41CL Beta Test Results

Monte Dalrymple

Background

The 41CL project was first discussed at the HHC in 2004, but little work was done on the project until the middle of 2009. In early 2010 two Alpha printed circuit boards (PCBs) were fabricated. After several months of debugging and a minor revision to the board, twenty-five assembled Beta PCBs were received in February 2011. First units were shipped to Beta testers at the beginning of June 2011. This is a brief report of the Beta test results.

Acknowledgements

Twenty-one users received boards. You probably recognize many of the names of Beta testers, and I would like to publicly thank them for their efforts:

Paul Dale	Roger Schoenthal	Massimo Gnerucci
Theo Kastrissios	Prabhu Bhooplapur	Eric Smith
Richard Nelson	Gene Wright	'Angel Martin
Geoff Quickfall	Geir Isene	Wlodek Mier-Jedrzejowicz
Namir Shammas	Stefano Dini	Peter Platzer
Diego Diaz	Jurgen Keller	Raymond Wiker
Antti Louko	Etienne Victoria	Matthias Wehrli

Beta testing

The Beta testing got off to a somewhat slow start. I notified the first 20 people on the sign-up list that boards were going to be available, but 5 never bothered to reply and another 5 were no longer interested. Eventually the 21 people listed above got boards, but only after a couple of weeks worrying about recouping my investment.

I did a significant amount of simulation before releasing the boards, so I didn't expect that there would be too many hardware bugs. To date, only one hardware issue has been found. The bug has been corrected in the design source code, and several Beta users have had their hardware updated with the fix. This is one advantage of the programmable nature of the hardware used in the design.

I did not originally include the WROM instruction in the design, thinking that the new WCMD instruction would take care of that functionality. When someone pointed out that several interesting modules (like HEPAX) used the WROM instruction I added it to the design. This was actually after the Alpha boards were assembled. I simulated the addition, but completely missed the fact that there was a subtle dependency on the address being written. Of course the HEPAX module was one of the first that several Beta users tested.

The new features of the 41CL, like the Turbo mode, the MMU and serial port, require significant software support, and this is where most of the bugs found by the Beta users were located. The new 41CL Extra Functions is 4k of machine code. I had tested as much of the code as possible using the V41 emulator program, but most of the new features required that I watch the execution trace carefully, which led to some missed problems. Some of the new features could not be tested this way, though.

Prior to the Beta release I had never simulated the entire design, mainly because I did not have a simulation model of the HP-41C display driver chips. After creating a Verilog model of the display chips I was able to simulate the entire design, which made it possible to verify software bugs and their fixes.

Nine separate issues were found in the 41CL Extra Functions software, and all of them can be fixed in the field. The 41CL Extra Functions allow the user to copy 4k blocks of memory from the Flash (non-volatile) memory into RAM, where the code can then be patched. The MMU allows the user to use the patched code directly, or the code can be written back to the Flash memory. The original Beta units shipped with version -1A installed, but the fixes raised the revision level to -1E.

There is one software issue that cannot be patched, in the Operating System area of memory. To protect users from "bricking" the calculator, this area is off-limits to modification. The problem occurs while executing in Turbo mode. There are places in the code that should always be executed at normal speed, most having to do with delays for keyboard scanning or message display. These locations can be tagged in memory to force the hardware to suspend the Turbo mode. Unfortunately, I missed one such delay loop, the one that waits to signal a keypress to change to "NULL". But this is fixed in the production Flash image.

Several new commands were added to the 41CL Extra Functions based on Beta user feedback, which accounted for two of the revisions (and one bug). The production PCBs will be issued with revision -2B, which adds several new module images and streamlines the code for several of the commands.

In addition to the new commands that were added, a new hardware feature has been included along with the fix for the one hardware problem. This feature allows the user to substitute code in RAM for the Operating System of the calculator. This feature is obviously only for advanced users, and requires a passphrase in ALPHA to be enabled.

One bug discovered during Beta testing is still not resolved. When the Time Module is present in a 41CL and the machine is turned off and then back on very quickly, the Time Module appears to be reset. I have simulated this scenario, watching the bus signals that the Time Module sees for any anomalies, but everything appears to be within specification. I will probably have to hook up the logic analyzer to real hardware to try to find the root cause of this issue. But I do not consider this to be a reason to hold up the next batch of boards to be assembled.

It has taken forever to bring this project to fruition, but I think that it has been worth the wait. The 41CL on my desk is my go-to calculator.

In addition to becoming intimately familiar with both the hardware and software of the 41C, I have learned a number of lessons:

Lessons Learned

1. Designing the chip is only about a third of a project like this. Doing the schematic and layout of the printed circuit board took nearly as long as the chip design. Writing the 41CL Extra Functions took the remaining third of the time. Interestingly, there are more "lines of code" in the 41CL Extra Functions than in the Verilog HDL that describes the design:

41CL Extra Functions	4050 lines of Nut assembly code
NEWT top level	1300 lines of Verilog code
Nut CPU	1101 lines of Verilog code
Keyboard scanner	196 lines of Verilog code
Serial port top level	105 lines of Verilog code
Serial receiver	149 lines of Verilog code
Serial transmitter	137 lines of Verilog code
CPLD portion of design	198 lines of Verilog code

2. Having a printed circuit board assembled is expensive. The cost is not linear, either. Assembly of the two Alpha boards cost almost the same amount as the 25 Beta boards.

3. Verifying software is much more time-consuming, and much more difficult, than verifying hardware.

4. The power supply portion of the design is a significant portion of the cost. The power-supply chip is more expensive than the CPLD used in the CPU. And the two high-quality capacitors in the power supply together are also more expensive than the CPLD!

5. Only about half of people who say they are interested will actually step up. This has caused me to hold off on the first production batch until I think there are enough people interested. One batch is a significant investment for me.

6. International shipping, except to Canada, Australia and Switzerland, is almost not worth the trouble. The Customs departments of various EU countries in particular seem to be staffed by people of questionable competence.