

スキャン方式による肢体不自由者用 キーボードエミュレーションプログラムの開発

小山智史

弘前大学 教育学部

概要 重度肢体不自由者用のキーボードエミュレーションプログラムを開発した。1 接点の外部スイッチを用いて、表示文字からスキャン方式で選択する。OSの一部として動作するため、既存のアプリケーションプログラムと組み合わせて利用できる。文字表示がアプリケーションプログラムの表示と競合しないよう工夫した。また、2 操作選択方式を用いて入力効率を改善した。

**Development of a keyboard emulation program
using the scanning method for the severely motor disabled**

Satoshi KOYAMA

Faculty of Education, Hirosaki University
Bunkyo-cho 1, Hirosaki-shi, 036 Japan

Abstract We developed a keyboard emulation program of MS-DOS personal computer for the severely motor disabled. Characters in the scanning table can be selected by an external sensor. This program is a part of MS-DOS operating system, and many application program can be used. New ideas are included; single line scanning table and two stage selection mechanism.

1. はじめに

MS-DOSパソコンで動作するスキャン方式のキーボードエミュレーションプログラムを開発した。キーボードの表示位置と文字の選択方法に工夫をした。

重度肢体不自由者用のパソコンインタフェースとして、スキャン方式が用いられることが多い。その実施方法は次の3つに大別できる。

- (1)専用機によるもの (例えば[1])
- (2)パソコン+専用アプリケーションプログラムによるもの (例えば[2][3])
- (3)パソコン+汎用アプリケーションプログラム+専用インタフェースによるもの (例えば[4][5])

(1)(2)の方法は、優れた操作性が期待できる一方、ワープロなどに用途が限られる。(3)の方法は、操作に支障がないならば、既存のアプリケーションプログラムを利用できるだけでなく、健常者と機器やデータを共有できるなど利点は多い。

専用インタフェースは、外部に専用のハードウェアを接続する方法 (例えば [4]) と、パソコン内の基本プログラムで対処する方法 (例えば [5]) がある。前者は、ほとんど全てのアプリケーションプログラムに適用できること、後者は、特別な装置を必要としないため簡便であることにそれぞれ特徴がある。また、後者では、キーボードの表示がアプリケーションプログラムの表示と競合するため、使用時に表示位置を切り替えるなどの工夫が必要である。文献 [6] ではエディタの画面分割機能を利用してこの問題を回避している。

本報告では、筆者らが開発した専用インタフェースプログラムについて述べる。次の2点に特徴がある。(1)表示の競合を避けるために、キーボードをシステム行 (ディスプレイの最下行) に表示した。(2)スキャン速度を変えた2操作選択方式を用いて入力効率を改善した。

スキャン方式における入力効率については、文献 [7] を参考に検討を行った。

2. インタフェースソフトの詳細

2.1. エミュレータの動作

図1はインタフェースの動作説明図である。図において、キーボードエミュレータ (1finger) がディスプレイにキーボードを表示し、マウスからのクリック操作により、該当するキー情報をキーボード BIOS に引き渡す。この内容はアプリケーションプログラムにおいてはキーボードからの入力と同等に扱われる。

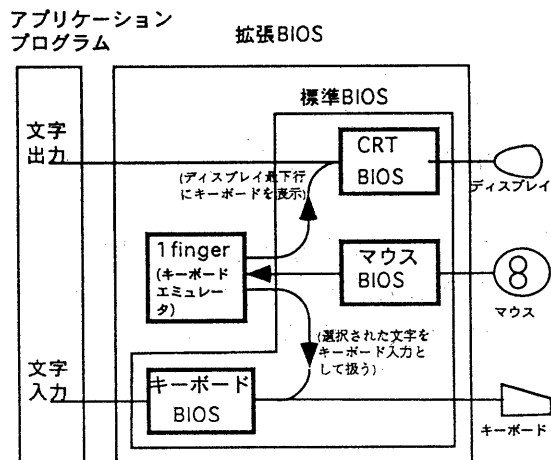


図1 キーボードエミュレータの動作

本プログラムの動作環境は次のとおりである。

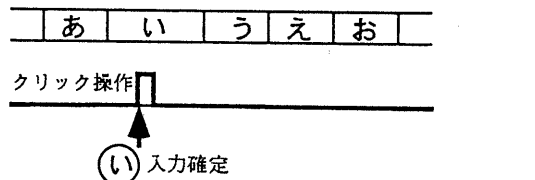
- ・パソコン 富士通FMRシリーズ, FM16β
- ・OS MS-DOS V3.1
- ・メモリ消費量 約4.6kB

スキャン入力を受け付けながら、通常のキーボードからの入力操作も並行して可能である。SHIFTキーやCTRLキーの同時操作を順次操作で代行する機能も持つが、ここでは述べない。

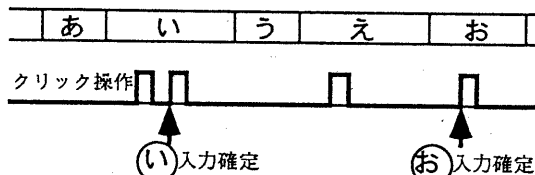
2.2. 操作の詳細

文字の選択方法は、利用者の状況に応じて以下に示す3つのモードのいずれかを選択する。操作方法については文献[8]を参考にした。図2にタイミングチャートを示す。操作スイッチは、ここでは、マウスボタンの操作として説明するが、実際

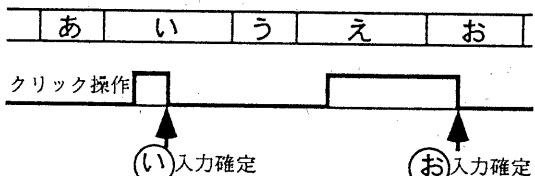
は利用者の状況に応じて呼気スイッチやタッチスイッチなどをマウスコネクタに接続して利用する。ただし、操作スイッチについての具体的な検討は行っていない。



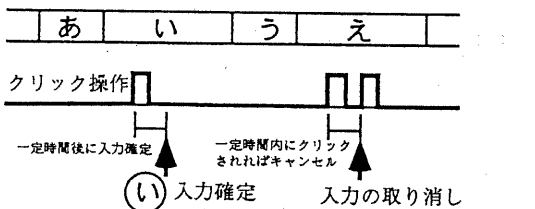
(a) シングルクリック操作



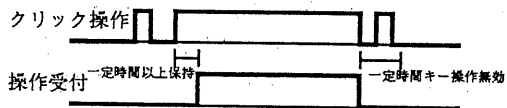
(b) ダブルクリック操作



(c) クリックホールド操作



(d) 入力の取り消し(シングルクリック操作の場合)



(e) チャタリングの除去

図2 操作とタイミングチャート

(1) シングルクリック操作 (図2(a))

システム行 (ディスプレイの最下行) に「次 CtrlShftCap a b c d e ...」のように「スキャンキーボード」が表示され、その中の1箇所が反転表示される (図3)。反転表示位置は自動スキャンされるほか、マウス操作で手動スキャンすることもできる。入力したい文字のところでマウスの左ボタンをクリックすると、その文字が選択され、「キー入力」される。入力後、その位置から再びスキャンが行われる。

次を選択するとキーボードの表示面が「次 12345...」に入れ替わる。次の選択はマウスの右ボタンのクリックでも行うことができ、この場合は任意の選択位置で操作できる。キーボード表示面は次の4つである。

次CtrlShftCap a b c...x y z, . TabEscBsCr
次123...0!?:+*/=#%&({}[]<>「」¥' " ^ _ , TabEscBsCr
次アイウエオ...ワヲンアイウエオヤユヨツ' . . . BsCr
次←→↑↓漢変無登 f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 DelBsCr

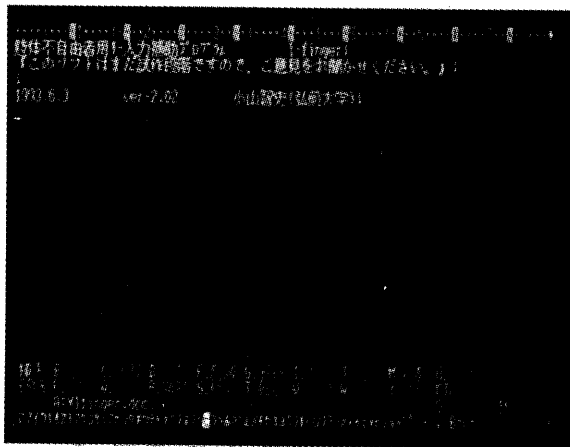


図3 ワードプロソフトと組み合わせた使用例

(2) ダブルクリック操作 (図2(b))

(1) のクリック操作を次のように変更したものである。第1クリックを行った後スキャン速度が遅くなり、第2クリックで「キー入力」される。同じ位置で2度クリックする必要は無いので、第1

クリックは目的位置の少し手前で行うことができる。第1クリックを行ってから一定時間を過ぎると、第1クリックは無効となる。

(3) クリックホールド操作 (図2(c))

(1) のクリック操作を次のように変更したものである。クリックホールド中にスキャン速度が遅くなり、クリック開放時に「キー入力」される。同じ位置で開放する必要は無いので、クリックの押し下げは目的位置の少し手前で行うことができる。押し下げたまま一定時間を過ぎるとクリックは無効となる。

(4) 誤操作の取り消し (図2(d))

操作から入力確定までに遅延時間を設け、その遅延時間内にクリック操作がなされると入力を取り消される機能を用意した。図はシングルクリック操作の場合についてのみ示した。

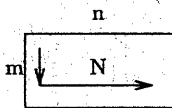
(5) チャタリング防止 (図2(e))

ひとつのクリック操作が正しく解釈されるために、一定時間以上クリック操作を保持して初めて操作が有効になる機能と、クリック操作解放後の一定時間を操作無効にする機能を用意した。

3. 操作方法に関する検討

ここでは、入力方法と操作時間の関係について検討する。

(1) 行列スキャン2操作選択法



広く用いられている方法である。スキャンの単位移動時間は、その時間内に確実にスイッチ操作ができるようにする必要があるが、この時間を T とする。キーボードの表示を、 m 行、 n 列とすると、文字の平均確定時間は次式で計算できる。ただし、入力する文字は等確率であるとする。

$$\begin{aligned} \bar{t} &= \text{縦の平均確定時間} + \text{横の平均確定時間} \\ &= \frac{m}{2} T + \frac{n}{2} T \end{aligned} \quad (1)$$

例えば、5行13列の中から文字を選択する場合の平均確定時間は $\bar{t}=9T$ である。ここで、表示文字数を N 文字 ($N=m \times n$) 一定とすれば、

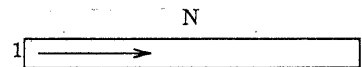
$$\bar{t} = \frac{N}{2n} T + \frac{n}{2} T \quad (2)$$

で、 $\frac{\partial \bar{t}}{\partial n} = 0$ より \bar{t} は、 $m=n=\sqrt{N}$ の時

$$\bar{t}_{\min} = \sqrt{N} T \quad (3)$$

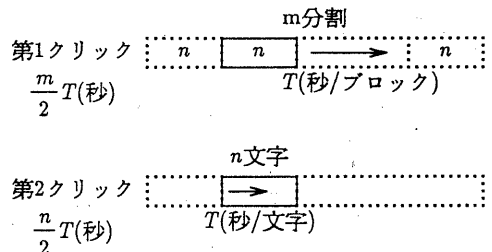
で最小となる¹。

(2) 行スキャン1操作選択法



表示を1行にすれば操作は1度ですむが、 $\bar{t} = \frac{N}{2} T$ の時間がかかるので効率は悪い。2章に示したプログラムはシステム行 (ディスプレイ最下行) 1行を用いて文字表示を行っている。 $N=65$ (仮名表示面) の時、 $\bar{t}=32.5T$ である (シングルクリック操作)。この効率を改善するために、以下に示す2操作選択法の考え方を導入した (ダブルクリック操作²)。

(3) m分割2操作選択法



まず、 N 文字を n 文字ずつ m 個のブロックに分割する ($N=m \times n$)。第1クリックでブロックを選ぶ。ブロックの単位移動時間を T とすれば、目的文字を含むブロックを選ぶための平均時間は $\frac{m}{2} T$ である。次に、第1クリックで選んだプロ

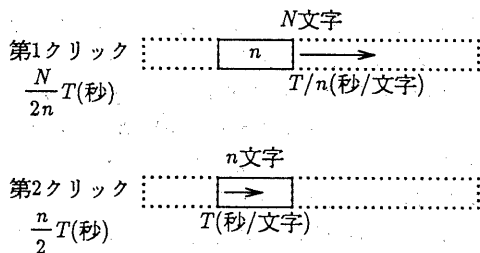
¹ 文献[8]の(2)式に相当する。

² クリックホールド操作の場合は、第1クリックをスイッチ投入時、第2クリックをスイッチ開放時と読み替える。

ックの中で、文字を選ぶための第2クリックの平均確定時間は、ブロック内での文字の単位移動時間を T とすると $\frac{n}{2} T$ である。従って、この方法による文字の平均確定時間 \bar{T} は (1) 式に等しく、 $m=n=\sqrt{N}$ の時 (3) 式で最小となる。

この方法は、 m 行 n 列の表を単に1行に並べなおしたと考えれば、(1) 行列スキャン2操作選択法と等価である。

(4)窓スライド式2操作選択法



幅 n の窓を文字単位で単位移動時間 $\frac{T}{n}$ で高速にスキャンさせる。目的の文字が窓に入っている T の時間内に第1クリックを行い、第2クリックでその窓の中の文字を選択する。第2クリック時の文字の単位移動時間は T にする。文字の平均確定時間 \bar{T} は (2) 式に等しく、 $n=\sqrt{N}$ の時 (3) 式で最小となる。 \sqrt{N} が整数とならない場合、 $\lfloor \sqrt{N} \rfloor$ または $\lceil \sqrt{N} \rceil$ のいずれか (2) 式が小さくなる方を選べば $\frac{N}{n}$ が整数でなくとも良いという点が、(1) や (3) の2操作選択法と多少異なる。例えば、 $N=65$ の時は $n=8$ とすると $\bar{T}=8.06T$ である。

2章に示したダブルクリック操作(クリックホールド操作も同様)はこの方法を用いた。ただし、窓を明示せず、窓の先頭位置を反転表示した。

なお、第1クリックが正しく行われれば、第2クリックは n 文字以内に行われるはずであるから、もしも n 文字以内に第2クリックが行われなかった場合は、第1クリックを取り消し、第1クリックの操作からやりなおすようにした。

(5) k 操作選択法

(3) m 分割2操作選択法を k 操作選択に拡張した方法について考える。まず、 N 個の文字を m_1 個のブロックに等分割し、第1クリックでそのいずれかを選択する。選択したブロックを更に m_2 個のブロックに等分割し、以下同様に分割を繰り返し、 m_k 個の文字から1文字を選択し終了する³。

$$N = \prod_{i=1}^k m_i \quad (4)$$

である。文字の平均確定時間は

$$\bar{T} = \frac{T}{2} \sum_{i=1}^k m_i \quad (4)$$

で、 $\frac{\partial \bar{T}}{\partial m_i} = 0$ より、 \bar{T} は $m_i = N^{\frac{1}{k}}$ の時

$$\bar{T}_{\min} = \frac{k}{2} N^{\frac{1}{k}} T \quad (6)$$

で最小となる。 k に関しては、 $\frac{\partial \bar{T}_{\min}}{\partial k} = 0$ より \bar{T}_{\min} は

$$k = \ln N \quad (7)$$

の時最小となる。

$N=65$ の場合について(6)式を計算すると表1のようになる。

k	\bar{T}_{\min}
1	32.5 T
2	8.06 T
3	6.03 T
4	5.68 T
5	5.76 T

表1. 選択回数 k と最短平均所要時間 \bar{T}_{\min}
(T はスキャンの単位移動時間)

(7)式より、 $k=4.17$ で \bar{T}_{\min} が最小となる。実際上は、 k や m_i の計算値は整数でないため、この時間で文字を確定することはできない。

³ 文献 [7] の2階層文字盤が本稿の4操作に対応する。

同様に (4) 窓スライド式 2 操作選択法を k 操作選択に拡張することができる。まず、 N 文字に対し、幅 n_1 の窓を単位移動時間 $\frac{T}{n_1}$ でスキャンし、第 1 クリックで n_1 個の文字を確定する。次に、第 1 クリックで確定した n_1 個の文字に対し、幅 n_2 の窓を単位移動時間 $\frac{T}{n_2}$ でスキャンし、第 2 クリックで n_2 個の文字を確定する。以下同様に繰返し、第 k クリックで n_{k-1} 個の文字から $n_k=1$ 個の文字を選択し、終了する。次に、例のみ示す。 $N=65$, $k=4$ とし、窓の幅を $n_1=22$, $n_2=7$, $n_3=3$, $n_4=1$ とすれば、 $\bar{T}=5.72 T$ となり、ほぼ表 1 の平均確定時間を実現できる。

4. 既存ソフトの活用

専用インタフェースを用意する方法の意図は既存ソフトを有効に活用することにあることは既に述べた。そこで、汎用⁴ アプリケーションプログラムと組み合わせて動作を確認した例を表 2 に示す。利用者による評価はまだ行っていない。

用途	ソフトの名称	備考
かな漢字変換	VJE-β V2.5	VJE-β V3.0では動作しない。FM16βではVJE-α(VJE-β V2.5も可)で動作する。
ワープロ	アシストワード	アシストレターでは動作しない。FM16βではEWで動作する。
表計算 データベース テキストエディタ	アシストカルク アシストカード RED2	VZでは一部の操作が動作しない。
通信	WTERM	入力速度が遅くオンラインでの操作が適さない場合には、自動通信機能を活用するとよい。

表 2 アプリケーションプログラムとの組み合わせ

かな漢字変換については、「かなモードで起動」「先読み一括変換」「スペースキーで変換」「改行キーで確定」「句読点で変換」「選択モードは使わない」などの機能を設定しておく、少ない操作で日本語入力を行うことができる。「単語

⁴ 「障害者専用でない」という程度の意味。

登録」もできる。登録する読みの入力がシステム行で競合し、キーボードの表示が優先されるため見えないが、操作を正しく行えば正常に登録できた。

5. おわりに

スキャン方式のキーボードエミュレーションプログラムを開発した。キーボードをシステム行に表示することで、アプリケーションプログラムの表示との競合を避けた。

各種のアプリケーションプログラムと組み合わせて動作を確認した。これらのプログラムは、重要な操作が必ずしも少ない入力操作で実行できるようにはなっていない。例えば、前掲のワープロソフトでは、印刷に「ESC P P」の 3 文字の入力が必要であった。このような操作を簡略にするにはキーボードに一連の入力文字を登録できるようにすれば良い。今後改善を試みたい。

3 章では、操作回数を増やして入力に要する時間を短縮する方法について検討した。しかし、この計算には、操作に対する利用者のストレスは考慮されていない。また、連続操作がうまくできるかどうかにも利用者によって異なる。その他、上記の計算には現れない要素が多くある。実際は、これらの点を勘案して操作方法を決める必要がある。

汎用アプリケーションプログラムの使いやすさは、専用のコミュニケーションエイドに勝ることはない。一方で、本稿のようなアプローチが障害者と健常者の距離を縮めることになることも疑われない。既存の資源を有効に活用するという立場で、今後も利用者による評価を進めていきたい。

謝辞 システム行を用いるスキャン方式についての基本的なアイデアは久保真一氏(旭光園)によるものです。お礼申し上げます。

文献

- [1] 依田勝, 藤木武夫, 高村昌利: “言語および肢体不自由者のための意志伝達装置について”, 第 1 回リハ工学カンファレンス講演論文集, pp.51-54(1986).

- [2] 相良二朗, 奥英久: “ 肢体障害者 (児) のための日本語コミュニケーション・エイド ”, 信学技報, ET88-8, pp.31-36(1989).
- [3] 伊藤和幸, 数藤康雄: “ MS-DOS 上で稼働する走査入力方式ワープロソフト ”, 第 7 回リハ工学カンファレンス講演論文集 , pp.251-254(1992).
- [4] 松本廣, 新居泰司: “ 上肢運動機能障害児のためのコンピュータ操作環境の改善 (1) ”, 信学技報, ET91-120, pp.69-76(1992).
- [5] 青木久, 政木憲司: “ グラフィックキーボード HA123 の使用方法 ”, 第 7 回リハ工学カンファレンス講演論文集, pp.263-266(1992).
- [6] 久保真一: “ 肢体不自由者へのパソコン活用 ”, 弘前大学教育学部教育実践センター研究員報告書, vol.1, pp.45-54(1993).
- [7] 奥英久, 相良二朗: “ 走査法によるキーボード・エミュレータにおけるキー選択効率の改善 ”, 第 3 回ヒューマン・インタフェース・シンポジウム論文集, pp.73-78(1987).
- [8] 川上博久, 鈴木理司: “ 重度身障者用ソフトウェアにおけるユーザ・インタフェース ”, 第 2 回リハ工学カンファレンス講演論文集 , pp.5-10(1987).