UDC 001.81: 681.3

# Single-Hand Input Scheme for English and Japanese Text

●Masakatsu Sugimoto

*(Manuscript received June 30,1997)*

**This paper introduces a new high-speed input scheme for mobile computers that is operated with a single hand. The new scheme consists of our Single Hand Keys (SHK) keycard, associated ambiguity resolution logic, and other support programs. This input scheme can be used for both English and Japanese text. First we explain the background of our research, then we describe the design and operation of the SHK scheme.**

## 1. Introduction

Desk-top and notebook PCs now have the performance of computers which 20 years ago occupied an entire room. In fact, so much progress has been made that today, notebook PCs weighing less than one kilogram are available.

Up to now, computer users have had to sit in a chair and operate the equipment while looking at a monitor or display unit placed on a desk. Notebook PCs are operated in a similar fashion even though they are small enough and light enough to be carried around.

With the incredible pace of development in electronics technology, we can expect the eventual arrival of miniaturized computers.[1] To identify the potential bottlenecks with miniaturized computers, we can speculate how people will use a computer when the size and weight of its microprocessor, memory, battery, etc., become negligible.

When such a computer is perfected, people will be able to use it wherever they are and whatever they are doing. The usage could then truly be called “mobile”.

The mobile era of computers is just beginning. Unlike desk top use, which has a long history of research and development, mobile use will require much research and development in various areas.[1]

A mobile computer should include the following functions: document preparation, electronic mail, and Web browsing. In the future, mobile computers will be able to do everything that a desk-top PC does plus many other things.

The input and output schemes for mobile computers are important research themes. Input through a full keyboard is reliable and fast. However, because a full keyboard is large and requires operation with both hands, it is not suited for mobile use. Also, it is difficult for people to carry full-keyboard computers around with them.

There are several alternative input methods for mobile use. One is a small full keyboard with tightly packed keys. However, although this reduces the size of the computer, the keyboard is too small for serious typing.

Another method is voice recognition. But this has the disadvantage that, in certain situations, voice input will not be appropriate, for example, it may disturb the proceedings of a meeting.

Handwritten character recognition is also promising. However, the input speed that can be attained by handwritten character recognition is much lower than can be attained by touch typing.

When we considered the above, we wondered whether typing speeds similar to those achieved on a full keyboard were possible using a single-handed scheme. This led to our developing the

Single Hand Keys (SHK) scheme.

Basically, we cut the full keyboard in half and looked for ways to input data with only one hand.

In this paper we propose a new input device, a new way of operation, and support programs that enable high-speed input of English and Japanese text and mouse commands using a single hand.

## 2. Design Targets for the SHK

We designed the SHK for people who are experienced with a full keyboard and use mobile computers for long periods everyday. Of course, beginners must also be able to learn how to use the new scheme.

Two other goals were that the scheme should enable input of all the key codes available on a full keyboard and that it be usable with either hand.

## 3. Input of English Text

We began by designing an input scheme for English text. The keys are arranged in three rows and six columns, so the SHK has only 18 keys. Since there are 26 alphabetic characters and only 18 keys, there is a shortage of keys. The conventional solutions to this problem are as follows:[2)-4)]

1) Shift-key scheme

Each key is assigned two characters, and the user selects the required character using a shift-key.

2) Chord scheme

Several keys are used together to select the required character.

Input speed tends to be slow in solution 1) because of the need for a shift-operation. Also, special skill is needed in solution 2) because it is difficult to press two or more keys simultaneously at high speed.

The SHK uses the two character/one key approach of solution 1), but with no shift operation. This is achieved using ambiguity resolution logic and word prediction logic that is activated by pressing a special Ambiguity Resolution key after a sequence of characters in a word has been input. This key is called the AR key.

To resolve an ambiguity, a support program collects the candidate words for the input and displays them one word at a time. The user then selects the required word by pressing the AR key. In this scheme, therefore, it is important to reduce the number of candidate words.[5), 6)]

In addition to the 26 alphabetic characters, function keys for back spacing, lowercase/uppercase control, etc., are needed. We therefore use 14 keys for alphabetic characters and use the rest as function keys.

The first key we assigned was the AR key, which we located at the home position of the little finger. Details of other design points are given below.

1) Home positions

To minimize finger movement, the home positions were assigned the characters which appear most frequently in ordinary text.

2) Ease of finger movement

We determined that fingers are most easily moved in the sequence of index, middle, ring, and little finger and that the keys above the home positions are easier to operate than the keys below the home positions.

3) Frequencies of alphabetic characters in English text

The assignment is based on the frequencies of appearance in regular English text. We used the frequencies given in Ref. 7), which are as follows: E, T, A, O, N, R, I, S, H, D, L, F, C, M, U, G, Y, P, W, B, V, K, X, J, Q, and Z.

Because about 40 percent of all the characters in English are vowels and because it was clear that vowels play an important role in Romanized Japanese text input, we decided to locate the vowels (A, E, I, O, U) at the home positions or at the keys for the index finger.

We assigned independent keys for E and A because these are the vowels which appear most frequently.

4) Easy typing of common sequences

Because the AR key is at the home position for the little finger, sequences are ended with a key stroke at the right. Therefore, to establish a predominantly left-right rhythm for character sequences, keys are arranged so that common sequences such as the “io” found in “station” and “automation” are input using keys in a left-right sequence.

5) Balance of left-right finger movements

The balance of left-right finger movements is good when the AR key is pressed after pressing a far left key in the middle row.

6) Minimizing the number of candidate words

Next, we needed to allocate the pairs of alphabetic characters to the keys in a way that minimizes the number of candidate English words. The final arrangement is shown in **Fig. 1**.

The home positions are the E, H/I, S/O, and AR keys in the middle row. The index finger covers the P/N, A, U/D, G/T, E, and X/F keys; the middle finger covers the C/R, H/I, and Y/M keys; the ring finger covers the Z/K, S/O, and V/L keys; and the little finger covers the W/J, AR, Q/B, UpC, TC, and SP keys.

If we had chosen the shift-key scheme, each key would be assigned one high-frequency character and one low-frequency character. In the SHK, however, two home position keys are each assigned two high-frequency characters, i.e., O and S, and I and H.

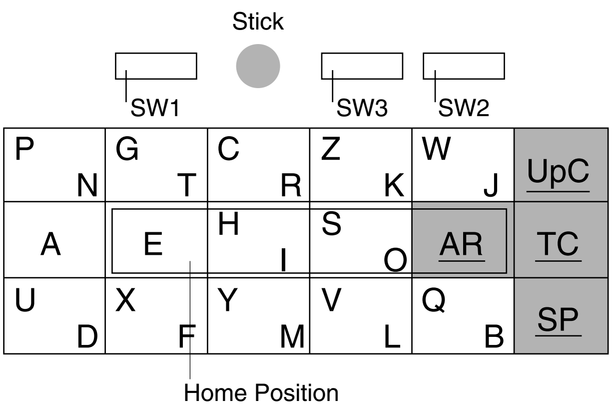


Fig.1– Arrangement of keys, stick, and switches of the SHK.

Some examples of candidate words are shown in **Table 1**.

**Table 2** shows examples of candidate words which were selected from Webster's Elementary Dictionary and an English text book used in high

Table 1

| Multiplicity | No. of chars in a word | Set of candidate words |
|---|---|---|
| 2 | 2 | [do, us] |
| | 3 | [the, tie] |
| | 4 | [they, them] |
| | 5 | [cough, rough] |
| | 6 | [accent, accept] |
| | 7 | [cunning, running] |
| | 8 | [location, vocation] |
| 3 | 3 | [can, cap, ran] |
| | 4 | [cone, cope, rope] |

Table 2

| L \ M | 2 | 3 | 4 or more | Sum |
|---|---|---|---|---|
| 2 | 3 | 0 | 0 | 3 |
| 3 | 30 | 4 | 0 | 34 |
| 4 | 46 | 7 | 0 | 53 |
| 5 | 17 | 0 | 0 | 17 |
| 6 | 5 | 0 | 0 | 5 |
| 7 | 1 | 0 | 0 | 1 |
| 8 | 1 | 0 | 0 | 1 |
| 9 or more | 0 | 0 | 0 | 0 |
| Sum | 103 | 11 | 0 | 114 |

L: Number of characters in a word
M: Number of candidate words

schools in the USA. There were a total of about 6,000 words in the vocabulary.

**Table 3** shows the role of each function key.

The TC key is used for changing between text input, number input, and symbol input modes.

We built a prototype front-end processor which realized this key arrangement. Then, we collected data about the ease of word input and other factors.

We formally demonstrated the input method of the SHK at the CHI'96 international conference in April 1996.

We will now describe some of the features of the SHK for English text input.

1) Prototype SHK support programs

The main features of the SHK support programs for English text input are as follows:

Windows95/NT compatibility
Front-end processor
Word prediction and word completion function
Internal dictionary of about 10,000 words
Utility programs for expanding dictionary entries

2) Time needed to learn SHK typing

Because the SHK has much fewer keys than an ordinary full keyboard, it takes less time to learn how to use it.

Generally, a new user will be able to use the SHK at a good speed after practicing for about two hours a day for 10 days.

3) Practicing system

We have written a Java program for practicing SHK typing that indicates the user's typing speed.

4) Input speed

Once the user becomes familiar with the SHK, input speeds of 40 to 60 words per minute are possible, which is comparable to speeds achieved with a full keyboard.

This high input speed is possible because pressing the AR key at the end of a word usually gives only one candidate word and because a space character is post-fixed to the word automatically.

5) Frequencies of key strokes

**Figure 2** shows the predicted frequencies of key strokes for each key. The data was taken from statistics on frequencies of alphabetic characters in English text. The text is about computers and has about 11,100 alphabetic characters.

## 4. SHK keycard

When we started our research for the SHK, we were using a keypad that was operated on a desk. However, we eventually realized that we could place all the keys on a small, thin device that the user could hold and operate with a single hand. This led to our developing the SHK keycard. **Figure 3** shows an SHK keycard.

The user holds the SHK keycard as shown in **Fig. 4**; in this figure, the fingers are at the home positions. We named this way of operation "Castanets

Table 3

| Function key | Roles of the key |
|---|---|
| AR | Ambiguity resolution or word prediction |
| UpC | Upper case/lower case control<br>Becomes a back space key during word input. |
| TC | Changes the input mode : numeric, symbolic, etc. |
| SP | Space character |

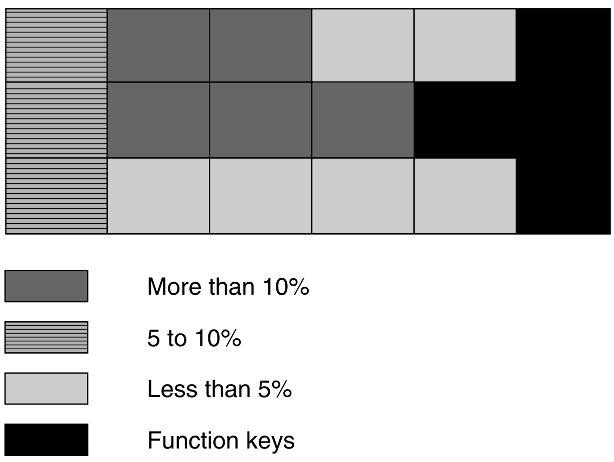


Fig.2– Frequencies of key strokes for English text.

Fig.3– SHK keycard.

Fig.4– SHK keycard: Castanets Operation

Operation” because this is how castanets are held.

We carefully designed details such as the height of each key, the distance between keys, and the shape of the keys so that input operations are as natural as possible.[8), 9)]

Also, we included mouse functions on the keycard, so the user can move a cursor and click without needing to use the other hand or put down the SHK.

We demonstrated the castanets operation at the CHI’96. This was the world’s first demonstration of high-speed text input with one hand. (A company called Handkey of the U.S. has recently marketed a single-hand input device called Twiddler that uses a chord scheme.)

Although the SHK keycard is an important element in the SHK, the user can also practice and perform SHK typing using the left half of a full keyboard.

## 5. Input of Japanese text

We have developed a Japanese input scheme for the SHK keycard which is similar to the one used for English input.

The main difference between English text and Japanese text is that there is usually no separating character or symbol between consecutive words in Japanese text.

In general, Japanese text consists of combinations of Kanji and Kana strings. A Kanji is a Chinese character and a Kana is a phonetic character. For each Kanji string, there is a corresponding Kana string, so there is a Kana string for every piece of Japanese text. However, the corresponding relation between a Kanji string and a Kana string is not one-to-one, but many-to-one; that is, many Kanji have the same pronunciation. Therefore, when translating a Kana string, there are multiple Kanji candidates.

Two typical ways of inputting Kana strings are as follows.

1) Use a full Kana keyboard whose shape and number of keys are similar to those of a conventional keyboard. The user inputs one Kana character per key stroke.
2) Use a conventional full keyboard and input Kana characters using alphabetic characters. (A Kana character is represented by one to three alphabetic characters.)

The SHK uses alphabetic representation for Kana characters. The alphabetic arrangement is the one used for English text, so the user only needs to learn one alphabetic arrangement.

We call the SHK’s alphabetic representation “compressed representation” because some pairs of Kana alphabetic characters are treated as the same character. For example, “が (“ga”)” and “た (“ta”),” are treated as the same, because “T” and “G” are assigned to the same key.

Romanized Japanese text is mostly an alternating pattern of a consonant and one or two vowels. As we mentioned before, because of their high frequency the vowels are assigned to the most easily accessed keys.[9),10)]

Because we usually do not put a separating space between words in Japanese text, we have developed software that allows the user to press the AR key at the end of a word, idiom, or phrase.

In conventional processors, a string of alphabetic characters is converted to a Kana string. Then the Kana string is translated into a combination of Kanji and Kana. In the SHK, however, a string of compressed alphabetic characters is directly translated into the corresponding combination of Kanji and Kana.

There are two ways to do this:

1) Control of the order in which candidate words are displayed

Conventional Kana to Kanji translation systems map a string of Kana to different representations in Japanese text. Since the SHK uses a compressed alphabetic string, the number of candidate words for the SHK tends to be larger than that of conventional systems.

For example, the compressed alphabetic string for "tenka" maps to two uncompressed strings: "tenka" and "genka". For "tenka", the corresponding Kanji strings are 天下, 添加, 点火, 転化, and 転嫁. For "genka", the corresponding Kanji string are 原価, 減価, 現下, 言下, 元価, and 弦歌.

We use a personal vocabulary space to reduce the numbers of candidate words as follows. The words that the user has input during the last few months are automatically saved in this personal space. When a compressed alphabetic string is input, the software first offers the user the candidate words in the personal space and only refers to the large dictionary space if the personal space does not contain the required word. If the user has recently used the required word, this process avoids the need to look through the long list of candidates words in the dictionary space.

2) Automatic collection and prediction of long expressions

In the above translation process, we are searching for a word, idiom, or phrase through a keyword in the compressed representation. We have extended this concept and introduced a function for automatic collection and prediction of long expressions. This function corresponds to the word completion or word prediction performed in text processing for English text.

Two main features of Japanese text input in the SHK are:

1) Prototype support programs

These include:

Programs for compatibility with Windows95/NT,

a front-end processor that is IME (Input Method Editor) compatible, and

an internal dictionary of about 100,000 words.

2) Input speed

Users will quickly be able to input 40 to 60 words of Japanese text per minute, which is comparable to speeds that can be achieved by touch typing on a conventional full keyboard.

3) Frequencies of key strokes

**Figure 5** shows the frequencies of key strokes for Japanese text input using the SHK

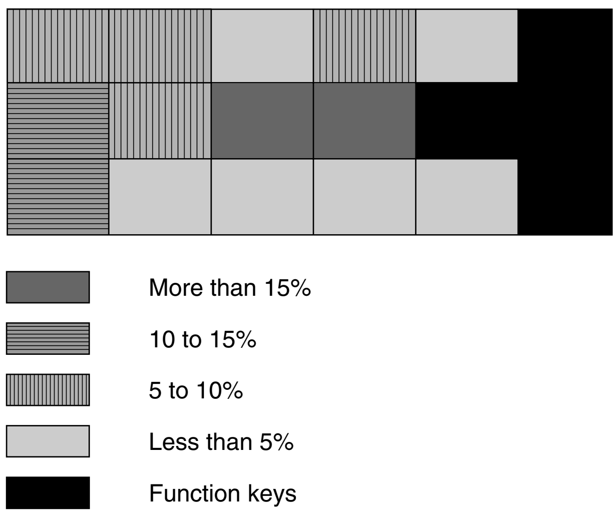


Fig.5– Frequencies of key strokes for compressed representation of Japanese text.

arrangement. The data is for eight of the Tenseijingo columns of the Asahi Newspaper. When converted to alphabetic representation, the columns have a total of about 12,700 alphabetic characters.

The figure shows that more than 80% of the strokes are for the keys in the home positions and the keys operated by the index finger. Therefore, the key arrangement of the SHK is suitable for both English and Japanese text.

4) No effect on skill in using conventional key arrangement

One may wonder whether the process of learning to use the SHK will undo the learner's skill with the conventional QWERTY arrangement. However, this does not happen because, like a bicycle and an automobile, the two input devices are sufficiently different from each other that there is no interference between the skills needed to operate them.

## 6. Conclusions

Our SHK keycard enables high-speed input of English and Japanese text with a single hand. The input speed is comparable to the speed of touch typing on a conventional keyboard with both hands and is made possible through the key arrangement and support programs such as a program for ambiguity resolution.

For English text we use ambiguity resolution applied word by word. For Japanese text we use ambiguity resolution applied at the end of a word, idiom, or phrase.

To make input easier, the SHK has a personal vocabulary space and a learning and prediction function for long expressions.

The SHK can be used in a standing, sitting, or reclining position and leaves one hand completely free. It is therefore highly suited to mobile use and will also be of great benefit to certain physically handicapped persons.

We will continue our research to make the support software more usable and develop suitable applications for the SHK.

## References

1) Smith, N.P.: Computing in Japan: From Cocoon to Competition. IEEE Computer, pp. 26-33, March 1997.
2) Darragh, J. and Witten, I.: The Reactive Keyboard, Cambridge, 1992.
3) Endfield, C.: Portable Word-Processor, US Patent, Appl. No.188, 571.
4) Matias, E. et al.: Half-QWERTY: Typing with One Hand Using Your Two-hand Skills. Conference Companion, CHI'94, pp.51-52.
5) Sugimoto, M.: Keyboard with ambiguity resolution logic. Filed at US Patent and Trademark Office, #383, 781, January 31, 1995.
6) Sugimoto, M. and Takahashi, K.: SHK: Single Hand Key Card for Mobile Devices. Conference Companion, CHI'96, pp.7-8.
7) Beeching, W.: Century of the Typewriter. Heinemann, 1947.
8) Kanaya, Y.: Operation Principles and Logic for Key-arrangement of a Single Hand Touch-typing Keys. (in Japanese), Workshop on Human Interface, Information Processing Society of Japan, 40-8, 1992.
9) Takahashi, K. and Sugimoto, M.: Single-hand Input Keyboard. Filed at US Patent and Trademark Office, 08/614,855, Filed March 12, 1996.
10) Sugimoto, M.: Input Scheme for Japanese Text with SHK Keycard. (In Japanese), Workshop on Mobile Computing, Information Processing Society of Japan, May 30, 1-1, 1997.

**Masakatsu Sugimoto** received the B.S., M.S., and Dr. degrees in Electronic Engineering from the University of Tokyo, in 1965, 1967, and 1970, respectively. He has been working for Fujitsu for about 25 years. He has written papers in various fields, for example, logic design automation systems, language directed computer design, software engineering, machine translation systems, the Japanese fifth generation computer system, and knowledge bases. His current research interest is mobile computers. He is a member of the IPSJ, JSAI, JSSST, IEEE, and ACM.