

KANJI MENTOR  
Version 1.0  
Re-released 95/09/26.  
An application for the HP48SX calculator.  
207498 bytes  
# 78BEh checksum (directory object only)

```
-----+
Please send your comments, gripes, spare cash, to:
James Unterburger
7534 SE 34th Ave
Portland, OR 97202
```

```
e-mail:  james_unterburger@mentorg.com
-----+
```

Prerequisites:

- 1) Two 128K RAM Cards, installed and configured as SYSRAM (MERGED RAM).
- 2) At least approximately 210Kbytes free memory

GETTING STARTED

Download the KANJI.DIR file off the disk into an object named 'KANJI' on the HP48SX. This object can be placed anywhere into the HP48's directory structure. When you press the VAR key, you will see a directory named 'KANJI' in the installation directory. Press the VAR key, then press the KANJI softkey to switch to the KANJI directory. Now press the CST key and you are ready to begin.

KANJI MENTOR AND THE JAPANESE LANGUAGE

To be able to use Kanji Mentor, you must first understand the way the kanji character information is arranged. Each of the 2000 characters comprises the following information:

- a 24x24 pixel graphic representation of the character (same as JIS 24 dot dot matrix printer standard)
- a unique character number in the range  $1 \leq \text{number} \leq 2000$ ; the number of the character corresponds to one of two numbering schemes of the two main reference works used in the creation of this application the two reference works are:
  - 1) Naganuma Naoe -- a set of 1500 flash cards (500 to a box, sold separately), Japan Publications Trading Co., Ltd., Tokyo, Japan, 1980
  - 2) O'Neill, P.G., Essential Kanji, John Weatherhill, New York, New York, 1973
- a list of readings for each character (yomi-kata)

In general, a Japanese kanji character will have readings from two distinct phonetic categories. ON-yomi is the Chinese-based pronunciation of the kanji, and kun-yomi is the original Japanese way of reading the character. For the purposes of the Kanji Mentor application, ON-readings are Romanized in UPPER CASE (e.g., "GYAKU" and "EN"), and kun-readings are Romanized in lower case (e.g., "yo.mu" and "hitsuji"). Furthermore, kun-yomi, which are essentially used to form words for which the word ending may change to meet certain grammatical requirements, may contain a separator character to indicate the point at which the character reading ends and the rest of the word, normally written in hiragana, begins.

Thus, for example the word "to read" in Japanese is "yo.mu", the "." indicating that the "yo" portion of the word is contained in the character itself, while the "mu" portion would be written in hiragana. (Note also that the word ending "mu" could change depending on the use of the word. Kanji Mentor attempts only to provide the dictionary entries of each character's readings, not every possible usage.) The position of this separator character is relevant and must be matched exactly when the "Searching" method is set to "try exact". More about the application's operating parameters later. The Romanization system used in Kanji Mentor comprises the following syllables:

\*\*\*\*\*

```

a   i   u   e   o
ka  ki  ku  ke  ko
sa  shi su  se  so
ta  chi tsu te  to
na  ni  nu  ne  no
ha  hi  fu  he  ho
ma  mi  mu  me  mo
ya      yu      yo
ra  ri  ru  re  ro
wa   n
ga  gi  gu  ge  go
kya      kyu      kyo
gya      gyu      gyo
za  ji  zu  ze  zo
sha      shu      sho
ja      ju      jo
da  zi      de  do
cha      chu      cho
nya      nyu      nyo
ba  bi  bu  be  bo
pa  pi  pu  pe  po
hya      hyu      hyo
bya      byu      byo
pya      pyu      pyo
mya      myu      myo
rya      ryu      ryo

```

s.s t.t <== used where the small "tsu" is directly following the kanji  
pp tt <== used where there is a small "tsu" embedded in the reading  
\*\*\*\*\*

Note that not all of them have necessarily been used.

#### OPERATING PROCEDURES

Once you have entered the Kanji Mentor application directory and displayed the CST custom menu (see GETTING STARTED above), there are 6 choices available in this custom menu (5, and a blank label, are on the first page, the 6th is on the second page of this CST menu). The normal sequence of operation from this menu is

- 1) SETUP -- this saves the flags and restores the last Kanji Mentor settings and enters the SETUP menu, described below
- 2) Perform whatever tasks you want
- 3) EXIT -- this saves the application's settings and restores the flags to the values they had when SETUP was executed

Here are the descriptions of each of the choices available:

FIND -- used when you know the reading and want to see all the characters with that reading; the characters are displayed in the VLIST utility

VLIST -- takes a list of integers in the range  $1 \leq \text{integer} \leq 2000$  and enters the multi-character viewing utility, which displays each character according to the current character numbering scheme (more about this later)

VIEW1 -- takes one integer in the range  $1 \leq \text{integer} \leq 2000$  and displays that character and its readings according to the current character numbering scheme

SETUP -- enters the application parameters setup menu

EXIT -- exits the application and restores the flags to their pre-SETUP state

ABOUT -- displays a nice message in friendly letters (version number, etc.)  
(Note: this is on page 2 of the CST menu.)

#### APPLICATION OPERATING PARAMETERS

The SETUP command is used to enter the Kanji Mentor application. First it briefly displays information about the application. Then it saves the current flags settings so the EXIT command can restore them later, if SETUP has not already been executed. The PICT graphics object is resized to 131x64 pixels and cleared. The binary word size is set to 64. Then the Kanji Setup menu is displayed. It is through this that you set the application's operating parameters.

NUMB -- toggles the current character numbering scheme between "Naganuma" and "O'Neill"; this setting determines which number to associate with which character; in Naganuma, for which there are only 1500 characters, characters numbered 1501 through 2000 were obtained from the O'Neill reference: thus character number 1501 (Naganuma) is the first O'Neill character which is NOT in Naganuma; since there are some characters in Naganuma which are not in O'Neill (there are 30 such characters), there must be an equal number of characters in O'Neill which could not be included in Kanji Mentor; thus, when the O'Neill numbering scheme is active and a non-resident character is requested, a bogus pixel pattern is displayed instead

SRCH -- toggles the searching method between "try exact" and "find all"; this affects the operation of the FIND routine: when set to "try exact", the reading you enter is matched against all the readings in the dictionary for an exact match (including the "." separator if finding a kun-reading) if no matching reading is found, you are prompted on whether to continue or not; the "find all" setting will find all readings, ON- or kun-, which contain the input string, disregarding any "." separator

FIND -- toggles between finding, using the FIND routine, all characters with any one reading, or all characters which have two readings (intersection of sets)

GO -- exits the Kanji setup menu and displays the main CST menu

#### OPERATION IN DETAIL

The main CST menu contains three major routines, FIND, VLIST, and VIEW1.

##### --FIND--

The FIND routine prompts for a character reading and attempts to find any and all characters which have that reading. At the "Enter Reading..."

prompt, type in a valid character reading (within the constraints of the Romanization method). Typing a non-Japanese or otherwise unparseable reading will result in an error message "Invalid reading:" along with the lower case version of what you typed in. If all goes well, the FIND routine will, depending on the current "Searching" parameter (described above in the SETUP routine explanation), try an exact match -- UPPER CASE for ON- and lower case for kun-readings, including the "." separator character; in the case that no match is found, you will be prompted if you wish to continue and attempt to find any match at all. This is essentially the same as having the "Searching" parameter set to "find all". In this case, all readings are searched, a match being found if any reading contains, in whole or in part, the input string. This search is conducted without regard to case or "." separator location. Thus, for example, the input "TARA" will fail to find an exact match since no ON-reading exists with the exact Romanization "TARA" (since it's not even an ON-reading). If you answer YES to the "Continue?" prompt by pressing the "YES" softkey (assuming that the Searching method is "try exact"; if not, then the FIND routine will produce the following directly without user intervention), it will match "TARA" with the readings "atara.shii", "hatara.ku", and "ta.rasu" instead.

While this "find all" search is proceeding, you will see the messages "Finding all matches..." "Found...<N>" "Press a key to interrupt search". While the search is proceeding the <N> will reflect the number of matching readings found so far. You may interrupt the search by pressing any key (except ON/ATTN). This will cause the searches to cease and will proceed directly to the reading extraction routine.

In the reading extraction routine, each reading is briefly displayed along with a "Press a key to interrupt" message. Pressing any key (except ON/ATTN) will cause the reading extraction to cease, and the program will proceed directly to the List Editor.

The List Editor is a special environment where you can select and unselect readings for which you wish to see the corresponding kanji character(s). In this environment, only a few keys have an action associated with them (remember not to press the ON/ATTN key, since that will abort the application). When you exit the List Editor with a press of the -> (right arrow) key, you will enter the VLIST routine with all the characters which have any of the readings which were selected (highlighted in reverse video) when you pressed the right arrow key. All readings are selected when the List Editor is entered. The active keys in the List Editor are:

```
-- UP ARROW
    moves the cursor up one item

-- DOWN ARROW
    moves cursor down one item

-- <RIGHT SHIFT> UP ARROW
    home (moves cursor to item number 1)

-- <RIGHT SHIFT> DOWN ARROW
    end (moves cursor to last item in list)

-- <LEFT SHIFT> UP ARROW
    page up (moves cursor up 8 items)
```

```

-- <LEFT SHIFT> DOWN ARROW
   page down (moves cursor down 8 items)

-- ENTER
   selects (or unselects, if it is already selected) the item pointed
   to by the cursor

-- +/-
   same as ENTER

-- 1/X
   inverts, that is, selects all unselected items and unselects all
   selected items

-- digits 1 through 8
   moves the cursor to the corresponding line of the current screen

-- CLR (<RIGHT SHIFT> back-arrow)
   exits the List Editor and abandons program execution

-- ->
   exits the List Editor and enters the VLIST routine with all characters
   with readings which were selected

```

In addition to the readings, the List Editor also displays information about the current number of selections and the currently viewed page of selections (useful if there have been a lot of readings found).

If the "Finding" application parameter is set to "2 readings" in the SETUP menu, then the FIND routine will, after you have chosen readings from the List Editor and pressed the -> key (if "find all" was in effect), prompt for a second reading and begin this process over again. Upon the second match, only the characters having BOTH readings will be sent to the VLIST command. In the event that the second match set is empty, all readings from the first set will be used.

--VLIST--

The VLIST routine, intended primarily as the character viewer which is called from FIND, can also be operated from the command line. It takes a list of integers representing character numbers (in the current character numbering scheme) and enters an interactive browser. This browser displays information about each character one character at a time, in the form of a "slide" which displays the character number, the character, the slide number for the current VLIST session, and optionally the readings for the character. Like the List Editor, the VLIST routine is a special environment where only a few keys have actions. The active keys in the VLIST environment are:

```

-- ->
   moves to the next slide (increasing numbers); wraps around if necessary

-- NEXT
   same as ->

-- <-
   moves to the previous slide (decreasing numbers); wraps around if

```

necessary

- PREV  
same as <-
  
- <RIGHT SHIFT> ->  
displays the last slide in the list
  
- <RIGHT SHIFT> <-  
displays the first slide in the list
  
- ENTER  
displays all the readings (yomi-kata) for the currently displayed character; furthermore, saves the current slide (including the readings displayed) as graphics object, making the next viewing of the slide relatively quick; saved slides are available only in the current VLIST session; also, because of the memory usage involved, there is an upper limit to the number of slides you can save and speed up this way
  
- OFF  
turns the HP48SX off without exiting the VLIST environment; at powerup the application resumes where it left off (assuming that no control alarm has come due)
  
- digits 1 through 9  
enters a command line prompt which asks for a slide number; finish entering the number and press ENTER to jump to that slide; it is possible to enter any command at this prompt, so make sure that what you enter results in exactly one integer being pushed onto the stack
  
- + (PLUS key)  
enters a command line prompt which asks you for a character number; enter a valid character number (1<=number<=2000) and press ENTER; the VIEW1 utility is called to display this character (which may or may not be in the list that VLIST is viewing); press any key (except ON/ATTN) to return to the VLIST routine after viewing this "extra" character; it is possible to enter any command at this prompt, so make sure that what you enter results in exactly one integer in the range 1<=integer<=2000 being pushed onto the stack
  
- CLR (<RIGHT SHIFT> back-arrow)  
exits the VLIST routine

NOTE: A control alarm coming due during execution of VLIST will cause it to exit prematurely. Try to make sure that no control alarms will come due if you're using VLIST. Message alarms don't seem to cause any trouble.

--VIEW1--

The VIEW1 routine takes one integer in the range 1<=integer<=2000 and displays that character according to the current numbering scheme, along with the readings of that character. Pressing any key will exit this routine. This routine is called from the VLIST routine if the "+" (PLUS key) is pressed and the prompt answered.

\*\*\*\*\* LEGAL STUFF \*\*\*\*\*

The software programs described in this document are non-confidential, non-proprietary products of James Unterburger. This version 1.0 is offered to the public as freeware, and as such may be freely transferred, given, and used by anyone, provided that I get credit for the implementation of any routines and/or data which remain, in whole or in part, in the distributed versions. This program may not be sold for any purpose or for any reason without my permission (which, quite frankly, will cost you!) Furthermore, I reserve any rights not explicitly given away in this paragraph.

I make no warranty of any kind with regard to this material including, but not limited to, the implied warranties or merchantability and fitness for a particular purpose. Same goes for the programs described here.

I shall not be liable for any incidental, indirect, special, or consequential damages whatsoever (including but not limited to international incidents) arising out of or related to this document, the information it contains, and the software it purports to describe.