Keyboard Clutter, Keystroke Counts & Calculator Menus

Jake Schwartz

Following the HHC2007 HP Calculator Conference in San Diego last September[1], I was considering the feature differences between the hp33s and HP35S, and also the subject of Richard Nelson's talk titled "How Many Keys Should A Personal Calculator Have?". Originally, one of my biggest beefs with the 33s (aside from the demotion of the HP-signature double-wide ENTER key) was the change in the soft-key "menuing" system from its predecessor, the HP32S, to a numbered, displayed menu list. The fact that this numbered-menu scheme was maintained in the HP35S as well was a major discussion point in my own conference presentation. It was (and remains) my contention that it is intuitively more convenient to press a key directly under its menu label rather than to have to first locate and then press the numeric key corresponding to the numbered menu choice up in the display. After the conference, with thoughts of keyboards and menus swirling in my head, I wondered if it might make sense to attempt to quantify a handful of keyboard attributes which have come up in discussion over these past 35 years since the HP35A was introduced.

Opposing Keyboard "Forces"

In considering keyboards, there seems to be a handful of attributes which sometimes work against and other times with each other to affect calculator ease of use. Some of these are (1) the overall number of keyboard keys, (2) the number of shift keys, (3) the number of functions per key and (4) the number of soft-key menus. First, let us establish some Cardinal Calculator Rules for the sake of this discussion:

I. **The more functions a calculator has, the better.**

II. **The more information is available in the display, the better.**

III. **The fewer keystrokes required to execute each function, the better.**

IV. **Less keyboard clutter is better.**

Now, the ultimate keystroke saver would be a machine containing an individual button for each function but due to the size constraints for handhelds, there must be compromises. Hewlett-Packard's Classic Series all fit rather nicely in the hand and all contained the same 35 keys despite increasing functionality as later models were devised. The initial solution to adding functionality without adding keys was to employ shift keys which would multiply the number of functions available in the other keys. Now, since keys with primary plus shifted functions require multiple

[1] Lots of information on HHC2007 is still available at http://holyjoe.net/hhc2007 .

function identifiers as opposed to a single one when no shifted functions are present, the clutter on the keyboard increases. Let us define *keyboard clutter* as the ratio of the total number of function identifiers on a keyboard to the total number of keyboard keys. Therefore a keyboard with no shifts and one function per key would have a clutter value of exactly 1.0 by definition. Both the HP35 and HP70 fit this profile, with only a single marking associated with each of the keys on their keyboards. Moving to the HP80 with its single gold shift key plus eleven identified shifted function positions, it's clutter value is (34+11)/35 or 1.286. The HP45, with its single gold shift plus twenty-four marked shifted positions would have a clutter value of (34+24)/35 or 1.657. In the Classic Series machines, the record holder is the HP67 with its three "f", "g" and "h" generic shift keys, permitting up to four functions labelled on each key and a clutter value of 114/35 or 3.257.

Another measure of keyboard effectiveness is the *average keystroke count* per function. This would be computed as the ratio of the total number of keystrokes required to execute all functions divided by the total number of functions. In the ideal machine with exactly one function per key and no shifts, the keystroke count value would also be exactly 1.0. However, no such "perfect" calculator keyboard exists even in the Classic series. If we consider the HP35 or the HP70 models, allegedly with no formal shift keys, both contain other keys which act like shifts since they must be combined with another key to have any meaningful functionality. On the HP35 the "arc" key combines with the SIN, COS and TAN keys to produce the two-key inverse-tan functions. Likewise, on the HP70, the DSP key combined with 0 through 9 or the decimal point sets the display mode and the STO key along with either the "M" or "K" keys saves numeric data into the M or K registers. In both machines, these "functional" shift keys (as opposed to "generic" shifts like the gold key on the HP45 or HP80) would serve no purpose by themselves. As a result, the HP35's 34 primary functions plus three 2-keystroke shifted inverse-trig functions yields an average keystroke count of 40 total keystrokes divided by 37 functions or 1.081. Also, the HP70, with its 33 primary 1-keystroke functions, eleven 2-key DSP functions plus two 2-key STO functions would have an average keystroke count value of (33+22+4)/46 or 1.283. For the Classic Series, the model with the highest average keystroke count turns out to be the HP55, due to its containing the most 3-key function sequences due to its functionality including the GTO combined with a 2-digit program line number 00 through 49. Its total is 2.401. Table 1 shows a comparison of all the Classic-Series machines with respect to their keyboards. For the complete charted data regarding any of the individual machines referenced in this article, please consult http://www.pahhc.org/keyboards.htm.

Now, increasing the total number of functions by merely adding more shift keys produces diminishing returns rather quickly with respect to the enormous amount of keyboard clutter which would ensue. Imagine a fictitious Classic-Series machine

	HP 35	HP 45	HP 55	HP 65	HP 67	HP 70	HP 80
tot # keys	35	35	35	35	35	35	35
# generic shift keys (i.e. "f", "g")	0	1	2	3	3	0	1
# functional shift keys (i.e. "DSP", "STO")	1	3	3	5	4	2	0
tot # functions	37	164	227	184	263	46	55
Keyboard Clutter (# fcns per key)	1.000	1.657	2.343	2.000	3.257	1.000	1.286
Keystroke Count	40	379	545	387	601	59	76
Average Keystroke count per function	1.081	2.311	2.401	2.103	2.285	1.283	1.382

Table 1. HP Classic-Series Machines and Their Keyboards

whose top two entire rows of keys are shifts. This means that (35-10) or 25 keys would each be allowed to have a primary function plus ten additional shifted functions, totalling 25*11 or 275 functions. The question would be as to how the keyboard would allow eleven function identifiers for each key – producing a clutter nightmare. While the keystroke count per function would be a reasonable 525/275 or 1.909, the clutter value computes to an astronomical 275/35 or 7.857. Obviously, another mechanism would be required to manage clutter while continuing to add functionality. One additional solution to the clutter problem is to increase the number of keys (while leaving the number of functions constant). This is essentially what happened with the HP67 and HP97, whereby by increasing the key count from 35 to 56, the number of shift keys was able to be reduced from three to one. As a result, the number of primary functions (not requiring a shift) increased by twenty, the number of 2-key functions went up by ten and the number of 3-key functions dropped by 29. This caused the average keystroke count per function to be reduced roughly 9% from 2.285 to 2.098 and the keyboard clutter to drop a whopping 40% (or more than a function label per key position) from 3.257 to 1.964.

Enter the ALPHA Keyboard

In 1979, the HP41 Series machines were born and they had a relatively simplified keyboard with only gold and ALPHA shift keys. However the number of functions could be astronomical because the ’41 allowed function names to be spelled out. As a result, a “loaded” HP41CX could literally have many thousands of functions but maintain a 39-key keyboard with a manageable clutter value. The downside of this scheme is that keystroke counts jumped incredibly high. For example, to insert the PROMPT function into a program requires [XEQ] [ALPHA] P R O M P T [ALPHA], or a whopping 9 keystrokes. In fact, computing the keyboard metrics on

the HP41C keyboard produces a clutter value of only 2.641 (based on 103 keyboard markings spread over the 39 keys) but yields an average keystroke count of 6.213 (based on 4915 functions requiring a total of 30539 keystrokes). The saving grace of the ’41 is its key-assignment capability so at least if a long-keystroke function is to be used repeatedly, it may be assigned to a key to be pressed in USER mode. Perhaps it was this high-keystroke burden which provided the motivation for the creation of USER mode in the first place.

Soft-Key Menus Come To the Rescue

From the introduction of the clamshell HP18C in the summer of 1986 onward, most HP calculators employed a new mechanism to increase the number of built-in functions – the soft-key menu and top row of (up to six) re-definable soft keys. By placing a menu name on a keyboard, a row of key functions could be represented by a single function position. If a menu required more functions than the number of soft keys in the row, either the rightmost soft key would act as a “MORE” function (on many of the business machines) or there would be a dedicated “NEXT” keyboard key (on the scientifics) which could switch between multiple menu “pages”. Now, this solves the clutter problem fairly nicely, but at the same time slightly complicates the situation with respect to average keystroke counts, since executing a page-1 menu function requires a minimum of three keystrokes if the menu name itself is on a shifted key position. If a function is on page 2 of the menu, then four keystrokes are required. (Still this method provided a marked improvement over the one used in the HP41 series.) So while attempting to increase calculator functionality, it becomes increasingly difficult (if not impossible) to minimize both keyboard clutter and average keystroke count at the same time while leaving the number of physical keyboard keys relatively constant.

In an almost complete reversal from the HP67-to-97 comparison where the number of keys was doubled for the same number of functions, the transition from the HP18C (released 6/86) to the HP17B (released 1/88) halved the key count from the clamshell to the vertical Pioneer layout. In shrinking the keyboard of the 18C’s 72 keys to the 17B’s 37, the dedicated alphabetic keys became buried in an ALPHA menu. Interestingly, while the number of primary functions went down from 62 to 27 and the number of visible shifted functions dropped from 36 to 9, the 17B actually added capabilities such as increasing the number of memories from 4 to 10 and also tacked on 6 additional financial menus (plus the ALPHA menu), increasing that total from 23 to 30. The overall function counts increased from 251 in the 18C to 343 in the 17B, however with most of that being in menus, the overall keystroke counts rose 65% from 492 to 814. Amazingly, the clutter was reduced from 1.431 on the 18C to 1.189 in the 17B, but the average keystroke count per function rose 21% from 1.960 to 2.373.

In the case of the 18C-to-17B transition, the designers must have felt that business and financial users would be turned off more by a cluttered keyboard rather than the necessity to press lots of keys in order to navigate to functionality. While the MATH sub-menu was moved from the MAIN menu to a key onto the shifted keyboard, they stopped there. This contradicts the fact that the lesser Pioneer financial machines, the HP10B and HP14B both contain single-shifted function identifiers on practically every key,. As a result, their clutter values turn out to be relatively high at 1.946 and 1.919, respectively. On the other hand, since the 10B and 14B machines were introduced slightly after the 17B, they may have changed their minds. Personally, I prefer the exact opposite, whereby fewer keystrokes required to do something far outweigh the sacrifice of cluttering up a keyboard. As a result, perhaps Cardinal Calculator Rule IV above should be amended as follows:

IV. Less keyboard clutter is better, but not at the expense of increased keystroke counts.

If tasked with personally redesigning the 17B (or better yet the 17BII+ with RPN), I would have followed in the 10B/14B design and allowed the large number of lower-level menus to be brought out on the keyboard, altogether eliminating the "MAIN" menu (consisting of "FIN", "BUS", "SUM", etc.) and probably also enabling more direct access to other, lower-level menus. Instead of having the top row of keys be unmarked, they could have the various menu functions of the items inside the former main menu. Similar to the scientific Pioneer machines, these top-row keys would only perform as soft-keys when a menu was present. Thus, instead of requiring the soft-key menu to permanently occupy the lower level of

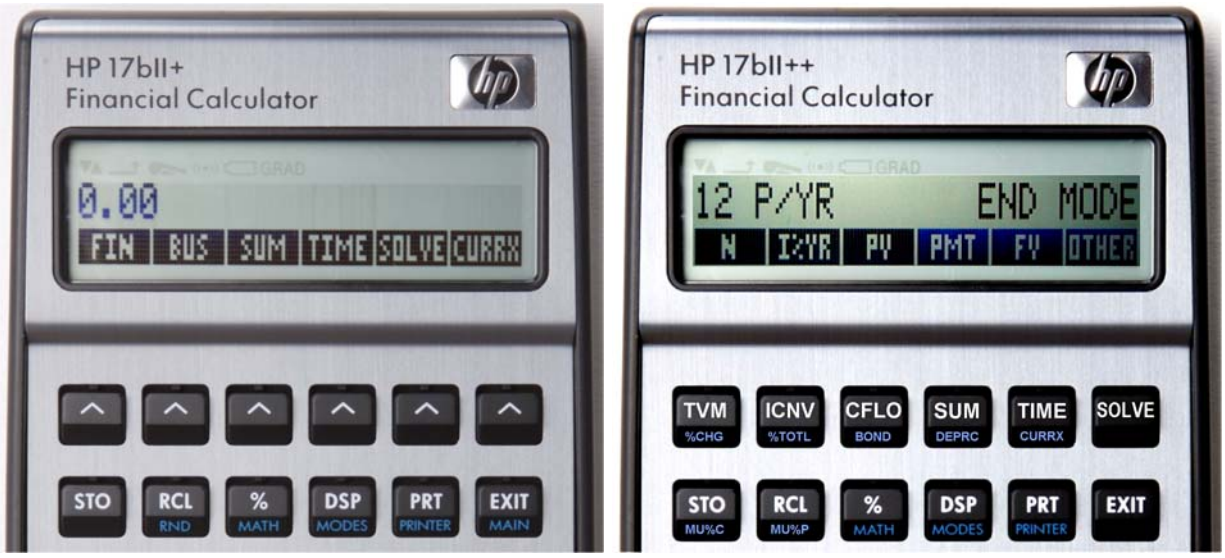

Figure 1. HP17BII+ and a fictitious "17BII++" with the MAIN menu removed and the top-two rows of shifted positions populated with many of the high-level financial menus. Keystroke counts would decrease significantly.

the LCD, it could optionally be turned off via the EXIT key so two numeric values could be seen in the display, like was done in the HP42S. Figure 1 compares the

HP17BII+ to a fictitious “HP17BII++” which incorporates this change. To me, this increases both ease of use and access to information.

Bad Clutter, Good Clutter

In comparing the procession from the Pioneer HP32S (introduced 6/88) to the HP32SII (in 3/91) to the hp33s (in 4/04) and finally to the HP35S (in 7/07), a number of keyboard-related items come to mind. First, when Eric Vogel of HP Corvallis visited our Philadelphia Area HP group in April of 1991, he presented the story of the HP32SII and how it was created to increase the appeal of the original 32S[2]. It apparently had been perceived to be less capable than its inferior sibling, the HP20S, due to the fact that the keyboard of the 20S contained so many more visible functions. What was not understood was that all of the 20S’s functionality was showing on the keyboard, accessible via its two shift keys, whereby the 32S had concealed 276 functions within its sixteen menus. In order to psychologically level the playing field for those prospective buyers who were lining these machines up for visible comparison, the 32SII added a second shift key and was able to appear more complex. This occurred while actually reducing the number of menus from 16 to 14, the number of menu functions from 276 to 125 and simultaneously increasing the number of built-in functions from 732 to 801. This reduction in the menu function count allowed the average keystroke per function to drop from 3.120 to 2.869 but by design, increased keyboard clutter from 2.730 to 3.622. This compares to the HP20S’ clutter figure of 2.919, which was indeed higher than the 32S but lower than the 32SII. So on rare occasions, more clutter turns out to be better.

Now, in progressing from the 32SII to the 33s, the number of keyboard keys was increased from 37 to 48, which permitted more functions to be spread out. So, as some functionality was added, increasing the function count from 801 to 814, the clutter was actually reduced from 3.622 to 3.229 (although personally I feel that it is hard to tell). Also, since the number of menus was able to be decreased from 14 to 13, the average keystroke count also dropped from 2.869 to 2.757.

With the HP35s, key count was scaled back from 48 to 43, which would imply a necessary increase again in key clutter; and this machine followed suit with a value of 3.465 – more than the 33s’ value of 3.229 but still less than the 32SII’s 3.622. Likewise, the average keystroke count per function went back up from 2.757 (in the 33s) to 3.095. With respect to the clutter however, perhaps I am biased, but the 35s appears slightly less cluttered than the 33s, perhaps because each key has only

[2] DVD copies of the Eric Vogel visit to the PAHHC meeting are still available. For more details, see http://www.pahhc.org/video.htm on the web.

one shifted function above it. Also, neatly tucking the other shifted function(s) on the key fronts (like all the machines prior to the Pioneer models) also helps to give the illusion of more empty space separating the key rows. Personally, I feel that the departure from the traditional bevelled keys on all models up to and including the Voyager Series machines over to the flatter, rounded keys of the Pioneer series was a mistake.

Lessons To Be Learned From the HP42S

Returning to the original subject of this article, namely the contention that the change in the menu system from the 32S/32SII to the 33s/35s was a step backward, there are a few things to consider. Starting with the 32S and ending with the 35s, all four machines' menus are the type whereby after a menu key has been pressed, the menu is dismissed. This was a necessity in the 32S and 32SII due to its LCD having only a single line. However, in the subsequent two-line screen models, this should not necessarily have been the case. In a menu which contains a handful of items which need to be used repeatedly (such as the "h", "d", "o" and "b" symbols in the "BASE" menu), it is a burden to have to re-enable the menu each time in order to access the function(s) again. There was a missed opportunity to return to the much-more-flexible HP42-like display. This LCD is a continuous dot matrix through its entire length and width as opposed to being partitioned into 5-by-7 dot matrices for each character, separated by empty space (with the exception of a pixel for a decimal point). An HP42S-like display enables the bottom row to show up to six soft-key menu labels directly above the top row of keyboard keys. (HP still knows this, since they continue to employ this type of LCD and menuing system in the current incarnation of the HP17BII+, as mentioned earlier.) With the 42S-type LCD, we have the bonus of not only being able to share the display between the soft-key menu and a numeric value (and thus be able to keep the menu active between keystrokes), but we also could have the option to turn it off to show two numeric values, such as the 33s and 35s currently do. This is the best of all possible worlds, in my opinion: menus reducing clutter with *persistent* menus reducing keystroke count at the same time.

Over the past several years at the HP calculator conferences, Richard Nelson discussed his technique for counting keystrokes, which might be relevant here. He contends that a keystroke consists of two parts – the first half is locating the key and the second half is pressing the key. As a result, pressing the same key twice in succession eliminates the need to locate the key a second time, which might justify a double press counting as only one and a half keystrokes. Likewise, I contend that in the HP42S-like menus, the step in locating a menu key after reading the menu function in the display is almost eliminated, since the key is directly adjacent to the LCD menu label. As a result, perhaps we could say that the soft-key menu key press only requires three-quarters of a keystroke. This is compared to the full-

keystroke key press on an HP35s, since its menu functions are numbered and require the user to move his eyes away from the LCD to visually locate the correct numeric key on the keypad at the bottom of the keyboard. Based on this, the 142 HP35S menu functions would each lose a quarter keystroke (or 35 keystrokes) from its total. In addition, with the six soft-key menus, the INTG, LOGIC, CLEAR and SUMS menus could each be reduced from two-pages to a single page, lopping off another 16 more keystrokes in total. So, the 545 keystrokes originally required to complete those 142 menu functions would be reduced by 35 +16 or 51, bringing the average keystroke count for menu functions in the 35S down from 545/142 or 3.838 to 494/142 or 3.479. It's a step in the right direction. This difference would be even more pronounced based on keystroke sequences which require repeated use of functions from the same persistent menu. In a sequence using the same menu *n* consecutive times, the menu-access keystrokes are reduced by *n-1* times.

Finally, in order to enable the 42S-like soft-key menus on the 35s, the top row of keys would have to be a conventional row of six. This is a relatively minor keyboard change, as shown in figure 2. Bring on the HP 35s+!!

Figure 2. HP35s and a fictitious "35s+" with the third row of keys moved up to row one so as to enable an HP42S-like LCD and soft-key menus. This would be my "fantasy" 35s.