Impressions of the HP 12C Platinum

Jordi Hidalgo, #1046 (Revised version of the article in *Datafile* V22N3)

"At first sight the Hewlett-Packard 12C Financial Calculator appears to be a real jewel of the calculating art. With its gold, brushed, anodized aluminum highlighting and its liquid crystal display, along with its slim, horizontal design, it represents an aesthetically pleasing jump in human engineering." [1] Whether I find the new Platinum model as elegant as the HP-12C or not is point of little importance, but let me relate to you my impressions after a few days using a production unit of HP's first RPN calculator in four years.

Functions and Bugs

There are two major changes: an algebraic entry mode and four times more memory for programs; but the designers did not make the most of these features. Like most business models, the algebraic mode does not use precedence (1+2×3=9), but there are no parentheses either, so intermediate results need to be stored or written down to be reused.[2] *Arithmetic* mode would therefore be a more appropriate name. RPN is, of course, the default mode! Two new programmable keys toggle between both modes: ALG and RPN, which also clear the stack and the last x. The only new function is x^2 (quite handy in algebraic mode). But, why not include more functions? It could be argued that program instructions continue to be 8-bit long (one 56-bit storage register is converted into seven program lines), which would mean that there can only be 2^8=256 possible instructions, 253 of which are used by the 12C's built-in functions,[3] leaving three codes for RPN, ALG and x^2; but then, how does the 12C Platinum encode GTO 100 through GTO 399? There seems to be no other reason why the longed-for trigonometrics have not been included than HP's will to stick with a well-known function set. Anyway, four times more memory means enough room for Valentin Albillo's trig functions (Datafile V21N1p12) and polynomial root finder (V21N2p35), Tony Hutchins' 99-step Black-Scholes program and Philippe Heilbronn's keystroke-saving calendar routines (V19N4p21) … and even one hundred steps left! Since program memory still consists of 8 program lines initially, one could expect to find at least 63 data storage registers, 56 of which would be converted into 392 steps as needed, leaving 7 registers – six of them being the statistics registers. In fact, the manual states the same on page 93. This is not the case, though: there are only 20 general purpose registers (R_0 through $R_{.9}$), which start converting into program memory when keying in the 310th instruction. These registers are still used for storing cash flow

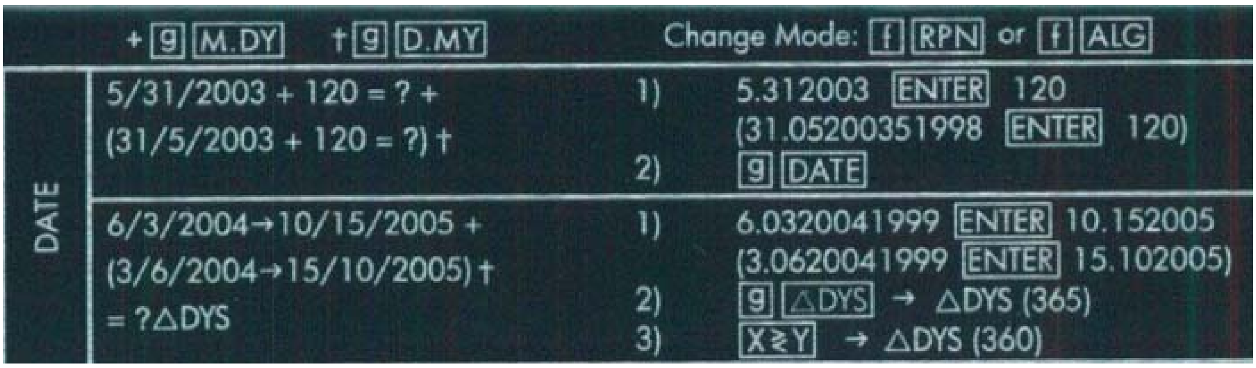
---

1 "The HP-12C: Reflections and Applications" by Edward M. Keefe, V9N2p11 of the PPC Calculator Journal, March 1982.

2 However, the stack handling operations (R↓, X<>Y, CL *x* and LST *x*) also work in ALG mode. For instance, the following keystroke sequence would solve 50 – (38 / 15): 50 ENTER 38 ÷ 15 ENTER X<>Y – X<>Y ENTER (Also: 38 ÷ 15 ENTER 50 – X<>Y ENTER)

3 Refer to "Keycodes and 256 Instruction Codes" by Gregory L. Nelson, PPC Calculator Journal, V9N4p10, May/June/July 1982.

amounts, but there are now 10 more registers for this purpose: CF_{21} through CF_{30}, which can be recalled by *j* n RCL g CFj, but are overwritten by program lines 240 through 309. Storage register arithmetic is still only possible with registers R_0 through R_4. Despite the extra memory, the limited keystroke programming features of the 12C have not been improved with flags, labels, subroutines or indirect addressing – not even with some facilities in program editing. The fact that the source code for the 12C was lost (V20N5p47) and a redesign of the CPU was therefore necessary[4] becomes evident by differences in the accuracy and execution times in some calculations. While the calculator is in general faster (especially the calendar functions), Tony Hutchins, to whom I am indebted for his valuable inputs, has found out that changes and omissions in the TVM solver make it converge much more slowly. For example, f CLEAR-FIN n=10 PV=100 PMT=-20, and solve for i: it takes 15 seconds on the 12C and 40 seconds on the 12CP, but *re-solving* for i only takes 4 seconds on the 12C since the existing i is used as an initial guess; on the 12CP, it still takes 40 seconds, which implies that the solver cannot take user-supplied initial guesses any more. This example is also useful to show the occasional discrepancies in the results: solving for FV and setting the display format to scientific notation (f ·) reveals a difference of 0.44%. The third case in Valentin's article "HP-12C's Serendipitous Solver"[5] (V21N2p40) turned out to be the largest observed difference in running times: 3 min 16 seconds on the 12C vs. *1 hour 51 minutes* on the Platinum! An ill-conditioned case for the new algorithm or simply a bug? Ugly bugs have indeed been found in prototypes[6]… and in production units: When program memory consists of at least 260 lines, the instructions GTO n – 255 through GTO 255 cause an Error 4, n being the last allocated line: 260, 267 ... 393 or 400 (NB: not 399); this was first noticed by Tony Hutchins. Also, pressing f CLEAR-PRGM in run mode no longer resets the current program line to 000 despite being stated in the manual (page 208); this will go unnoticed, however, by all those users who prefer using g GTO 000 or f P/R twice, in order to avoid clearing the program memory by accident. Speaking of bugs, some typos have *again*[7] crept into the list of examples on the back when trying to



[4] For the evolution of the 12C's Nut CPU, refer to Włodek Mier-Jędrzejowicz and Tony Duell's articles in Datafile V19N4p17, V19N5p5 and V20N5p33.

[5] CLEAR-FIN 1 CF_0 0 CFj 80 Nj 3 CFj 0 CFj 53 Nj 8 CFj 5 CFj -2002 CFj IRR

[6] Gene Wright reported that his prototype turns itself off when a running program reaches a R/S or a GTO 000 instruction.

[7] Inspect the cover of V20N5 of Datafile with a magnifying glass, and you will see a wrong 31.51998 instead of 31.051998.

update the arguments for the date and bond functions. The 12C's ON+PMT key combination has been replaced by a reset hole near the 3V CR2032 Lithium battery compartment. Inserting a straightened-out paper clip causes a Pr (power?) Error, i.e. a memory clear.

Display and Keys

Digits no longer seem bold but are a bit larger, which makes the annunciators below them smaller. The display is somewhat dimmer – dim enough indeed to make a few users look up how to adjust the contrast, in vain. One of the two unused annunciators in the 12C ("GRAD") has been removed to leave a space for "RPN" and "ALG", even though the "RPN" status indicator is neither necessary nor consistent with the D.MY/M.DY and BEG/END toggling keys, which only light annunciators for the non-default modes ("D.MY" and "BEGIN"). A battery symbol in the upper-left corner replaces the 12C's asterisk. The characteristic display blinking after non-numeric keys to show the user they have been pressed [8] is now an almost unnoticeable flicker, compensated in part by a very audible key click. The keys are indeed much better than the rattling ones of my 1998 Malaysian unit and as good as the ones on my 1982 12C, made in the USA. The ENTER key on my 12C Platinum, however, is particularly noisy (I wonder if that is why a "Noise Declaration" has been added to Appendix F!). The ON key is still, though very slightly, lower than the other keys. A new typeface is used for the primary key labels, which seem smaller but bold; they are still painted on the keys, not moulded into them, though. Blue legends can be easily read due to the nice cyanish shade, but the gold-prefixed keys are almost illegible under poor light conditions.

Manual and Pouch

The Declaration of Conformity at the front of the manual, signed by the Emerging Business Unit, reveals the manufacturers' names: Hewlett-Packard Co. and Kinpo Electronics. The manual is a revision of the 12C's "Owner's Handbook and Problem-Solving Guide" still printed in black and blue but stretched to A5. The text is not right-justified, for ease of reading. Unfortunately, the very nice drawings

[8] A genuine classic feature: see the "HP-35 Operating Manual", page 2.

from the 1970s that illustrated the examples are missing. A more understandable omission is that of the section "Moving Up From an HP-38?" in the Introduction. Several corrections and additions have been made: a more complete list of error conditions; [9] the SL, SOYD and DB functions, along with RND and AMORT, are now correctly described as functions that actually alter the displayed number according to the display format (page 71); a keystroke procedure to calculate the coefficients of the linear equation $y=A+Bx$ (page 80); and a new brief appendix that covers the algebraic mode, as RPN is assumed throughout the manual. It carefully avoids dealing with calculations such as (49 + 42) × (10 – 95), though. There is an interesting remark in this appendix (page 175): "To change the sign of an already displayed number (it must be the *rightmost* number), press CHS." It appears as if it were taken from the manual for a calculator like the 18C … one of the manuals requested by Kinpo? (See V22N1p21). The "HP 12C Platinum Solutions Handbook" is mentioned several times – will the newly expanded program memory be reflected in more advanced programs? The calculator comes with a nice black leather pouch that resembles the 92169B but is less pliant.

All in All ...

I like it. It is very pleasing to see an HP calculator with a good keyboard and a big ENTER key anew! As the 12C remains in the market, the question is: will it be ousted by the 12C Platinum? In the HPCC Conference mailing list,[10] Frank Wales wrote: "This machine won't outlive the 12C, unless someone puts a bullet in the 12C's still healthy body." Sure enough, there's no comparison between the 12C Platinum and the good *old* 12C, but every new version since 1997 has been a caricature of what they used to be. In my considered opinion, the golden age of the 12C has gone forever. The 12C Platinum may not be an oil painting (its "gold" key is certainly as orange as this issue's covers), but is better than the *modern* 12C provided that all bugs are fixed and the legibility and speed concerns are addressed in a timely manner. [11] After all, considering the present circumstances, it would not have been realistic to expect as well-finished and bug-free [12] a product as the successful 12C at the first attempt. In the blurb of the package, HP wonder how to improve a classic. Did the old 12C need to be improved in the first place? Once modified, though, recovering as many of the features that made it a classic as possible is the most sensible thing to do. Let's appreciate HP's effort to try to do things right.

---

[9] Some are not only additional notes but also actual changes in performance: CF_0 cannot be followed by Nj: N_0 is now always 1. (Page 62 of the manual needs to be updated accordingly).

[10] http://lists.handheld.org/mailman/listinfo/hpcc-conf

[11] Bug reports from early purchasers will make it easier: the Singapore reseller educalc.net was granted permission to sell the 12CP up to three weeks before the official release date (27th May).

[12] John H. Meyers has recently reported that he did find a bug in the 12C: "It has to do with trying to enter non-integers as cash flow repetition factors, some values like 1.000000001, which *should* be detected as invalid, [are sometimes] accepted, and then when recalled, the value actually stored (or recalled) may be found to have been interpreted as zero". Fixed in the 12CP, since all non-integer values for Nj trigger an Error 6.