み 品詞	注 意
例	希臘 ~キ° リシャ \$ 葡萄牙 ~ホ° ルトカ° ル \$ 西比利亞 ~シヘ° リヤ \$ 日並皇子 ヒナミシノ~ミコ ! 柿本人麻呂 ~カキノモトノ~ヒトマロ !	~ は、マスあけに相当 読みは半角カタカナで入力

以下に品詞コード表を示す。

品詞コード	意 味	品詞コード	意 味
!	人名	R	ら行五段動詞語幹
#	人姓	W	わ行五段動詞語幹
%	名詞	G	が行五段動詞語幹
\$	固有名詞	B	ば行五段動詞語幹
K	か行五段動詞語幹	I	一段動詞語幹
S	さ行五段動詞語幹	C	さ行サ変動詞語幹
T	た行五段動詞語幹	Z	ざ行サ変動詞語幹
N	な行五段動詞語幹	A	形容詞語幹
H	は行五段動詞語幹	V	形容動詞語幹
M	ま行五段動詞語幹	-	その他

謝 辞

本研究を進めるにあたり、様々な助言ご意見を頂いた、藤芳先生（大学入試センター）、山口先生（日本大学）、佐藤教授（お茶の水女子大学）、鈴木教授（九州大学）、原教授（筑波技術短期大学）に感謝致します。

A EXTRA インストール方法

本報告第II部で述べた、日本語英語自動点訳プログラム EXTRA for WINDOWS に関するインストール手順を、以下に述べる。なお、動作環境環境は Windows95/98 で、メモリーは 32MB 以上のパソコンを推奨する。

A.1 EXTRA for Windows のインストールプログラム

1) 「EXTRA インストールプログラム」

EXTRA for Windows の CD-ROM または フロッピーディスクを挿入すると、自動で「EXTRA インストールプログラム」画面が起動するので、「次へ」をクリックする。

注： 起動しない場合は、以下のいずれかを行なう。

- (1) エクスプローラーから CD-ROM または フロッピーディスクが存在するドライブを探し、ファイル名「Setup.exe」をダブルクリックすればインストール画面が起動する。
- (2) スタートアップ・メニューから「ファイル名を指定して実行」を選び、「参照」をクリックして(1)と同様にドライブ、およびファイル名を選び、「OK」をクリックすれば、インストール画面が起動する。

2) 「インストール先のディレクトリの選択」

表示内容でよければ、そのまま「次へ」をクリックする。もし、表示されている場所以外にインストールしたい場合は、「参照」をクリックして希望の保管場所を指定する。

3) 「サンプルディレクトリの選択」

表示内容でよければ、そのまま「次へ」をクリックする。もし、表示されている場所以外にインストールしたい場合は、「参照」をクリックして希望の保管場所を指定する。

4) 「インストールの準備を完了しました」

「次へ」をクリックする。

5) 「インストールを完了しました」

「完了」をクリックして、インストールを終了する。

これで、「EXTRA for Windows」がインストールされた。スタートアップメニューも追加されているはずである。続いて、プリンターを使用するためのインストールを行う。

A.2 プリンタドライバ

1) 「プリンタ」

スタートメニューの「設定」を開き、その中の「プリンタ」を更に開く。開いたら、「プリンタの追加」アイコンを2回クリックして選ぶ。

2) 「プリンタウィザード」

「プリンタウィザード」が開いたら、「次へ」をクリックする。

- a) 「プリンタはどこに接続されていますか」の答えとして、ローカルを選択し「次へ」進む。
- b) プリンタの製造元を指定する。「ディスク使用」を選び、「フロッピーディスクからインストール」の画面を開く。「配布ファイルのコピー元」に「1 EXTRA for Windows のインストール」での CD-ROM または フロッピーディスクを指定し、「OK」をクリックする。

注：「1 EXTRA for Windows のインストール」からの流れでなく「2 プリンタドライバのインストール」を単独で行う場合、EXTRA for Windows の CD-ROM または フロッピーディスクを挿入してから、「b)」を行う。挿入すると CD-ROM または フロッピーディスクから「EXTRA インストールプログラム」画面が自動起動するので、「キャンセル」をクリックして元の画面に戻る。

- c) 次の画面の「ファイル」の欄に「EXTRA Braille Printer Driver」が表示される。選択（反転表示）されていなかったら、1 回クリックして選択し、「次へ」をクリックする。
- d) 「現在のドライバを使う」か、「新しいドライバに置き換える」かを選択し、「次へ」をクリックする。
- e) プリンタで使用するポートを選ぶ。「利用できるポート」のリストから、該当するもの（後述の3）「プロパティ」参照）が選択（反転表示）されていなかったら、1 回クリックして選択し、「次へ」をクリックする。

注： ・ 現行の BT5000、BT3000 機種ではパラレル出力はできない。
・ 現行の ESA721、TP32 機種では PC98 の COM1 からのシリアル出力はできない。

- f) 次の画面でプリンタ名をつける。このとき「Windows アプリケーションで、このプリンタを通常のプリンタとして使いますか？」の質問には、「いいえ」を選ぶ。
- g) 最後に、「印字テストを行いますか？」に「はい」を選択し、「完了」をクリックすれば、インストールは完了。

3) 「プロパティ」

スタートメニューの「設定」を 1 回クリックして開き、その中の「プリンタ」を更に 1 回クリックして開く。開いた画面に、「EXTRA Braille Printer Driver」のアイコンができていないはずである。このアイコンが選択されて反転表示していなかったら、アイコン上で 1 回クリックして、選択（反転表示）する。そのまま 1 回右クリックしてするか、メニュー・バーの「ファイル」を 1 回クリックするかして、メニューリストを表示し、「プロパティ」を選ぶ。

a) パラレル接続の場合

注： ・ 電源を入れる前に、点字プリンターをパラレル・ポートに接続していない場合は、一度 Windows を終了し電源を切り、接続してから再起動する。その後、2 の 3）からやり直す。
・ すでに一般のプリンター（墨字用）がパラレルポートに接続されている場合は、増設のパラレル・ボードを用意し、パソコンに設置してから当作業を行う。

「EXTRA Braille Printer Driver のプロパティ」の「詳細」タブをクリックする。「印刷先のポート」の下向き矢印を 1 回クリックし、リストを出す。一般のプリンター（墨字用）がパラレルポートに接続されている場合は LPT2 を、接続されていない場合は、LPT1 を選ぶ。

b) シリアル接続の場合

注： ・ 電源を入れる前に、点字プリンターを COM1 または COM2 に接続していない場合は、一度 Windows を終了し電源を切り、接続してから再起動する。その後、2 の 3）からやり直す。
・ 点字プリンタ・メーカーが供給するシリアルケーブルを使う必要がある。汎用のシリアル・ケーブルでは動作しない。

「EXTRA Braille Printer Driverのプロパティ」の「詳細」タブをクリックする。「印刷先のポート」の下向き矢印を1回クリックし、リストを出す。事前に接続したポートに合わせて、COM1または、COM2を選ぶ。

次に、「ポートの設定」をクリックして開き、通信パラメータの設定を行う。
(基本的に点字プリンターの要求する通信パラメーター設定に合わせる必要がある。)
詳しくは各点字プリンターのマニュアル及び現在の通信パラメータの設定を確認する。

例) 以下は一般的な設定内容。

通信速度	:	9600 baud
パリティチェック	:	なし
ビット長	:	8
ストップビット	:	1
フロー制御	:	Xon/Xoffまたはハードウェア

注：フロー制御についての解説

フロー制御の選択は、点字プリンターの仕様と設定、使用するRS232Cケーブルの結線により異なる。以下のことを参考にしながら、プリンター・メーカーにサポートを求める。

- (1) 点字プリンターがRTS-CTS制御のハードウェア・ハンドシェイクをサポートしている場合プリンターをそのように設定し、市販のクロスケーブルを使い、パソコンの「ハードウェア制御」を選択する。ただし、ブレイルコメントのRTS-CTS制御は特殊なので、メーカーが供給するケーブルを使う必要がある。
- (2) 点字プリンターがDTR-DSR制御のハードウェア・ハンドシェイクをサポートしている場合プリンターをそのように設定し、パソコンの側のRTSとプリンター側のDSR、パソコン側のCTSとプリンター側のDTRのつながったクロスケーブルを使い、パソコンの側で「ハードウェア制御」を選択。ケーブルの入手についてはプリンター・メーカーに問い合わせる。
- (3) 点字プリンターがXON-XOFF制御のソフトウェア・ハンドシェイクをサポートしている場合プリンターをそのように設定し、パソコンの側でも「XON-XOFF制御」を選択する。

パソコン側のコネクタが9ピンの場合には、TXDとRXDのクロスがなされているケーブルで動作する。メーカー供給のケーブルはこの条件を満たしている。詳しくはプリンター・メーカーに問い合わせる。

パソコンがNEC-PC98シリーズで、コネクタが25ピンの場合、パソコン側のCTSにONを供給する必要があり、プリンターの側のRTSかDTRと繋いで対処しなければならない。

注：RTS、DTRから信号を出さないプリンターでは使えない。

なおパソコン側でのCTSとDTRあるいはRTSのループは機能しない。点字プリンター・メーカー供給のケーブルは、とくにPC98用ということわりがない限り使用しない。メーカーによってはシステムをクラッシュさせるケーブルもあるので注意する。適切なケーブルの入手については、プリンター・メーカーに問い合わせる。

続いて、点字表示用ソフトウェアを使用するためのインストールを行う。

A.3 点字フォント

- (1) スタートメニューの「プログラム」を開き、その中の「エクスプローラ」をクリックして開く。
- (2) Windows フォルダに「EUDC.TTE」という外字フォント・ファイルが既にあるか確認する。もしあるようなら、点字フォントのインストールは勧めない。

注： 点字フォントは外字を利用しているため、点字フォントをインストールすると、この「EUDC.TTE」ファイルを上書きすることになり、現在使用中の外字が表示されなくなる。

点字フォントのインストールは、点字ファイルの内容を確認したいときにだけ必要なものであるので、Extra for Windows にとっては本質的な部分ではない。それでも点字フォントを表示したい場合は、「EUDC.TTE」ファイルをしかるべき場所にコピーして、保存しておくことを勧める。

- (3) Extra for Windows の CD-ROM または フロッピーディスクのルート・フォルダに「EUDC.TTE」という点字外字フォント・ファイルがあるので、それを選んで Windows フォルダにコピーする。

以上で EXTRA for Windows を使用するための準備は終了。