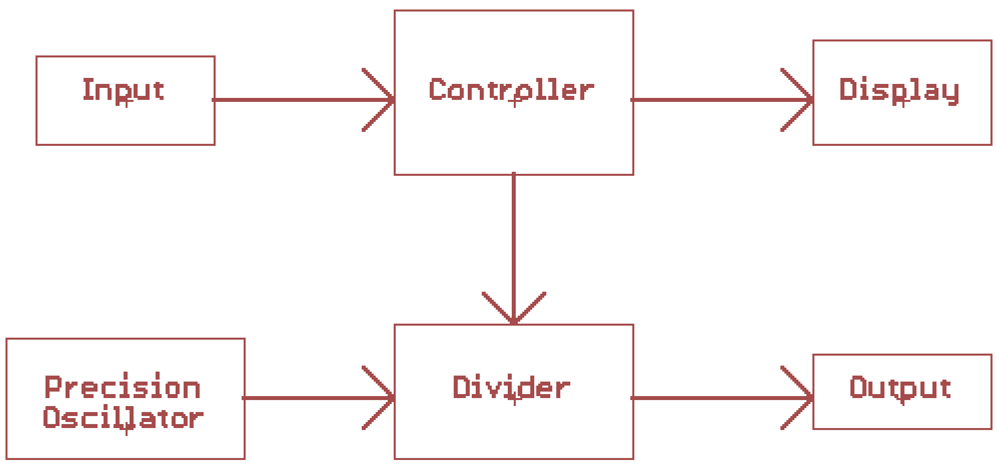
Alternatives and Variations

The Frequency Reference is made up of just a few functional blocks connected as shown below.



The heart of the device is the precision oscillator. The final output frequency is an integer division of the oscillator. The output block in this case is merely a resistor and a BNC connector. The divider circuit is an integrated peripheral of the controller. Once the divider is set up by the controller, it operates independently. The controller can change the setup of the divider at any time and does so in response to input.** Input can either be the rotary encoder / switch or the USB/serial port. Note that the controller sends setup information to the divider but does not ever read anything back. The display is driven by the controller and does not rely on the state of the divider.

The precision and stability of the entire circuit depends on the oscillator. The nominal design uses the RTX-230 which actually has pretty good stability considering it costs around a dollar. To make a better reference, choose an oscillator with better specs. Rev-B of the circuit board retains support for the RTX-230 but also allows for oscillators with very good published specifications for, granted, a lot more money. Because most of the circuit is digital, the overall precision of the device increases with the better oscillator. That is, the fact that the rest of the circuit is cheap does not mean that it degrades the better oscillator performance.

Here are a few examples of alternate oscillator parts with increasing accuracy, stability and price. The Raltron part is also shown for comparison. Any of these will work with the Frequency Reference circuit, but not with the Rev-A PCB. I will work to include support for as many of these as I can on the Rev-B PCB (although due to cost, I will not be able to test them all).

Type	MFR	Part #	Initial (+-)ppm	Temp (+-)ppb	Voltage (+-)ppb	Aging (+-)ppb/day	Inclusive (+-)ppm	Over(yrs)	Temp Range	Cost	Supply Voltage	Supply Current max mA
TCXO	Raltron	RTX-230	Adj	2500	300		1	1	-30-75	$ 1.10	5	2
TCXO	Connor-Winfield	M100F-020.0M	1	100	200	40	3.0	20	0-70	$ 23.70	3.3	3.3
TCXO	Connor-Winfield	DOT050F-020.0M	1	50	20	10	4.6	15	0-70	$ 39.38	3.3	10
OCXO	Connor-Winfield	DOC020F-020.0M	1	20	20	10	4.6	20	0-70	$ 54.33	3.3	758
OCXO	Connor-Winfield	OH100-70503CF-020.0M	0.1	5	2	1	0.3	20	-20-70	$ 115.50	3.3	970

Prices are from Digi-Key

**Changing frequencies is not phase synchronous. When the change occurs, the clock is stopped, the new divisor (or prescaler) value is programmed, the count is zero'd, then the clock is restarted.

Other Frequencies

The board and software do not really know what frequency they are running at. You could therefore substitute oscillators of other frequencies. The output frequency would scale but otherwise everything would work the same. The maximum frequency oscillator is in the neighborhood of 20MHz. Higher frequencies would be over-clocking the controller, but it will still work up to a point. Lower frequencies are also okay, however the display might start to flicker as the lower clock means a slower controller. The flicker can be fixed by changing the delays in the software. In either case baud rates will need to be scaled for correct serial communications.

As an example, the exact same (source) code will run on a SparkFun Redboard (Arduino Uno clone) which runs at 16MHz and derives its clock from a ceramic resonator rather than a crystal. The software works fine (with degraded accuracy, tempco and jitter), however the frequency table shifts down to the following.

Frequency (Hz)	Display	Scale (Implied)
8,000,000	10.00	MHz
4,000,000	5.000	MHz
2,000,000	2.500	MHz
1,600,000	2.000	MHz
1,000,000	1.250	MHz
800,000	1.000	MHz
400,000	500.0	kHz
320,000	400.0	kHz
250,000	312.5	kHz
200,000	250.0	kHz
160,000	200.0	kHz
100,000	125.0	kHz
80,000	100.0	kHz
64,000	80.0	kHz
40,000	50.0	kHz
32,000	40.0	kHz
20,000	25.0	kHz
16,000	20.0	kHz
10,000	12.5	kHz
8,000	10.0	kHz
6,400	8000	Hz

```
    4,000             5000          Hz
    3,200             4000          Hz
    2,500             3125          Hz
    2,000             2500          Hz
    1,600             2000          Hz
    1,000             1250          Hz
      800             1000          Hz
      640              800          Hz
      400              500          Hz
      320              400          Hz
      200              250          Hz
      160              200          Hz
      122.07           153          Hz
```

Note that the display does not change since it is unrelated to the actual frequency. You can make the display correct (sort of, without more work, decimals will be wrong sometimes) by substituting the following data table in the source.

```
static uint16_t dispFreq[] = {
  8000,      //  8,000,000
  4000,      //  4,000,000
  2000,      //  2,000,000
  1600,      //  1,600,000
  1000,      //  1,000,000
  8000,      //    800,000
  4000,      //    400,000
  3200,      //    320,000
  2500,      //    250,000
  2000,      //    200,000
  1600,      //    160,000
  1000,      //    100,000
   800,      //     80,000
   640,      //     64,000
   400,      //     40,000
   320,      //     32,000
   200,      //     20,000
   160,      //     16,000
   100,      //     10,000
  8000,      //      8,000
  6400,      //      6,400
  4000,      //      4,000
  3200,      //      3,200
  2500,      //      2,500
  2000,      //      2,000
  1600,      //      1,600
  1000,      //      1,000
   800,      //        800
   640,      //        640
   400,      //        400
   320,      //        320
   200,      //        200
   160,      //        160
   122       //        122
};
```

By further adjusting these tables, oscillators of other frequencies can be accommodated.

Cost

The original intent of the "*$3 Frequency Standard*" has been compromised by the cost of the rotary encoder, PCB and display circuitry. Still the entire cost remains under $22 total, which is not too bad in my opinion. Here is a cost breakdown of the BOM.

Qty	Value	Device	Parts	Source	Part #	Cost
1	20.000000MHz	TCXO-RTXO230LC	OSC1	eBay		$ 1.10
1		PCB		OSHPark	a4BIq3K5	$ 5.67
5	0.1uF	Capacitor	C2, C3, C5, C6, C9	Digi-Key	399-1171-1-ND	$ 0.27
2	10k	Resistor	R1, R2	Digi-Key	311-10.0KLRCT-ND	$ 0.01
1	10uF	Capacitor	C4	Digi-Key	587-2225-1-ND	$ 0.26
3	10uF	Capacitor	C1, C7, C8	Digi-Key	478-8456-1-ND	$ 1.40
2	2.5k	FERRITE_BEAD	FB1, FB2	Digi-Key	1276-6377-1-ND	$ 0.20
1	49.9R	Resistor	R4	Digi-Key	311-49.9CRCT-ND	$ 0.14
1	5V	Diode, zener, 4.7V	D1	Digi-Key	568-6286-1-ND	$ 0.01
1	3.3k	Resistor	R3	Digi-Key	311-3.30KCRCT-ND	$ 0.01
1	ATMEGA328P	ATMEGA328P	U1	Digi-Key	ATMEGA328P-AURCT-N	$ 3.70
1	BNC_RIGHT-ANGLE	BNC connector	J2	Digi-Key	A97553-ND	$ 2.03
1	FTDI Basic	pins	FTDI	SparkFun	PRT-00116	$ 0.23
1	HP_DISPLAY	HP_DISPLAY	DISP1	SparkFun	COM-12710	$ 2.95
1	MIC5205	Voltage Regulator, LDO	U2	Digi-Key	576-1262-1-ND	$ 0.62
1	PWR	SWITCH,BUTTON,ROUND,7MM	SW1	Mouser	611-PVB6THTEE300NS	$ 1.33
1	PWR_IN	Connector	JP1	SparkFun	PRT-08233	$ 0.45
1	Switch	ROTARY_ENCODER	S1	Digi-Key	987-1198-ND	$ 1.11
1	SN74LV164APW	SN74LV164APW	U3	Digi-Key	296-3788-1-ND	$ 0.46
						$ 21.93

If this is too expensive for you, the cost can be lowered by not populating various parts of the PCB.

You can eliminate the display by not populating DISP1, U3 and C9. The device will operate the same. You can use an external frequency counter to see where you are at.

The device will also operate if you do not populate the rotary encoder. Then user input will need to be supplied over the serial interface.

You can choose to not populate the power switch. Either wire in a cheaper switch or just jumper the NO to C pins on each side. You can choose to not populate the power input connector by just soldering wires in directly (or by powering only off of USB through the FTDI connector). If you only power from USB/FTDI you can choose to not populate the voltage regulator, U2, and associated components, JP1, C7, D1, FB2, R2, & R3.

If you exercise all of these, the cost comes out to be $14.37. You can squeeze out a few more dollars if you forego things like the BNC connector and use cheaper components, but the output signal will suffer.