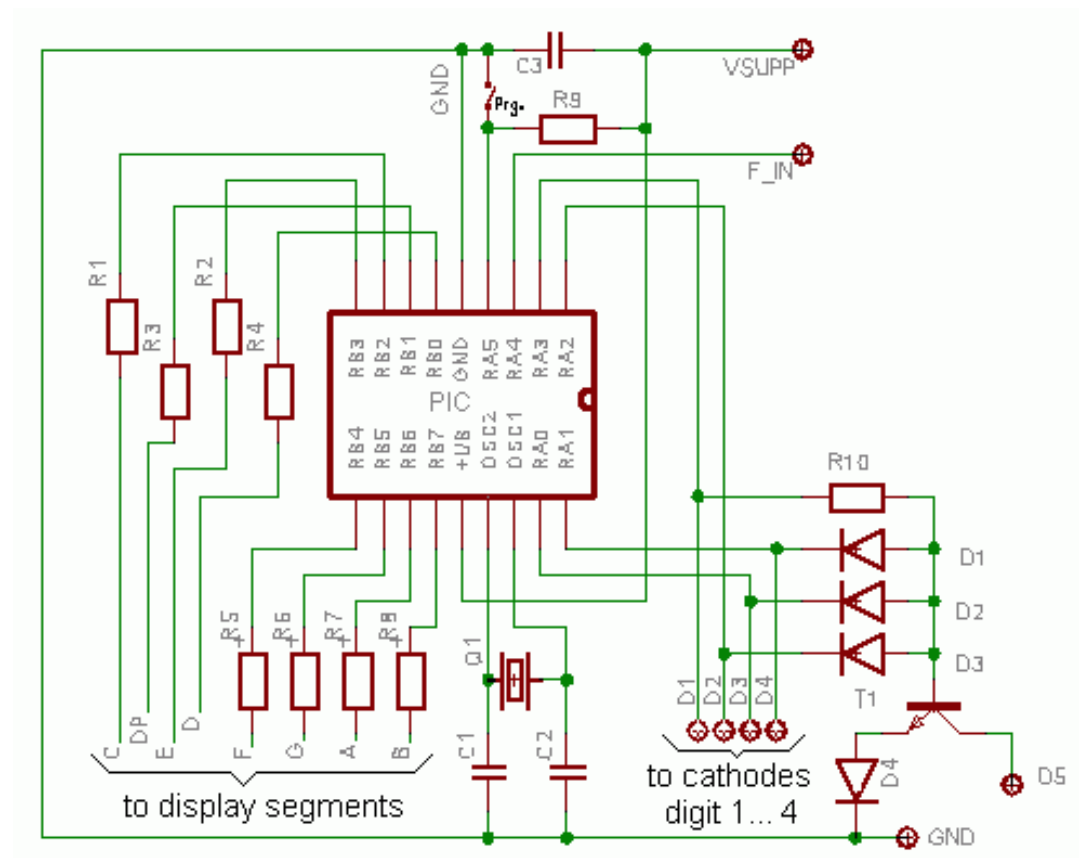


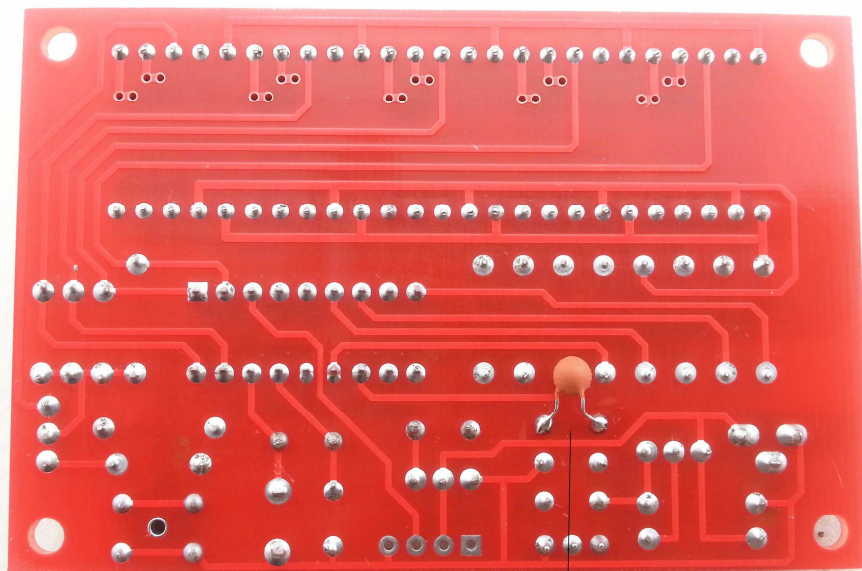
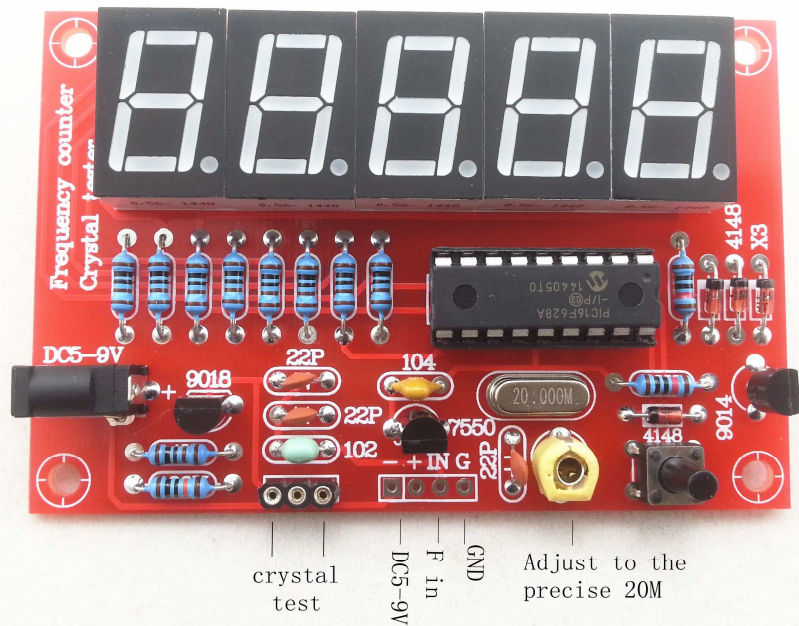
1-50Mhz Frequency counter

This document describes the construction of small frequency counter with a cheap PIC microcontroller and a few seven-segment LED digits. The main features of the frequency counter are:

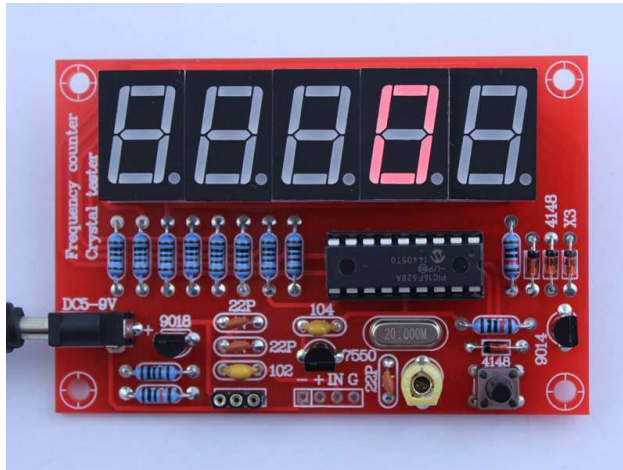
- frequency range 1 Hz ... 50 MHz
- five digits resolution (display for example x.xxxx kHz, x.xxxx MHz, or xx.xxx MHz)
- automatic range switching with different gate times
- optional addition or subtraction of a frequency offset (programmable)
- very low component count: a PIC 16F628, 5 7-segment LED displays, a 20MHz crystal and a few resistors
- optional (configurable) power-saving mode which automatically turns the display off if the frequency didn't change significantly within 15 seconds
- USB 5V power supply can be used.

main schematic:

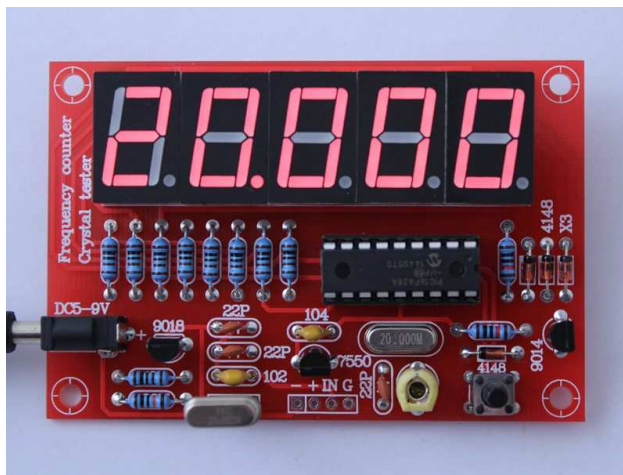




If the low frequency oscillator measurement is not stable, add a 22P capacitor in here.



Power on, no signal



Crystal test (20M)

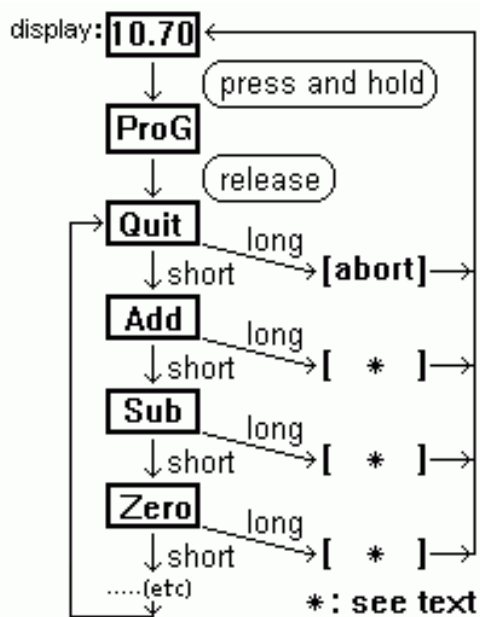
The display range is automatically switched to give the maximum readout accuracy (with 5 digits). The gate time is also selected automatically as listed in the following table:

Frequency range	Display	Gate time	Decimal point
0 ... 9.999 kHz	X.XXXX	1 second flashing	(which means "kHz")
10 ... 99.99 kHz	XX.XXX	1/2 second flashing	
100 ... 999.9 kHz	XXX.XX	1/4 second flashing	
1 ... 9.999 MHz	X.XXXX	1/4 second steady	(which means "MHz")
10 ... 50.00 MHz	XX.XXX	1/4 second steady	

Adding or subtracting an offset frequency

If the counter is used in a shortwave receiver or transceiver, you may want to add or subtract an offset value from the measured frequency. The offset frequency is the same as the intermediate frequency in many cases, because the counter is usually connected to the receiver's VFO (variable frequency

oscillator). For this purpose, a programming mode (aka "setup mode") has been implemented in the firmware so you can enter an offset frequency without reprogramming (or even reassembling) the PIC firmware.



The program flow chart on the left

shows how to enter programming mode, how to select a menu, and how to execute the associated function. To enter programming mode, press and hold the programming key until the PIC shows "ProG" on the LED display. Then release the key. You are now in the first menu of the programming mode.

To select the next menu, press the key for a short time (less than a second).

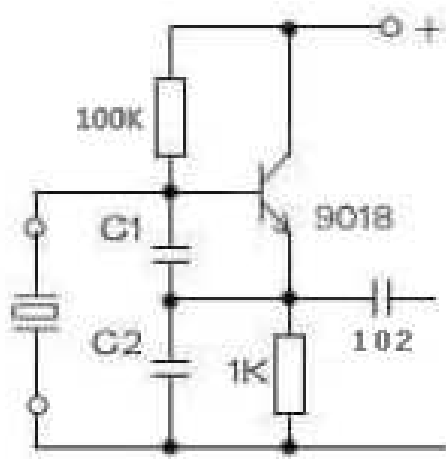
To execute the selected function, press the key for a longer time (more than a second). The menu functions are :

- "Quit" : Aborts programming mode without changing anything.
- "Add" : Saves the previously measured frequency permanently, so it will be added in future.
- "Sub" : Saves the previously measured frequency permanently, so it will be subtracted in future.
- "Zero" : Sets the frequency offset to zero, so the display will show the measured frequency without offset. The previously programmed offset will be lost.
- "Table": Allows you to select a predefined offset value from a table. The table itself is also located in the PIC's data EEPROM, so you may find different values in it. When skipping through the table, the frequencies are shown in numeric form, like 455.0 (kHz), 4.1943 (MHz), 4.4336 (MHz), 10.700 (MHz). After selecting an entry (long

keypress), you will be taken back to the main menu to select "Add" or "Subtract".

- "PSave" / "NoPSV": turns the power-saving on/off. In power-saving mode, the display is turned off after 15 seconds of no "significant" change in frequency, and on again as soon as the frequency changes by more than a few dozen Hertz (in the 3..4 MHz measuring range).

Crystal test schematic:



Attention: Max input signal voltage $\leq 5V$!

Crystal test range about 1M-50M.

