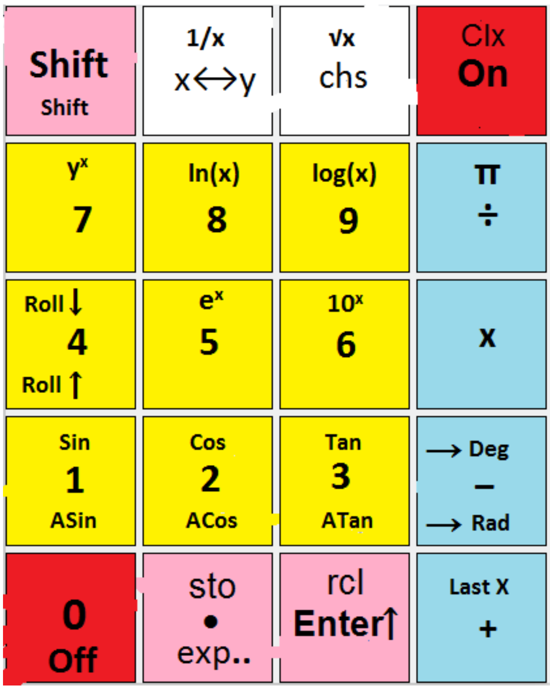
Further improving the $10 DIY RPN calculator from China Chris Morris XI/24

The big breakthrough here came in late 2019, when Jeff Yang wrote new C firmware[1] that turned this DIY kit (model CAI-150) from a simple 4 function calculator with a few frills into a serious RPN scientific with 18 digits of precision! This compares to just 8 digits precision for the 1980's Elektronika Soviet RPN family of calculators (eg the MK-61) and 10 digits for the HP25/33E/34E/15C of the late 70's and early 80's. To be fair, the latest Semico/Elektronika MK-161, made in Novosiburs k, Siberia, has 14 digits of precision. At the bottom of this precise list is Sinclair (UK) who used 6 digits, though in a clever way. This will be discussed later.



I built my $10 kit into a finished calculator and was very pleased with it, especially once I had figured how to flash Jeff's V1.14 RPN firmware (April 2022 released). The display is an attractive illuminated 2x16 dot matrix LCD, while the Beijing-made microcontroller is the STC Micro 8 bit 15W413AS in a socketed pdip28. Although using the ancient Intel 8051 architecture of 1980 with the usual limitations, it has several things in its favour. Produced in large quantities in China by several companies, it is very cheap and comes with many through-hole versions, as well as the usual surface mount options. Speeded up a dozen times from the 1980 original, it executes instructions at clock speed (with an internal and accurate 11MHz RC trimmable oscillator). Most important of all, it requires no programmer hardware. An ingenious STC bootloader is built-in to the microcontroller, using R_x, T_x and Ground pins. These pins are just connected to the usual FTDI dongle driven from the PC USB port. It can even reprogram while the original program is running, then interrupted by a power down cycle to start flashing. All this is well described at Jeff's GIThub site in the ReadMe file. PC software (to program in the compiled RPN DIY hex file posted there) is freely provided by STC Micro[3] as a simple .exe file, which I used in preference to the Python-based freelance STCgal.

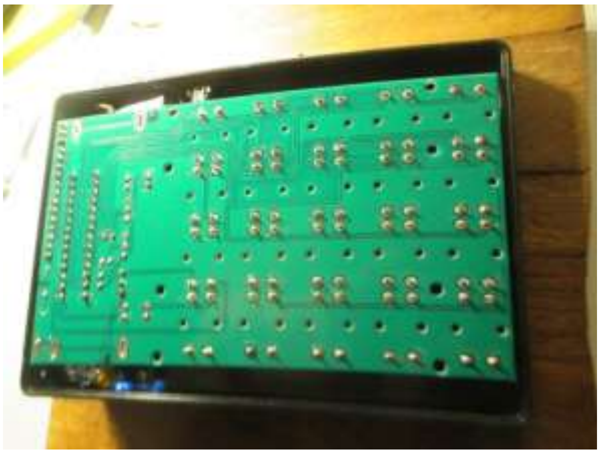
The injection-moulded black case provided in the kit is attractive and sturdy and has some space for the improvements to be described. It fits together very well. In 2019, awkward acrylic laser-cut panels were used to 'build' the DIY case with nuts and bolts. Evidently the success of the DIY kit led to a proper injection-moulded case being developed. Interestingly it resembles the Siberian MK-161 case in shape and has a similar LCD viewing angle. There is still a downside however.

The CR2032 twin batteries provided give a 14 hour life at 16 mA with the new firmware. This means dismantling the calculator case often to change batteries. The dozen steel screws, that hold the case together and the circuit board and separate LCD display in place, will not take long to split the plastic case towers with repeated use. It is also a fiddly business getting the angled LCD display to mate up with the main board, not something to keep repeating. Step 1 is to remove (or leave off) the soldered coin cell clips in the kit, once you know the kit works for the simple version.

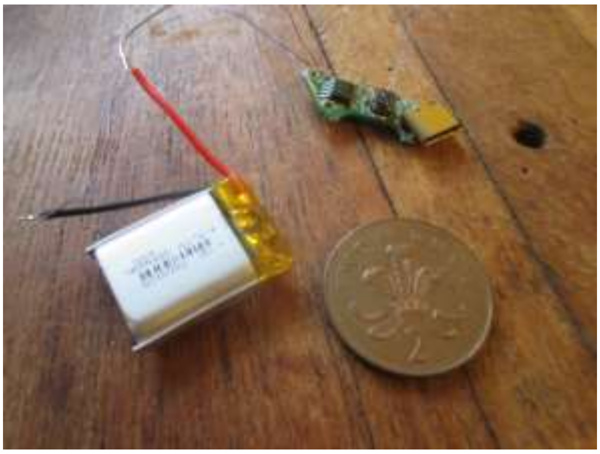
In other respects, the calculator kit is well thought out mechanically, with 20 sturdy 'Omron style' push buttons that have transparent plastic tops – a feature other calculator kits should copy. It is easy to design one's preferred lettering and style for the keys in *Word Paint* and slip the paper label underneath the tops. I have included my own version in this write-up.

Back to the batteries. In keeping with the thoughtful yet frugal nature of this project (Jeff managed to cram all the new RPN operating system into the same 13K bytes of the original micro that came with the kit, for a no-cost conversion), I sourced a $3 keyfob[4] that was intended to give a cellphone emergency power. Inside is a compact LiPo battery and a most ingenious controller board using the single FM9688 asic from Shenzhen Fuman Electronics Ltd. I converted that data sheet from Simplified Mandarin with the Google online translator to readable English. It controls the charging of the 3.7v Lipo, via a 5v USB cable, from trickle to constant current to constant voltage, then off. At the other end it raises the battery voltage from ~3.7v to 5v when a load (here the calculator) is turned on, with a choke running at 1MHz for boost conversion. This is all done with minimal external circuitry, even the signal LEDS need no dropping resistors. A remarkable home-grown asic. Most importantly, it uses these red and blue LEDs to keep the user well informed. During charging, the red LED flashes, going to a steady on when charging is complete on the calculator . The blue led is on steady when the calculator is on and not charging. There is no on/off switch, the calculator uses one of the keys to turn on and another to turn off.

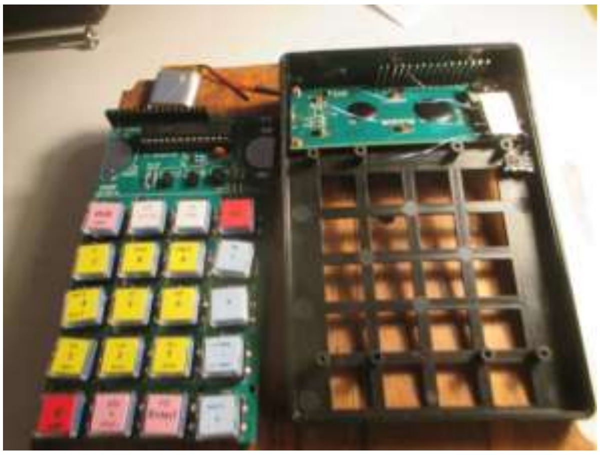
Essential for the life of the LiPo battery, the blue LED flashes as voltage drops (<3.3v), then turns off completely and automatically locks out as the battery lowers further to ~3v. It is not good to allow a Lipo to drop below 2.8v. All these pyrotechnics can observed through a 2mm hole I drilled in the side of the calculator case near the control board.



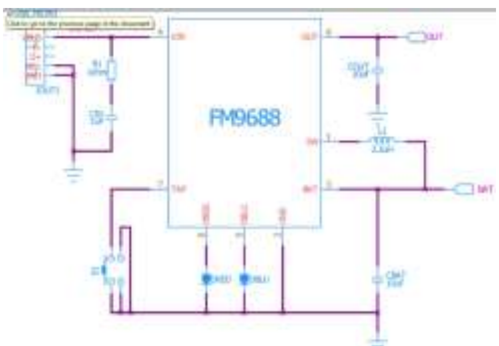
For charging from any USB socket, I built a special cable from scrap parts, ending in a submini stereo jack. This links up with the world's tiniest submini stereo socket from Switchcraft in the USA, held in place by cyanoacrylate adhesive in a 4mm hole in the case side. An earlier attempt with a too-large mono socket is now hidden by a rubber bumper. The cabling inside, see diagram, uses 'solderable' polyurethane insulated solid wire of 0.2mm diameter, very easy to route yet capable of ½ amp in short sections. This wire was recommended by Bernhardt at Panamatik for his LP Spice conversion to a PIC processor, it is amazing stuff for close clearances inside a calculator.

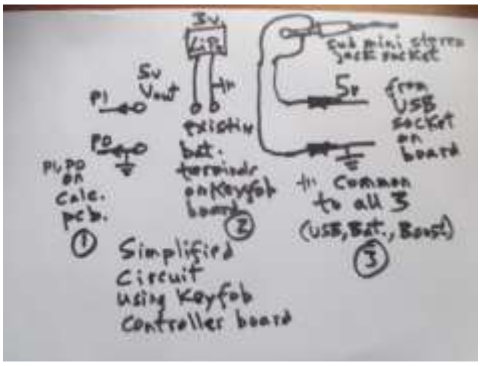


The control board is mounted to the left of the display, using thin tape. The 300 mAhr LiPo battery is to the right of the display, using thicker double-sided tape. I had a major worry before the case was finally closed. The DIY RPN calculator went into an infinite loop with the trig function $\cos^{-1}$ while doing Mike Sebastian's 9^{o} forensic test. Don't try this at home. The normal *shift OFF* command was being ignored, but re-reading Jeff's notes I used the emergency *shift hold OFF*. That did the trick.



Speaking of trig functions, there was barely enough room left in the 13K byte ROM to fit them in easily, so Jeff's co-developer used the brilliant (though inaccurate) Sinclair trig algorithms, devised by Nigel Searle[2] in 1974 to fit in almost no ROM space at all.. They work, sort of. By contrast, all the other transcendental functions are above reproach, as good as the late 1970's HP LED calculator algorithms, themselves still state-of-the-art in 2024. Combined with 18 digit precision, this makes for a formidable $10 RPN calculator with excellent positive keyboard action in a well-designed case, even if there are only 20 keys.. Sinclair to the rescue again with multiple key shifts.





Finally all was screwed together, hopefully good for the ~500 recharge cycles of the LiPo battery. It is nice to have a non-tethered and powerful pocket RPN calculator that does not have to be pulled apart frequently, all for an extra $3.

References:

1. https://github.com/jjj11x/stc_rpncalc
2. http://files.righto.com/calculator/sinclair_scientific_simulator.html