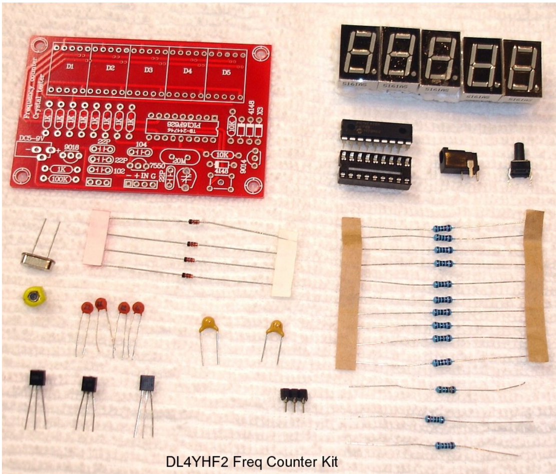


DL4YHF2 Frequency Counter & Crystal Tester Manual

By Steven Vagts

Z-100 LifeLine
Insert, Issue #131

DL4YHF2 Frequency Counter & Crystal Tester Description:

The DL4YHF2 Frequency Counter & Crystal Tester is sold as a kit and requires some experience in soldering skills. A soldering iron with a needle point tip would be very helpful here. The kit comes with no instructions, but most everything one needs can be found on the internet. However, I have assembled the kit, created a schematic, and included everything you need in this manual.

This is a five-digit frequency counter kit based on a PIC single chip 16F628 micro-controller, with a crystal oscillator measurement function, programmable frequency setting, and LED digital display. It is most commonly used to measure the oscillation frequency of a crystal or crystal oscillator.

Part/Model Number: YS9283, B8O8, and many others, but it is actually a generic, unbranded circuit board based on a circuit developed by Wolfgang Buscher, DL4YHF. I have also added the ability to test the four-pin crystal oscillator, so we will call the modified device the DL4YHF2 Frequency Counter and Crystal Tester.

All the parts are through hole components, so the kit is easy, if you have basic soldering skills, and simple to operate.

Please note, this item ships from an international seller. Expected delivery is 10-15 days.

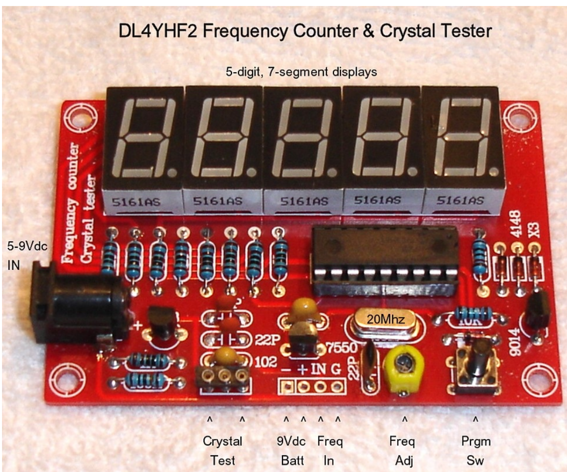
DL4YHF2 Features:

- Measures 1Hz-50MHz frequency and most of crystal oscillator frequencies.
- Can measure the oscillation frequency of most commonly used crystals and crystal oscillators.
- Large 5-digit LED display, easy and clean for readings.
- Five-digit precision resolution (e.g., 0.0050 kHz, 4.5765 MHz, 11.059 MHz)
- Automatically switches range, no need to switch manually, with different gate times.
- Can measure signals with amplitude up to 30Vdc.
- Programmable addition and subtraction frequency settings, for trouble-shooting radio transceivers and other equipment.
- Can select energy-saving mode, which will turn off the display if the frequency has not changed in 15 seconds.
- Powered by a universal USB 5V interface power supply, external 5-9Vdc AC/DC power supply or 9V battery

Specifications:

Color:	Red circuit board
Display:	Five seven-segment LED digits
PIC Type:	PIC 16F628 microcontroller
Power Required:	Universal USB 5V interface power supply, external 5-9Vdc AC/DC power supply (1.35x3.5x9mm) or 9V battery
Power Consumption:	The display draws a total current below 20mA and the PIC draws about 4mA at 20MHz
Frequency Range:	1Hz-50MHz
Crystal Test Range:	4MHz-48MHz
PCB Size:	80 x 53mm / 3.15 x 2.17"
Assembled Size:	80 x 55 x 7mm / 3.15 x 2.17 x 0.28"
Package Size:	11.5 x 9.5 x 2cm
Package Weight:	32g

Physical Layout of the Frequency Counter:



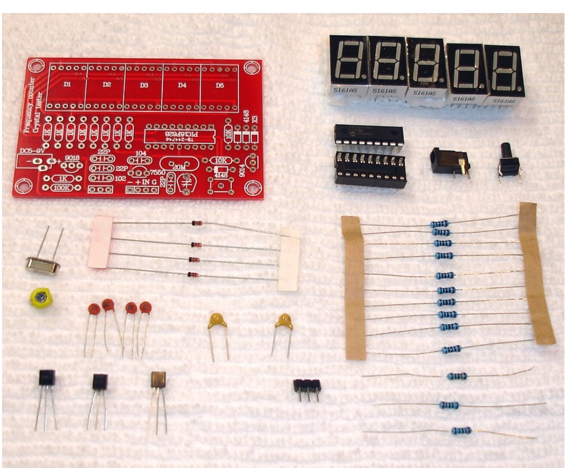
Power:

The DL4YHF2 Frequency Counter requires power from a universal USB 5V interface power supply, external 5-9Vdc AC/DC power supply or 9V battery. When the power is ON, the current is about 25mA.

The Frequency Counter has a press ON switch used for programming and adding or subtracting frequencies. More on this later.

Controls:

The DL4YHF2 has no power switch and comes ON when power is applied. It will switch itself OFF when there is no notable frequency change in 15 Seconds. It does, however, have one press ON switch used for programming and frequency addition or subtraction.



Assembly:

All the parts needed to construct the Tester are provided, except there is no parts list nor instructions. This manual will guide you through the rather simple construction and operation of this Counter.

A schematic is included at the end.

Assembly will require the following tools:

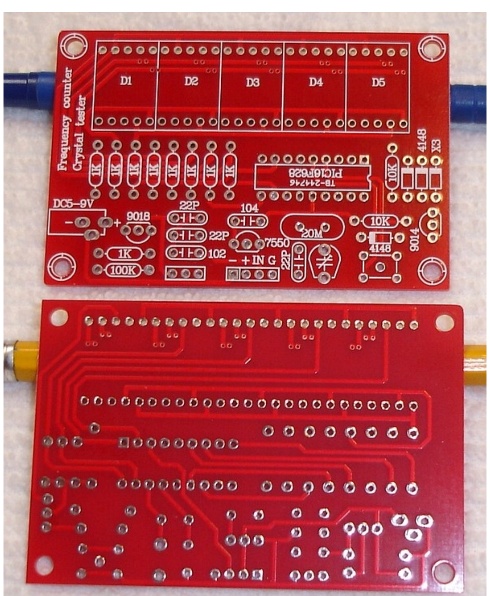
- Ohmmeter and Capacitor Checker (recommended)
- Needle tip soldering iron or gun
- Thin electronics solder w/flux core
- Fine solder wick or desoldering braid for mistakes
- Needle nose pliers
- Diagonal side cutters
- Phillip's head screwdriver
- 5-9Vdc AC/DC power adapter or 9Vdc battery

Before beginning construction, identify and compare the parts you received with the following list. I also suggest that you check those parts that you can with an ohm-meter and capacitor checker, if they are available. All parts are new; however, in the two Freqnency counter kits that I have constructed, I found I was missing a 3-pin header and had an extra push button switch. In the other China kits that I have constructed so far, I also found a shorted capacitor and a resistor included in one kit of the wrong value. A little extra time here to check components now, will save considerable time trying to troubleshoot a malfunctioning assembly.

Parts List: (Numbers in parentheses show component markings)

Part#:	**Component Description:**	**Number:**	
R1-9	1K ohm, 1/4W, 1%, Metal Film Resistor	9	[ ]
R10,R11	10K ohm, 1/4W, 1%, Metal Film Resistor	2	[ ]
R12	100K ohm, 1/4W, 1%, Metal Film Resistor	1	[ ]
C1,C2,C3	22pF, 20%, (220), Ceramic Capacitor	3	[ ]
C4	0.001uF, 20%, (102), Ceramic Capacitor	1	[ ]
C5	50pF, Yellow, Adjustable Trim Capacitor	1	[ ]
D1-4	Diode, 4148	4	[ ]
Q1	S9014 BJT-NPN Transistor	1	[ ]
Q2	S9018 BJT-NPN Transistor	1	[ ]
Q3	7550A PJET Transistor	1	[ ]
U1	PIC 16F628, 18-pin IC & 18-pin socket	1	[ ]
JK1	DC Jack, 1.35mm (ID) x 3.5mm (OD) x 9mm	1	[ ]
SW1	Normally Open, momentary ON switch	1	[ ]
P1	SIP Connecting Terminal, 3-pin	1	[ ]
	Main Circuit Board	1	[ ]

Assembly Procedures:



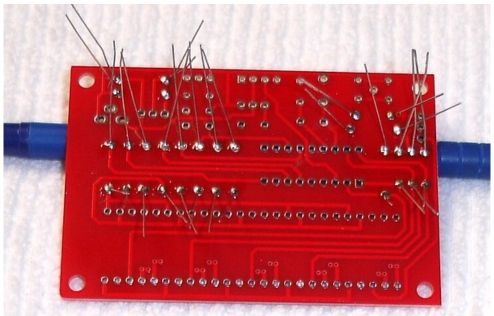
DL4YHF2 Bare Board - Front & Back

Begin assembly by inspecting the bare board. Get a feel for the layout of the parts, part numbers, and what is going to go where. The component side is the side with the silk screening, and while there may be some silk screening on the solder side, for our small kits, all the parts will be installed on the component side of the circuit board (top board in the picture).

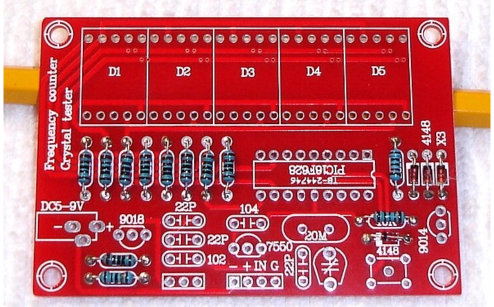
Hint: I always suggest installing those components with the lowest vertical profile first. This keeps the circuit board flat and stable for as long as possible during the assembly and soldering process. So, usually I start with any surface mounted components, while the board is empty and most stable.

Hint: Using a spare cotton towel under the circuit board helps protect the work surface and stabilize the board during soldering.

Note: All the solder pins on this board are adequately separated, however, if you accidently create a solder bridge across 2 or 3 pins, place solder wick over the solder bridge and carefully heat the wick only until solder flows into the wick. Take care not to overheat the component!

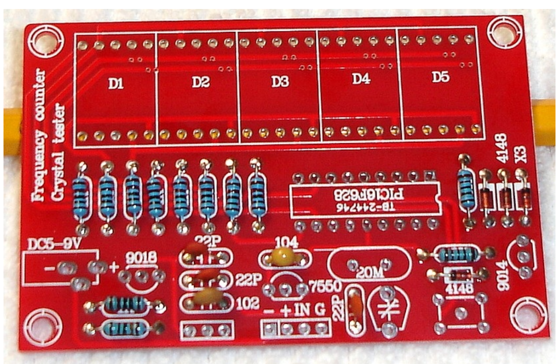


DL4YHF2 - Install Resistors & Diodes



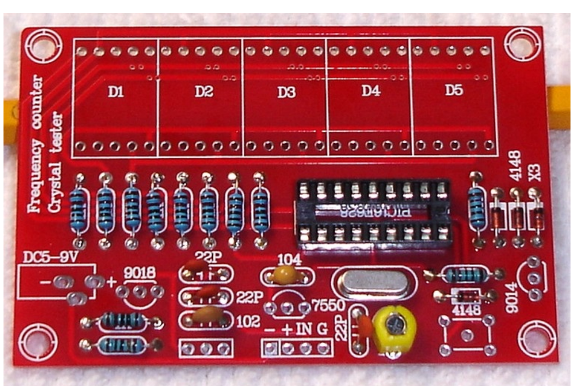
As this kit has no surface mounted components, begin with installation of the resistors. Install them one at a time, or insert all the resistors, bending the leads slightly to keep them in place. Then turn the board over and solder all the leads at once, clipping off the excess leads as you go.

For a small project such as this, the group method also ensures that all of the resistors are of the correct value, used correctly and in their proper location, BEFORE soldering any in place!



DL4YHF2 - Install small capacitors

Next, install the crystal (not shown) and capacitors (next in the height profile) and use the same techniques we used for the resistors; inserting them all, bending the leads to ensure they stay in place, then soldering and clipping the leads at one time, if you desire.



DL4YHF2 - Install other low components

Next, install the IC socket.

Hint: When you install multi-pin components, such as the 18-pin IC socket, always solder one lead at each end of the component first, check to insure the component is fully inserted in the board (not tilted to one side or one end is not fully seated), before soldering the remaining pins. It is much easier to fix a tilted socket with only one pin to heat to reposition the socket.

When installing the 18-pin IC socket, watch for the half-moon notch on one end! Install the socket so that this notch is over the similar notch symbol on the silk-screened circuit board. Do **not** install the IC until after the rest of the components have been installed and the solder side of the board has been inspected and cleaned.

Install the trim capacitor, flat side as shown on the silk screen and the three transistors. While in some kits, transistor leads may have to be rearranged, in this kit all the transistor leads are straight and installed as per the silk screen. Leave the transistor body about 1/4" above the surface of the board, bend the leads to hold the transistor in place, then solder.

We have some components left to install and some decisions to make, but it is time to visually check your work looking for solder bridges, parts with cold solder joints (meaning a poor connection, not having the same appearance of smooth solder flow as the other solder joints), or open, unsoldered joints. If you have an ohmmeter, check joints near each other for shorts.

Next, clean the solder side of the board. Many use a special flux cleaner product to clean the soldering side. Personally, I check each solder connection and use a fine screwdriver or dental pick to scrape away any flux residue, then use a toothbrush and a little water to remove the scraped residue.

When you are satisfied that the board is as good as you can make it, locate the five 7-segment LED displays. Note that there is a top and bottom! Make sure that when you install these, the decimal of each digit is located at the bottom right corner!

These digits usually come with a protective plastic film over the front. Remove these before you begin. Clearances between digits are tight, so if you fail to remove these, the digits will often not fit beside each other properly because of the overhanging plastic film.

For a similar reason, it is also recommended that you DO NOT solder in the digits one at a time. If one is slightly out of position, it will mess up the alignment of the remaining digits. Place all five digits into position on the board; double check that the film has been removed and the decimal is in the lower right of each digit. Pick the board up so the pins are fully expended through the board. Position your hands such that your fingers can trap some of the pins of each digit against the back of the board as you invert the board and rest the displays flat on the work surface.

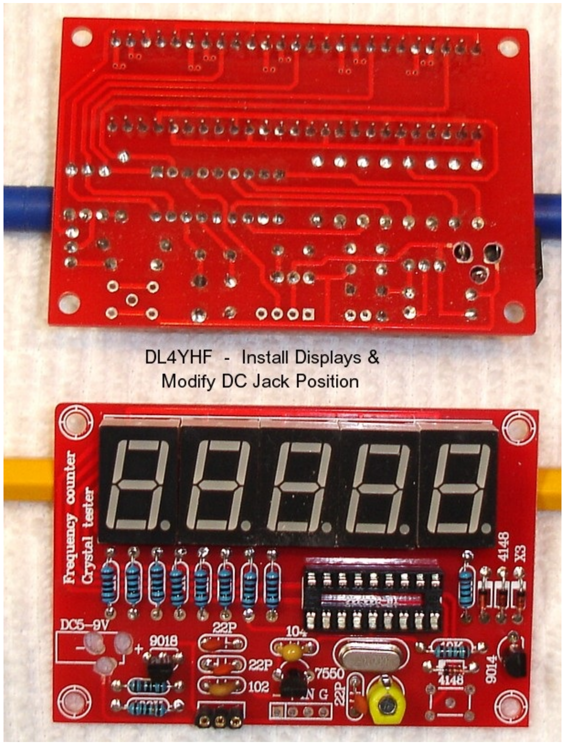
As with the IC socket, solder one pin on the top and the bottom row of each digit. Visually check that each digit is tight against the board and that the decimal is at the lower right. Once satisfied that all is good, solder the remaining pins. Depending upon the case you wish to use, you may or may not need to cut the digit leads. I'll mount mine to a wood board with sufficient clearance to leave the leads alone. The board will give some weight and physical stability to the finished counter.

Now it is time to decide what source of power you wish to use. This kit comes with a 1.35mm inside diameter x 3.5mm outside diameter male connector, a standard for 5Vdc connectors, especially for use with USB cables. This Counter, however, can be powered by up to 9Vdc, and can be powered by a 9Vdc battery. As all my other China kits came with a larger and more popular 2.1mm inside diameter x 5.5mm outside diameter x 9.5mm deep 5-9Vdc jack, I preferred using that DC jack.

If you would like to do the same, this requires severe modification to the circuit board. If not, solder in the given DC jack at this time, and skip the following section.

Changing the DC Jack to a Larger Size:

The kit comes with a DC jack that takes a 1.35 x 3.5 x 9mm plug. This size is smaller than that used in the other China kits I have, so I wanted to change it to the more common 2.1mm x 5.5mm jack size. Obviously, you will need to find the DC jack and the power cable you wish to use. There are lots of choices. The pictures show my changes.

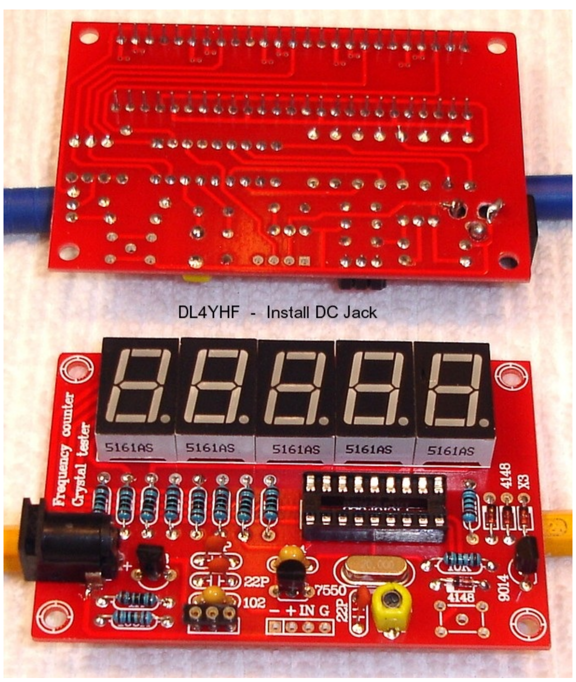


The first step is to mark and drill the new size holes required.

Hint: Do **NOT** just use the existing holes as a guide for the new holes; they will become too big and will not support the new DC jack very well. Do **NOT** try to drill the new holes with a large drill bit; the bit will drift into the existing holes. ALWAYS use a small bit to drill pilot holes where the new DC jack pins will be required. For mine, slightly less than about 1/16 inch outside the diameter of the existing holes. Drill the larger holes using the new pilot holes, maybe even using another intermediate size bit before the final bit size.

Test fit your new DC jack, then remove any burrs from the new holes.

Using a small knife, remove a portion of the insulation from the nearby wire traces as shown - as we have removed or destroyed the existing through hole solder pads, you will need bare copper to solder new wires. You must be able to attach all three jack lugs to the nearby traces - two on the solder side and one on the silkscreen (component) side.



Insert the new DC jack and bend the lugs to secure the jack to the circuit board.

Using small lengths of left over resistor leads, solder each of the three jack lugs to their nearby copper traces. Remember, there were two on the solder side and one on the component side.

Check all connections with an ohmmeter.

Hint: These traces are small and are difficult to properly solder, so use the ohmmeter to measure continuity from the jack's lug to some point along the trace away from the lug.

Install the 3-pin connector terminal and the switch (not shown).

Install the 18-pin PIC 16F628 IC, again watching for the location of the half-moon end, and the assembly is complete. Congratulations!

Operation:

Operation is simple. There is no ON/OFF switch. Plug in the 5-9Vdc power supply or a 9v battery and the device turns ON. If you don't have the proper connector for the 5-9Vdc power supply, the battery terminals, '-' and '+', are part of the 4-pin SIP holes at the center bottom edge. You could attach a 9Vdc battery connector here. However, it seems to be a weird place to locate the power connectors, instead of beside the power connector. Using the nearby Frequency Input 'IN' and 'G' often will cause wear on the thin wires of the battery supply and will likely cause them to break off.

I also found that those four blank holes were a bit odd for the design. I would have much preferred the little screw terminals of the earlier testers that I tried. But perhaps this was intentional to keep your options open - to attach the board to some larger piece of equipment. Some other options include maybe just using a SIP header that you can use to attach some clips, or maybe it might be best to solder a pair of test leads with clips on the ends and wrap the leads around the device when not in use. I'll address my solution in the last section.

When I first turned on the DL4YHF2 Counter and tried a crystal, the counter worked just fine. However, when I attempted to read a frequency from a frequency generator, all I got was some strange digits on the display. According to what I read on the internet, if your counter shows a strange frequency initially, enter setup mode to set the frequency offset to zero (it sometimes happens that PIC programmers don't erase the EEPROM where the frequency offset is stored). That solved my problem.

Calibration:

Before operating for the first time, calibrate the unit with a frequency from a known generator (I used the 50KHz frequency from one of the earlier kits I completed, the XR2206 Frequency Generator). Attach the frequency generator to the 'IN' and 'GND' terminals and input a known frequency.

Adjust the yellow adjustable capacitor so that the display is equal to the value of the signal frequency.

Hint: Do NOT use a metal tool; the metal affects the display readings. If you don't have a proper plastic diddle stick, you can whittle a small diameter dowel to use. If you must use a small screwdriver, note the frequency, adjust the capacitor either left or right slightly, then remove the screwdriver. Note the new frequency, and adjust again. Repeat as necessary, removing the screwdriver each time.

Note: The capacitor is fully rotational, meaning it can turn in any direction without limit. During rotation through 360 degrees, it is simply moving one small metal plate against another to vary the capacitance. That means there are two positions for each capacitance value (except maximum and minimum) - while increasing or while decreasing during rotation. Either position will work fine.

Display Ranges:

The display range automatically switches to give the maximum readout accuracy. The gate time is also selected automatically as listed in the following table:

Freq Range:	Display:	Gate time:	Decimal Point:
0 - 9.999 kHz	X.XXXX	1 sec	flashing (which means "KHz")
10 - 99.99 kHz	XX.XXX	1/2 sec	flashing
100 - 999.9 kHz	XXX.XX	1/4 sec	flashing
1 - 9.999 MHz	X.XXXX	1/4 sec	steady (which means "MHz")
10 - 50.00 MHz	XX.XXX	1/4 sec	steady

Note: Displays 0 (zero) in the fourth digit if no signal.

How It Works:

This counter is using 7-segment displays with a common cathode. So all the pins of the displays are connected the same, except the common cathode at pin 8. Pin 8 differentiates the display digits from each other.

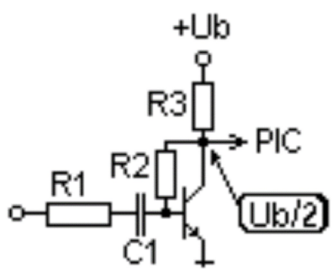
The DL4YFM was originally designed for four LED displays, but our unit drives five. For this reason, the 'single zero' (if no input signal is present) is not displayed in the 5th, but in the fourth digit.

The PIC firmware always drives a fifth digit as the least significant digit, if the measured frequency is above 10 kHz. Because there was no free output pin available on the PIC 16F628 to drive another digit, a logic combination from the first four digit multiplexer outputs was used (digits 1 ... 4). When all digit multiplexer outputs are passive (digits 1...4 = high), the optional fifth digit is driven.

The diode between the emitter and ground is used because without it, the transistor may conduct unwantedly if its base-emitter threshold voltage (typically 0.5 to 0.6V) was less than the forward voltage of the other diodes in this stage (which will be about 0.6 to 0.7V for a 1N4148).

Pre-amplifier:

This counter has NO input protection from accidental over voltage. For this reason a pre-amplifier is often desirable and could be attached as a separate board and used between the device generating the frequency we care to measure and the input to our board. One suggested pre-amplified is described here:



The pre-amplifier consists of a single HF silicon transistor, such as a cheap BF199 from the junk box. It works well up to 30 MHz and with reduced sensitivity up to 50 MHz. Make the connection from the collector to the PIC counter input (pin 3 = T0CKI) as short as possible, because every pF of capacitance counts.

With a BF199 in the pre-amp, R2 is 27 kOhm (+/-), and R3 must be as low as 560 Ohm to achieve the necessary bandwidth. The DC voltage at the collector and the voltage across R3 should almost equal - if not, adjust R2.

If you want to feed the Counter with a TTL signal, a pre-amplifier is not required because the TTL voltage is controlled by definition and saves about 4mA of current draw.

If the maximum frequency in your circuit is below 10MHz, you may increase the value of R3 and R2 by the same factor (say R3=1.2k, R2=56k) to also save some current when using the counter in a battery-powered device. R1 sets the input impedance and also the sensitivity. With R1=330 Ohm, the prototype required an input voltage of 600 mVpp (peak-to-peak) at 40MHz and 150 mVpp at 15MHz. If you need a higher input resistance, add a FET buffer before the bipolar transistor, or use a fast integrated comparator as the input stage.

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 receivers 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 signal RA5 (pin 4 of the PIC 16F628) is used to switch from normal counter mode into programming mode. Usually the level on RA5 is high because it is connected to the supply voltage via pullup resistor (10k to 22k). By pulling RA5 low using the momentary switch, the firmware is instructed to use the currently measured frequency as the new offset value. In other words, you must apply the offset frequency to the counter's input, wait until the value is displayed correctly, and then enter the programming mode as explained below.

Programming mode:

To enter programming mode, press and hold the programming key (the switch) 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 key press), you will be taken back to the main menu to select **'Add'** or **'Subtract'**.

PSave: Turns the power-saving ON. 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). This was added in May 2006 for battery-powered equipment like QRP transceivers.

NoPSV: Turns the power-saving OFF.

Note: There may be more menu items than shown here, but the principle remains the same.

The frequency offset values are saved as a 32-bit integer numbers in the PIC's data EEPROM (at the EEPROM's first four memory locations, high-byte first, low-byte last). If you have no signal generator to produce the offset frequency for programming, or cannot tap the BFO frequency of your homebrew shortware receiver, you can enter the offset value with a suitable PIC programmer (like DL4YHF's WinPic). Use a scientific pocket calculator to convert the frequency (in Hertz, positive or negative) into a hexadecimal number, and enter this value in the PIC programmer's EEPROM DATA memory window. If you use WinPic, enable the HEX editor before typing the values into the memory window. Some examples:

4.194304 MHz:	Add= 00 40 00 00	Subtract= FF C0 00 00
4.433619 MHz:	Add= 00 43 A6 D3	Subtract= FF BC 59 2D
0.455000 MHz:	Add= 00 06 F1 58	Subtract= FF F9 0E A8
10.70000 MHz:	Add= 00 A3 44 E0	Subtract= FF 5C BB 20

If the subtracted offset is higher than the counter's input frequency, the result of the subtraction is negative. The frequency counter makes the result positive before displaying it. This way, you can use the counter also in receivers where f_IF = f_RX + f_LO, or f_RX = f_IF - f_LO which means increasing LO frequency means decreasing RX frequency (the counter will seem to "run backwards" but that's no mistake).

Example for DL2YEO's 30 meter band QRP transceiver: f_RX = f_LO - f_IF = 14.314 MHz - 4.194 MHz = 10.120 MHz, which is the calculation inside the counter (f_LO=measured input, f_RX=display value, f_IF=programmed offset). If you don't need the 10-MHz-digit on the display, set the offset to -14.194 MHz instead of -4.194 MHz. This will give better display resolution, so you only need 4 digits (f_RX=10.120 MHz will be displayed as 120.0 kHz, which is sufficient because the receiver's tuning range is only 20 kHz anyway).

Some commonly used IF frequencies can be recalled from the "Table" menu, so you don't have to measure or enter them yourself. In many cases, there is a BFO for the last mixer (at the output of the IF amplifier) which produces a frequency close enough to the desired value.

How it works:

Basically the program runs in an endless loop, with the exception of the initial lamp test, programming mode, and power-saving mode which are not explained here.

In the main loop the following steps are performed:

Prepare a coarse frequency measurement for the automatic range switching: Program the asynchronous prescaler to divide by 64, so the highest external frequencies can be detected (theoretically 64 MHz, but this exceeds the PIC's specification).

Count the input pulses for 1/16 second, using the PIC's TIMER0 module in counter mode. During this time, the display multiplexer keeps running. In fact, the counting loop takes exactly 50 microseconds, including the multiplexer routine. 1250 counting loops result in a gate time of 1/16 seconds. In the source code, this is done in the subroutine 'count_pulses'.

Decide which prescaler and which measuring interval should be used, depending on the coarse frequency measurement.

Reprogram the counter's prescaler so the divided input frequency is below 1 MHz (which is the maximum input frequency for the hardware counter, if the PIC is clocked with 4 MHz). If the coarse measured frequency is way below 1 MHz, the prescaler is turned off to get the best possible frequency resolution.

Count the pulses during the measuring interval (alias gate time), which is 0.25, 0.5, or 1 second. During this time, the display multiplexer keeps running. Overflows of the 8-bit timer register ("hardware") are counted by software in two other 8-bit registers, so the effective pulse counter has 24 bits (8 hardware bits plus 16 software bits while counting).

Gate time finished -> stop counting pulses.

If the hardware prescaler was active while counting, multiply the pulse count with the prescaler ratio so we don't have to care for the prescaler setting in the following steps.

If you know a bit about assembler programming:

The multiplicator is always a power of two, so instead of a multiplication, the pulse count value (now expanded to 32 bit) is shifted left which is much easier on a PIC.

If the gate time was 0.5 seconds, multiply the pulse count by 2; if the gate time was 0.25 seconds, multiply the pulse count by 4. The result is the input frequency in Hertz, no matter which prescaler ratio or gate time was used. Like in the previous step, this "multiplication" is in fact a simple bit-shifting operation.

(Optional) Add the programmed frequency offset. If the result is negative, make it positive.

Split the frequency into eight (!) decimal digits. This is tricky with a PIC, see the source code. It is realized by repeatedly subtracting powers of ten from the 32-bit frequency value, beginning with ten millions (because the highest, theoretically possible frequency is 64 MHz).

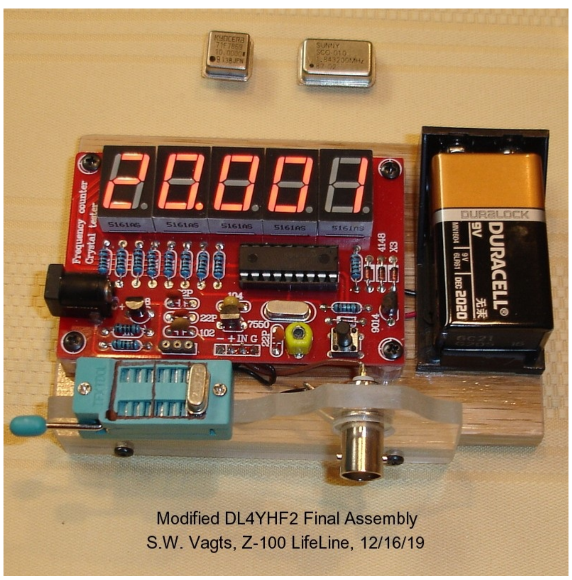
Skip leading zeroes, and insert a decimal point after the kHz- or MHz digit (the kHz-point is ANDed with a blink flag)

Beginning with the first non-zero digit, convert five digits from binary code into seven-segment-patterns, and copy the result into the "display registers". The display multiplex routine which is executed while counting will write these registers to the LED display of the next main loop.

Poll the 'programming function' input ("RA5"). If this digital input is low, enter programming mode (not explained here). If not, go to step 1 to begin the next measurement.

Final Assembly:

For my finished Frequency Counter and Crystal Tester, I made some modifications to include a frequency probe attachment (the BNC connector in the lower right of the photo) and the capability of testing the many popular 4-pin crystal oscillators.



Modified DL4YHF2 Final Assembly
S.W. Vagts, Z-100 LifeLine, 12/16/19

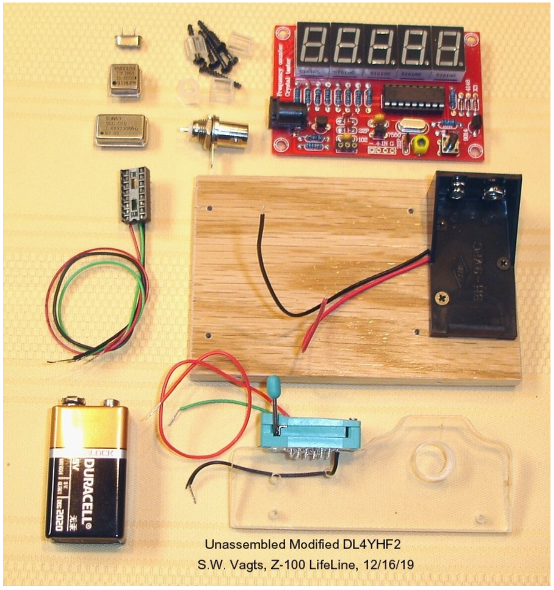
Crystal oscillator units work similar to the 2-pin crystal units, except they have four pins in a square or rectangular can (see the upper images in the photo). The third pin is required for 5Vdc power and the fourth pin (with the dot) has no connection.

This image shows the operational unit testing a two pin crystal. While a 14-pin low profile socket would work fine for the oscillators, the two rows of pins are too far apart for the crystals. So you would still need to use the SIP connector mounted on the board.

So I chose to use a 14-pin zero insertion force (ZIF) test socket for two reasons. First, I wanted to use a socket that would handle multiple insertions easily, and second, I wanted to use the same socket for crystal oscillators and the wide slots would better accommodate the two pin crystals, such as shown.

The ZIF socket is glued to the top of the plexiglass side with a glue gun. I’ve also drawn the outlines of the two pin crystal and the 4-pin square and rectangular oscillators.

The circuit board is screwed to the wood board base with 1/4" pieces cut from a BIC (tm) pen tube used as spacers. I also attached a 9Vdc battery holder, which is more convenient than keeping a wall wart handy.



This photo shows my circuit board and the other parts used to complete my tester.

I included the spare 16-pin low profile socket that I used to check the operation of the tester on the full-size rectangular and half-size square crystal oscillators.

As I already mentioned, the problem was that the 2-pin crystals would still need to use the board-mounted socket. The wide slots of the test socket would easily accommodate both crystals and the two standard sizes of the crystal oscillators.

The standard crystal oscillators need a 5-volt power source. This can be obtained from a tap off pin 14 of our PIC socket on the board.

Finally, the oscillators that I generally have on hand have been used on circuit boards and have their pins clipped fairly short. In the photo image on the right, I had to sand down the test socket's top layer to about half thickness to be able to insert the oscillators in the socket.

To sand down the test socket, remove the screw at the end of the socket away from the handle. Do NOT remove the screw near the handle, as getting the socket back together is somewhat tricky, if you remove that screw. Push the handle down to close all the socket slots.

I used a disk sander mounted to my drill. Holding the socket with a pair of pliers at the handle end, sand the top plate to about half its original thickness - lose about 1/16".

Hint: Sanding causes dust and melted plastic to stick to the sides of the openings in the top. Use a sharp knife or razor blade to trim away all the excess plastic from the sides of the socket and holes. A stiff paint brush or old toothbrush is great for removing any remaining dust. Do NOT lift the handle to open the slots until all dust has been removed with the brush.

You will also need to remove about the same thickness (1/16") from the length of the screw that was removed. I found that the screw was fairly soft and could be clipped with a good pair of diagonal side-cutters.

I have also drawn the outlines of the three case types on the 14-pin test socket. The 5Vdc Vcc must be provided at pins 11 and 14 of the test socket. Ground is at pin 7, and the frequency signal is found at pin 8 of the socket for all three case types.

As the threaded length of the BNC connector isn't sufficient to fit through the 1/4" plastic, you will also need to carefully drill the BNC connector hole, first with a 1/2" center-point wood bit to about half the thickness of the plastic (from the front) to fit the shoulder of the BNC connector, then with a 3/8" center-point wood bit to fit the threaded shaft of the BNC connector. Check the hole with the connector to ensure it fits properly. However, remove the connector until you are finished preparing the rest of the plexiglass.

I mounted the test socket on the top edge of a 3.5" x 1.5" x 0.25" plexiglass. The dimensions need not be precise, but the 1/4" plastic makes it easy to glue the socket with a glue gun. The physical shape of the plastic is also left to your imagination. A 1/2" sanding drum on your drill will quickly shape the plastic to whatever shape you finally decide on. For storage purposes, I wanted the top of the test socket to match the height of the plastic over the BNC connector.

Once the shape of the plastic is satisfactory, drill the holes for mounting the plastic to the side of the wood base and for the ground wire of the test socket. I found it easier to solder the necessary wires to the test socket, then mount the socket to the plastic using a glue gun. Next mount the BNC connector. Finally, solder all the wires to the circuit board, including the two from the battery holder, if used.

Mount the circuit board to the wood base using spacers from an old BIC (tm) pen or similar plastic tube, and 5/8" #4 screws (mine came from old VHS tape shells).

Testing Crystals and Oscillators:

Mount the full size crystal oscillator using pins 1, 7, 8, and 14 of the test socket, with the dot closest to the handle (pin 1).

Mount the half size crystal oscillator using pins 4, 7, 8, and 11 of the test socket, with the dot closest to the handle (pin 4).

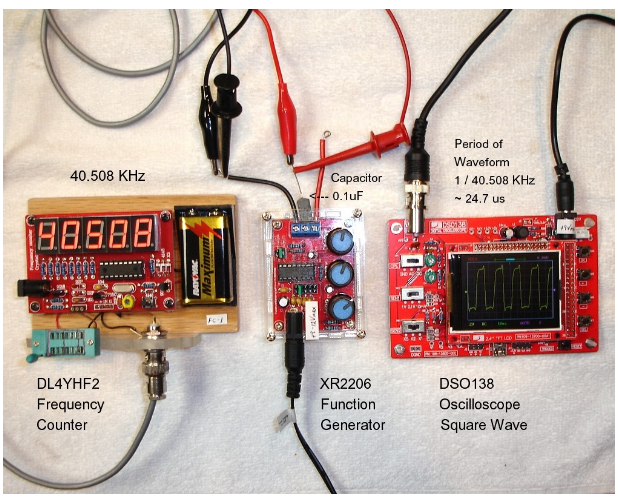
Mount the two-pin crystal cases between pins 7 and 8 of the test socket. They do not require separate power, but remember, these are bidirectional, meaning they can be mounted in either direction.

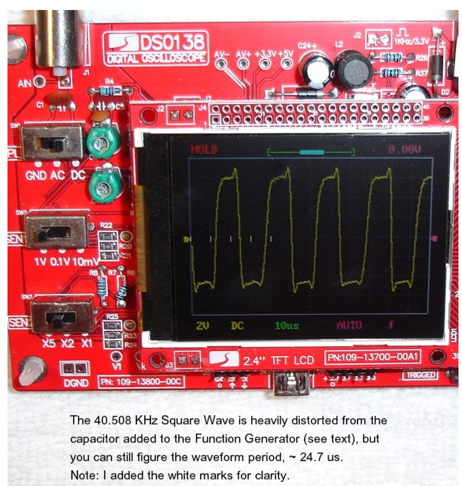
Troubleshooting:

If your counter shows a strange frequency initially, enter setup mode to set the frequency offset to zero (it sometimes happens that PIC programmers don’t erase the EEPROM where the frequency offset is stored).

December 7, 2020 Update:

While writing my article on using the DSO138 Oscilloscope Kit, found an issue with our DL4YHF2 Frequency Counter becoming unstable when displaying certain frequencies from my XR2206 Function Generator (another kit). So, let me describe the experimental setup, then the solution.





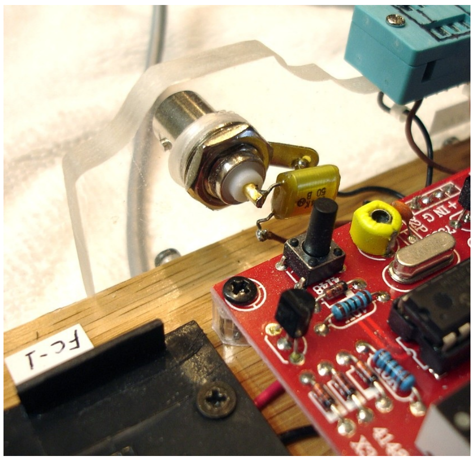
The 40.508 KHz Square Wave is heavily distorted from the capacitor added to the Function Generator (see text), but you can still figure the waveform period, ~ 24.7 us.
Note: I added the white marks for clarity.

As you can see, I’m using our DL4YHF2 Frequency Counter, our XR2206 Function Generator (set for a 40.5 KHz square wave), and our new DSO138 Oscilloscope to show the waveform. However, when I first put this circuit together, I was very disappointed with the DL4YHF2 Frequency Counter. While the oscilloscope showed that the waveform was nicely square and stable, the frequency counter would not stabilize, if I got a frequency at all. Yet, when I used the frequency counter to test an oscillator, it was nicely stable at the correct value.

Looking at the circuit diagram for the DL4YHF Frequency Counter, I saw that the circuit for the crystal & oscillator tester used a 0.1 uF capacitor to eliminate any unwanted DC voltage. But for the input of external frequencies, a capacitor was not included. Perhaps a cyclic DC voltage was confusing the counter? So, on a whim, I included a 0.1 uF (104K) capacitor (type is not critical) at the output of the generator.

The capacitor fixed the unstable counter, as you can see - a nice solid 40.508 KHz display, but the waveform was distorted by the charging and discharging capacitor.

I moved the 0.1 uF capacitor to the input of the frequency counter, between the input jack and the circuit board, as shown...



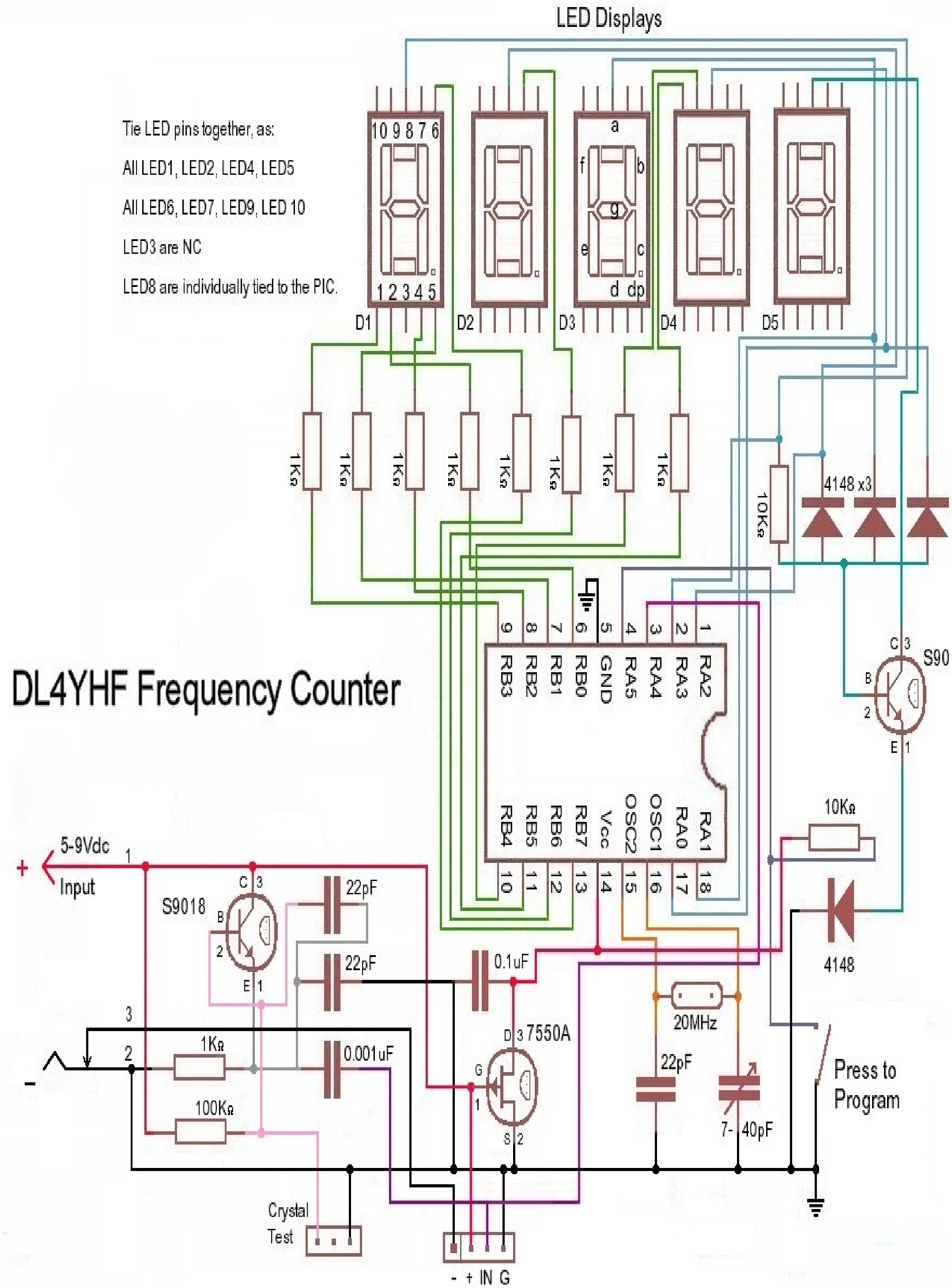
Both the Frequency Counter and the Oscilloscope were now happy!

Please report any errors or suggestions to the author by email at: z100lifeline@swvagts.com.

Thank you.

PIC 16F628 Pin Description:

Pin#	Symbol:	Type:	Description:
1	RA2	Out	Signal to LED Digit 2, pin 8
2	RA3	Out	Signal to LED Digit 1, pin 8
3	RA4	In	Signal IN from 4-pin connector
4	RA5	In	Signal from momentary ON switch
5	GND		Ground Pin
6	RB0	Out	Signal to all LED digits, Pin 2
7	RB1	Out	Signal to all LED digits, Pin 5
8	RB2	Out	Signal to all LED digits, Pin 4
9	RB3	Out	Signal to all LED digits, Pin 1
10	RB4	Out	Signal to all LED digits, Pin 9
11	RB5	Out	Signal to all LED digits, Pin 10
12	RB6	Out	Signal to all LED digits, Pin 7
13	RB7	Out	Signal to all LED digits, Pin 6
14	Vcc		Positive Power Supply (5-9Vdc)
15	OSC2	In	20MHz Oscillator Input 2
16	OSC1	In	20MHz Oscillator Input 1
17	RA0	Out	Signal to LED Digit 4, pin 8
18	RA1	Out	Signal to LED Digit 3, pin 8


LED Displays
Tie LED pins together, as:
All LED1, LED2, LED4, LED5
All LED6, LED7, LED9, LED 10
LED3 are NC
LED8 are individually tied to the PIC.
10 9 8 7 6
1 2 3 4 5
a b c d dp e f g
D1 D2 D3 D4 D5
1KΩ 1KΩ 1KΩ 1KΩ 1KΩ 1KΩ 1KΩ 1KΩ
10KΩ
4148 x3
S9014
DL4YHF Frequency Counter
RB3 RB2 RB1 RB0 GND RA5 RA4 RA3 RA2
RB4 RB5 RB6 RB7 Vcc OSC2 OSC1 RA0 RA1
10KΩ
4148
5-9Vdc
Input
S9018
22pF
22pF
0.1uF
7550A
20MHz
22pF
7- 40pF
Press to Program
1KΩ
0.001uF
100KΩ
Crystal Test
- + IN G