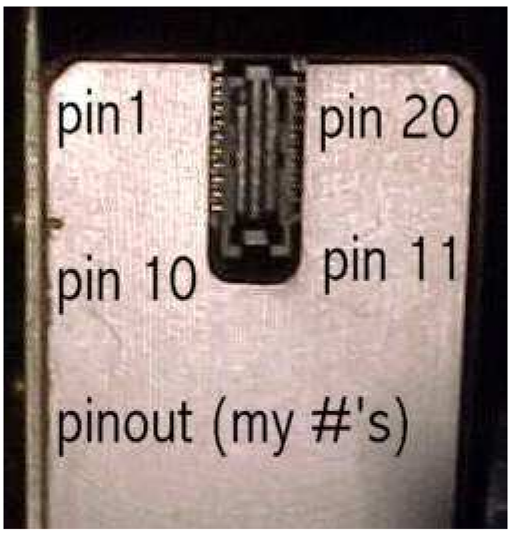
Fixing up the FEI 5650A Option 58 b package

Congratulations! You've got a Rb clocked frequency synthesizer! (With very few modifications) Basically, this clock seems to have been designed to produce a 1pulse-per-second similar to the one from a GPS running in timing mode. I asked FEI about it and was abused, they insulted me bigtime as well as the company that sold me this unit (Ridge) by telling me it was stolen, proprietary, and that was that. The proprietary part is undoubtedly correct, but the attitude was rude! In any case, here is my analysis of this unit.

First, look at the connector illustration below:



Hold the device so that the connector is upward. The pinouts are numbered here as if you were looking at a chip, that is starting at the upper left and proceeding down the left side and up the right side:

Pin 1	2	3	4	5	6	7	8	9	10
Com	Com	Com	Com	+15	+15	+15	+15	+15	+15
Pin11	12	13	14	15	16	17	18	19	20
Com	Com	Com	1pps	+5	Com	Com	Lock	Com	Nc?

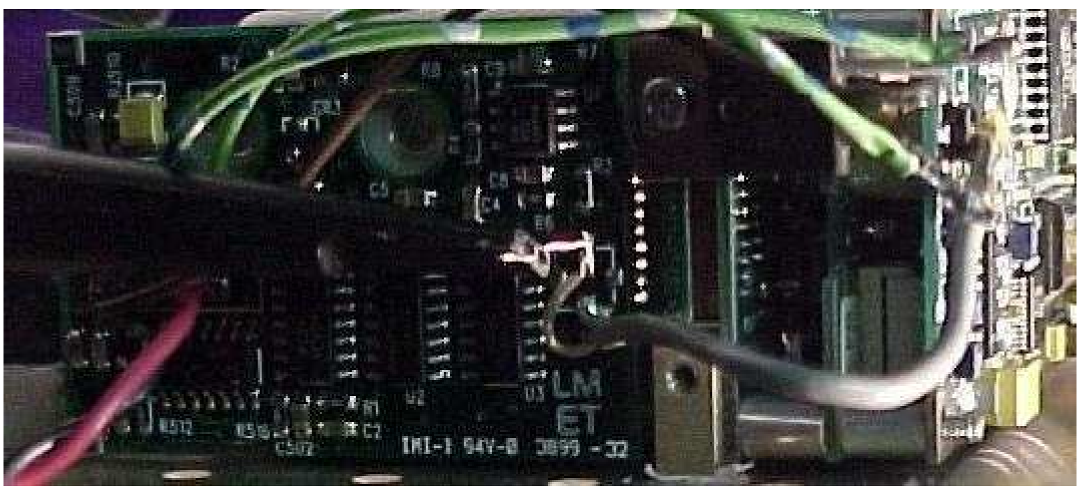
Lock is apparently an open collector lock indication signal. I haven’t tried this out yet. **

The mating female connector is an Amphenol type 140655-3, Digi-key A3107-ND It's made for soldering to a circuit board, but can be used with wiring.

Com is case ground and the common negative for the +15 and +5 volt supplies. The unit will run at +15 although that leaves about 1 volt drop across the regulators, which are connected in parallel. The regulated voltage out is 14.2 volts (great series regulators!). It will run at +15 with the aluminum heatsink removed from the regulators. Higher input requires the AL heatsink left on the regulators. Probably should not exceed 16-17 volts. Why this weirdness is beyond me! Current drain is a little over 1.5 amp on warmup, 700 ma after warmup. Warmup time is about 5 min. Current drain on the +5V supply is about 200 ma or less.

To Modify:

Take off the cover. The AL block (regulator heatsink) also needs to be removed. Take the power board off the AL block, which is part of the heatsink for the voltage regulators (there are two of them). The device works just fine with pin 18 and 20 floating.



Now that the power board is exposed, solder a stub on the coax board point; that's where the frequency output is. It's a sine wave, probably will not drive very much. Just to the left of this point is a comparator that will give a pulse out at the frequency of the output. This comparator drives the divider chain to get a 1 second pulse.

If you have fired up the system by now, you'll find a frequency of 8.38860 MHz at the coax point, which, when divided by 2^23 gives the 1 sec pulse on pin 14.

To begin the real fun, get a 9-pin or 25-pin Serial port cable and either cut one connector off (get the right one!) or else get a 9-pin or 25 pin solder pot connector. On the freq divider board and picture, look at the upper left corner, see a row of 5 pins. From the RIGHT as you look at them, the pins are Gnd(com), RS232 Transmit (from the device perspective), RS232 Receive, +5, and +5.

Hook up the connector or cable as follows. If it's a 9 pin, pin 5 to Gnd, pin 2 to the Transmit, and pin 3 to the Receive pin. Connect 6 and 7 together on the solder connector or connect the wires together to fool the PC com port. If you have a 25 pin connector or cable, put pin 2 to Receive, pin 3 to Transmit and pin 7 to Ground. Connect pins or wires on pins 4 and 6 together.

Fire up Hyperterm or other communications program, set it to the appropriate com port, 9600 baud, 8 bits, one stop, no parity, no control, and type a capital S <cr> to the device. It should respond with

    R=50255057.012932Hz F=2ABB504000000000
    OK

If you don't get this exact number, don't panic, but they should be close. You may have to put a LF after CR on the terminal program or some such. Also set the terminal to echo what you type, because the device apparently does not echo what is typed to it.

Now go to the Analog Devices website, and download the data sheet (PDF) for the AD9830A. Look on the data sheet, and see how it works, check the arithmetic between the F= first 8 hex characters and the reference frequency R= that you got above.

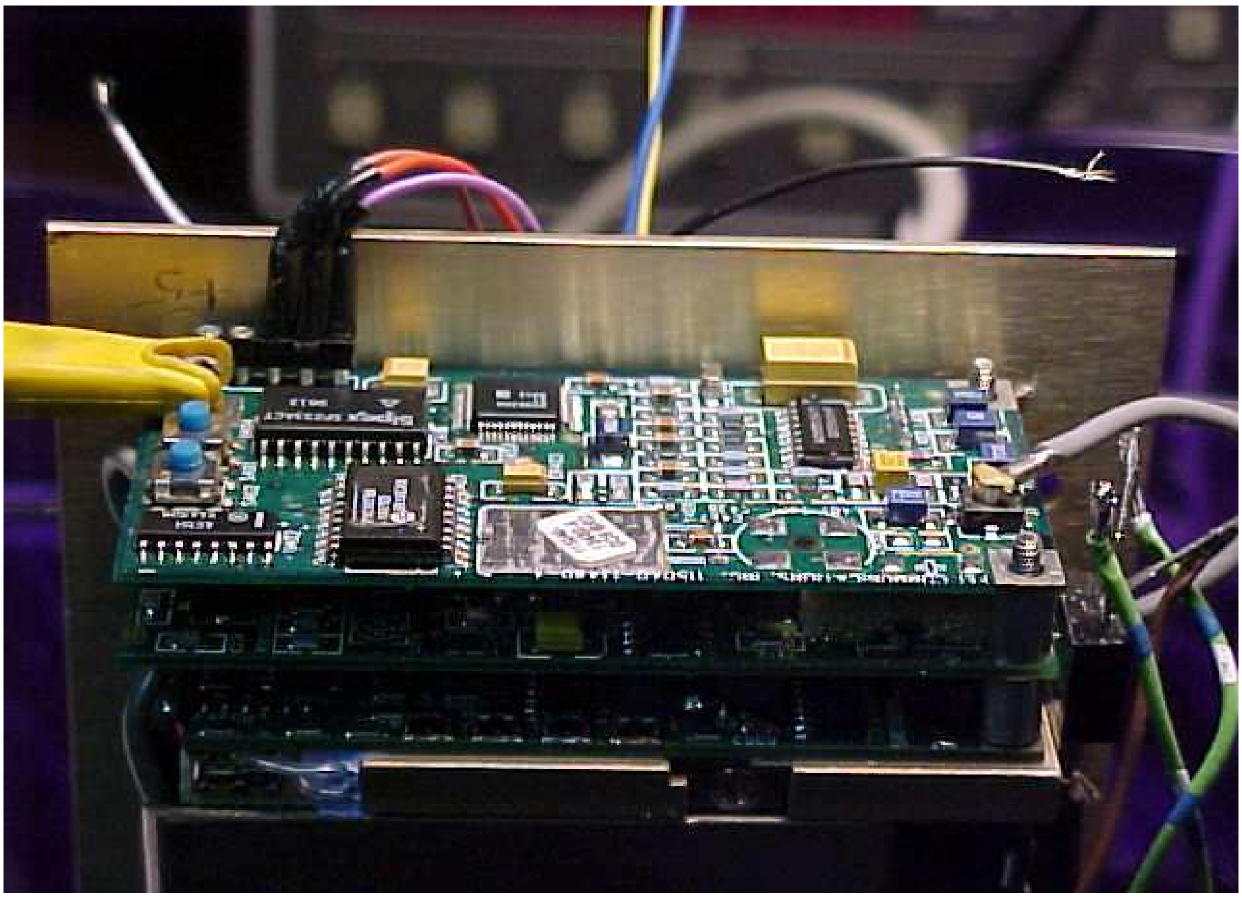
To change the output frequency, simply type F=(8 hex digits that you have calculated) <cr> and you will see the frequency change at the jury-rigged test point!

Typing R= followed by the reference frequency will set the reference frequency. Verified by typing S <cr>

If nothing further is done, the system will return to the original F and R values on a power cycle.

To fix the new values into the microprocessor, type E <cr> (presumably for enter) after you have modified the value.

The little blue buttons (two) on the divider board increment and decrement the hex divisor by one lsb per press. S402, furthest from the RS232 pins increments, and the other, s401 decrements. These decrements or increments stay fixed on a power cycle. Note that if you want to increment or decrement as a disciplining possibility, simply carefully stick some leads on the switch leads, the switches pull to ground on closing! So, a simple phase comparator setup with an external source could be used to discipline the divider.



Also, on the divider picture, see a test point near the center at the top. This is where you measure the reference frequency of about 50.0225 MHz.   I think maybe the frequency was measured at the factory, entered in to the PIC and that's what comes back as R=.   Then the appropriate divisor was set into the 9830 using the F= command, so the appropriate output frequency could be set without having to wait a very long time to measure the one second output against some other standard.   These numbers were set using the E <cr> command.

Enjoy the unit, it is truly useful when setup as a signal source. If anyone should write a little visual basic control program, that'd be nice.

Selah

** Note from Bob Scott:

Hi Don:

Well, I finally got the time to finish this up. Photo attached.
I discovered both the lock line and the test tone output are open collectors. Amazing what a couple of resistors can fix.

I wound up running the 1PPS through two back-to-back inverters as buffering/protection. I'm also using the output of the first inverter in that chain to trigger a 555 one-shot connected to a front panel LED.

The lock line similarly goes through two inverters to drive a high efficiency red LED.

1PPS and the test tone are both routed to front panel BNC's.

The thing ran WAY too hot for my liking - too hot to touch anywhere.
I'm guessing it was attached to a serious heatsink when it was in service. I attached a copper forced air CPU cooler from the local computer store ($9 !) to the steel plate and dropped the temperature tremendously. Current consumption went up 100 mA (after subtracting the fan draw), so I'm guessing the heater is running a bit harder, but overall I'm happy with the result.

Thanks again for cracking the code on this gadget!

Cheers,
Bob