

[<- Back root](#)

Replacement Display for the Yaesu FT-290R, FT-690R and FT-790R

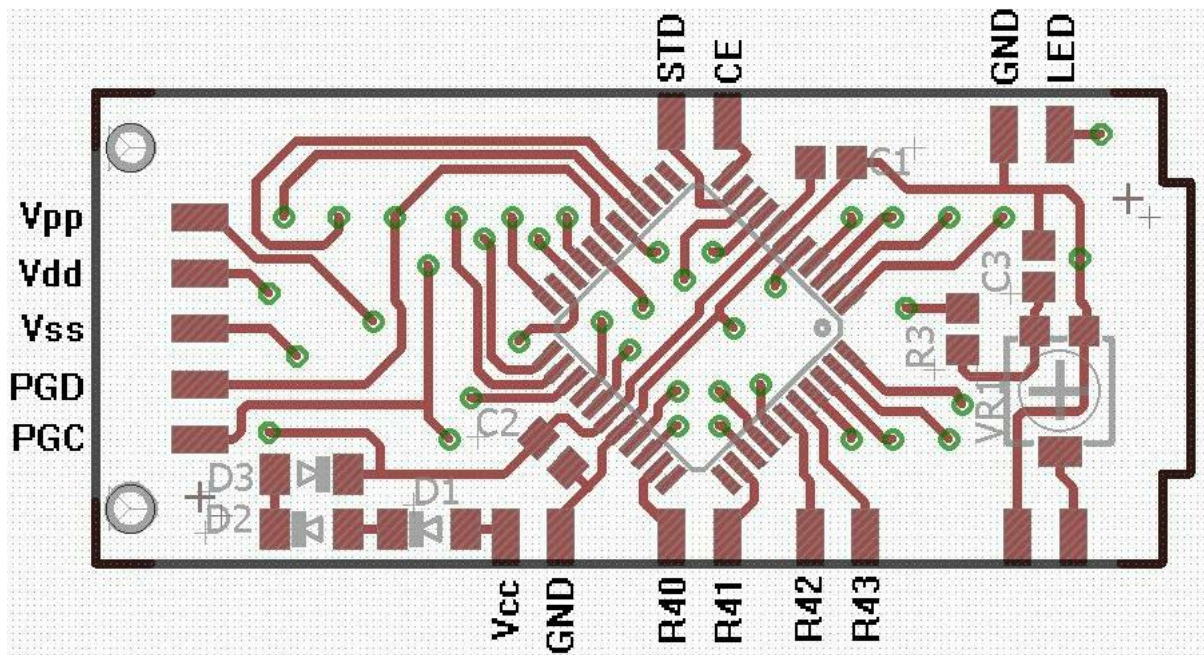
So I mentioned that I [reverse-engineered the FT-290R display](#) and designed a replacement. It's PIC based, uses a 16(L)F1937. Here's the [schematic](#). It's not pretty -- it exists to get a netlist to do the PCB layout.

I had a few 16LF1937 lying around so I added three diodes to drop the 5V down to 3.3-ish. If you use a 16F1937 you can substitute a jumper from D1 anode to D3 cathode.

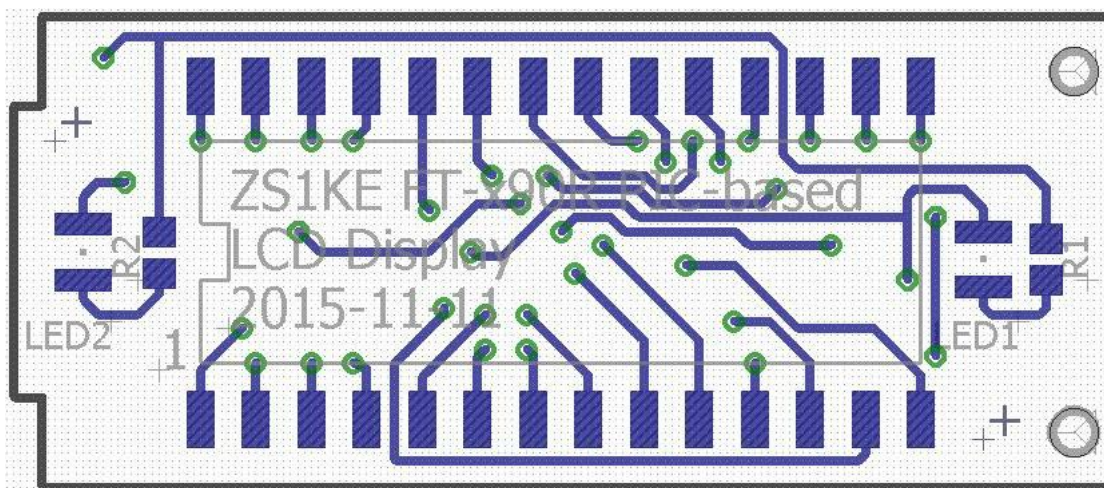
The GND connection should be connected to GND, i.e. chassis ground, and not the Vss normally supplied to the display board via the connector. The easy way to do this is to remove R08 and to replace R19 with a jumper.

Bill of Materials (BoM)

- PIC16(L)F1937 -- pre-programmed, or programmable in-circuit if you have a PICKit
- LCD Display
- C1, C2: 100nF Ceramic 0805 SMD
- D1, D2, D3: 1N4148 MiniMELF (if using the PIC16LF1937)
- LED1, LED2: I tried using a 4mm right-angle LED but they're not bright enough. Fit a little [filament bulb](#).
- R3, C3, VR1: Not currently used



Back layout and connections.

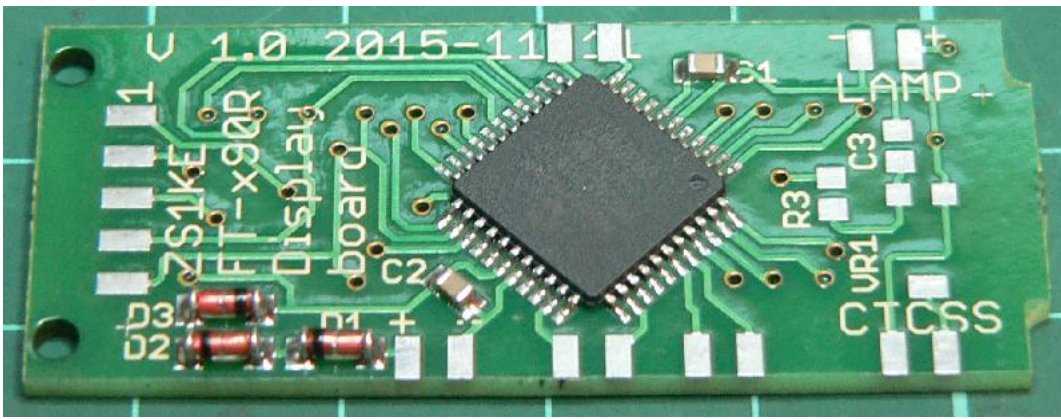


The front layout is straightforward.

Here's the [Eagle board file](#).

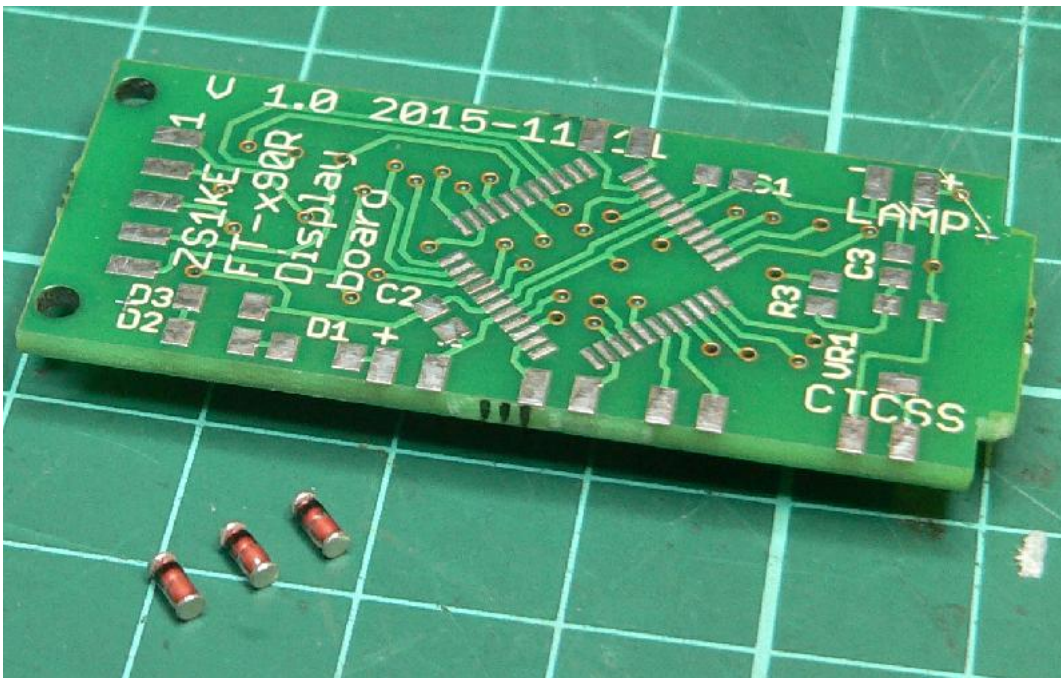
Assembly

PIC



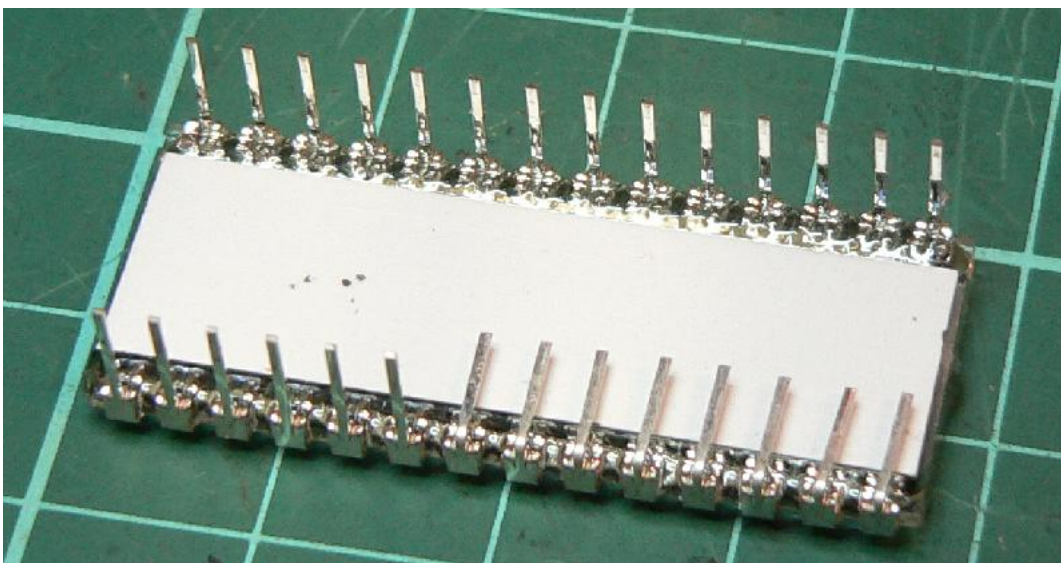
Pin 1 is not indicated on the PCB but it is indicated on the layout above. With the text the right way 'round pin 1 faces to the right.

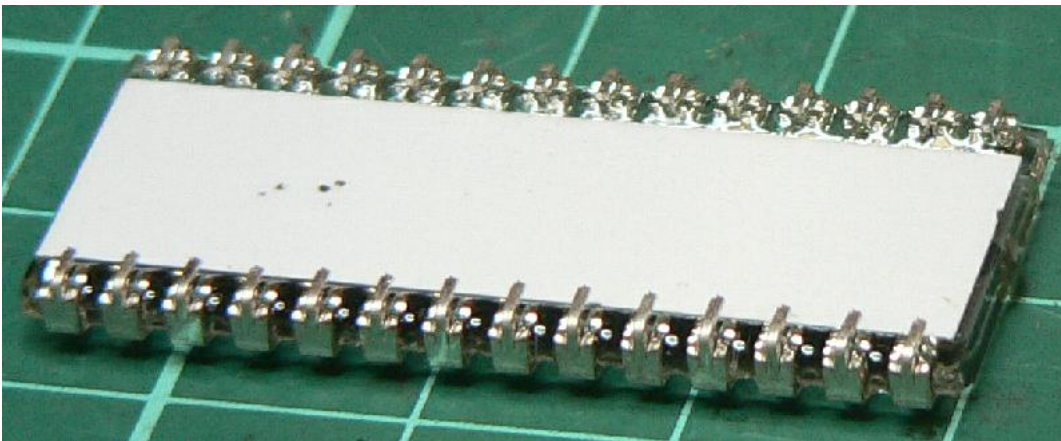
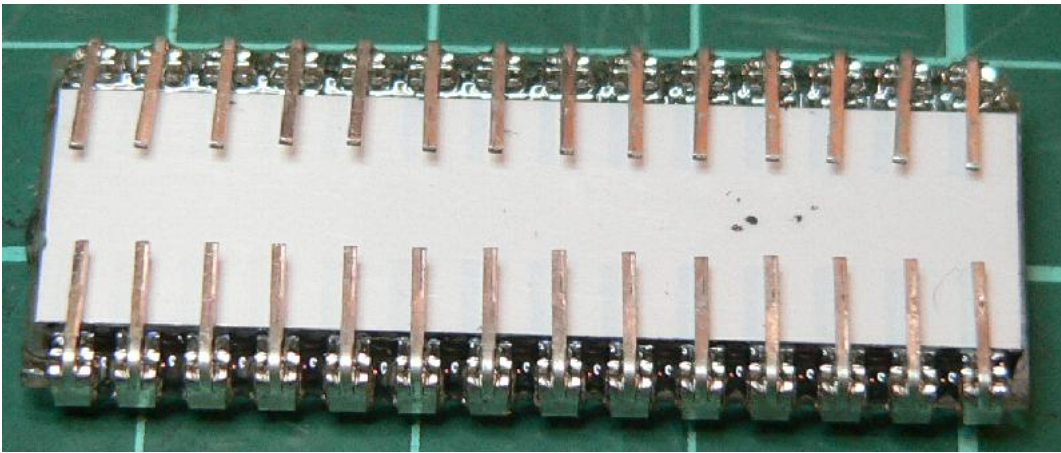
MiniMELF diodes



The MiniMELF diodes are tricky to fit. I ended up using masking tape to hold them still while soldering.

LCD

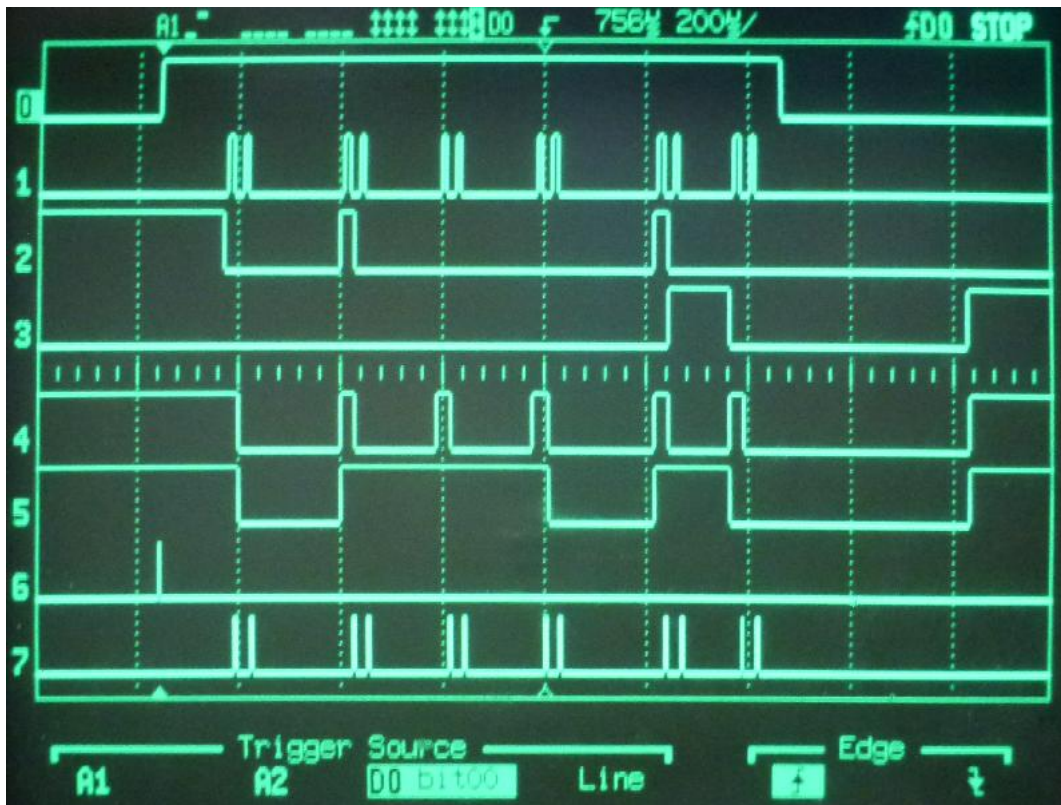




The legs of the LCD module have to be bent at right angles as close to the body as possible.

Be careful not to overheat the LCD while soldering. It's probably best to fit the LCD last.

Software



Top to bottom, left to right

0011 0000 "0"
 1011 0001 "1."
 0011 0001 "1"
 0011 0000 "0"
 1011 0101 "5."
 0010 0000 The rightmost three bits wo

Here's a logic analyser capture of the signals from the processor to the display.

CE goes high, then STD is pulsed 12 times, clocking in 4 bits each time (traces 6 and 7 are debug outputs to show that I'm detecting the CE rising edge and the STD pulses correctly).

```
// enable falling edge on STD = RB2
IOCBN2=1;
IOCBP2=0;
```

```
// enable rising and falling edges on CE = RB3
IOCBN3=1;
// enable interrupts
IOCIE=1; // Interrupt on change (port B)
GIE=1; // Global int on
```

So I get an interrupt on the rising edge of CE, on each falling edge of STD, and on the falling edge of CE.

Here's where all the work happens.

```
void interrupt isr()
{
    static int index=0;
    static char disp[12]={0,0,0,0,0,0,0,0,0,0,0,0};

    PIR1=PIR2=PIR3=0;
    if (IOCIF) // STD or CE changed
    {
        if (IOCBF3==1) // CE
        {
            if (CE == 1) // On the rising edge of CE, reset the index
                // into the array where we store the digit
                // values (4 bits per entry)
            {
                index=0;
            }
            else // it's the falling edge - display what we stored in the array
            {
                display(disp);
            }
        }
        if (IOCBF2==1) // On the falling edge of STD, take the four
            // bits and put them in disp[++
        {
            if (index < 12) // because it always pays to check
            {
                disp[index] = (IN3<<3) | (IN2<<2) | (IN1<<1) | IN0;
                index++;
            }
        }
    }
    IOCBF=0; // clear IOCIF via IOCBF
}
```

I might be calling display() too often, but the code seems to work so...

2019-11-27: Pushed the [source code](#) to github.

ZS5DF

I [replaced a display](#) for ZS5DF's. It's a pain.

Radcom

Dave Crump G8GKQ springboarded this information to [build an LED-based replacement display](#) which was published in the December 2016 [Radcom](#).

PS: From the schematic it looks as if the FT-230R uses the same display. Dunno if it's different mechanically.

 Hit Count hits since 2016-01-28. [Back](#) (This page last modified 2019-11-27)