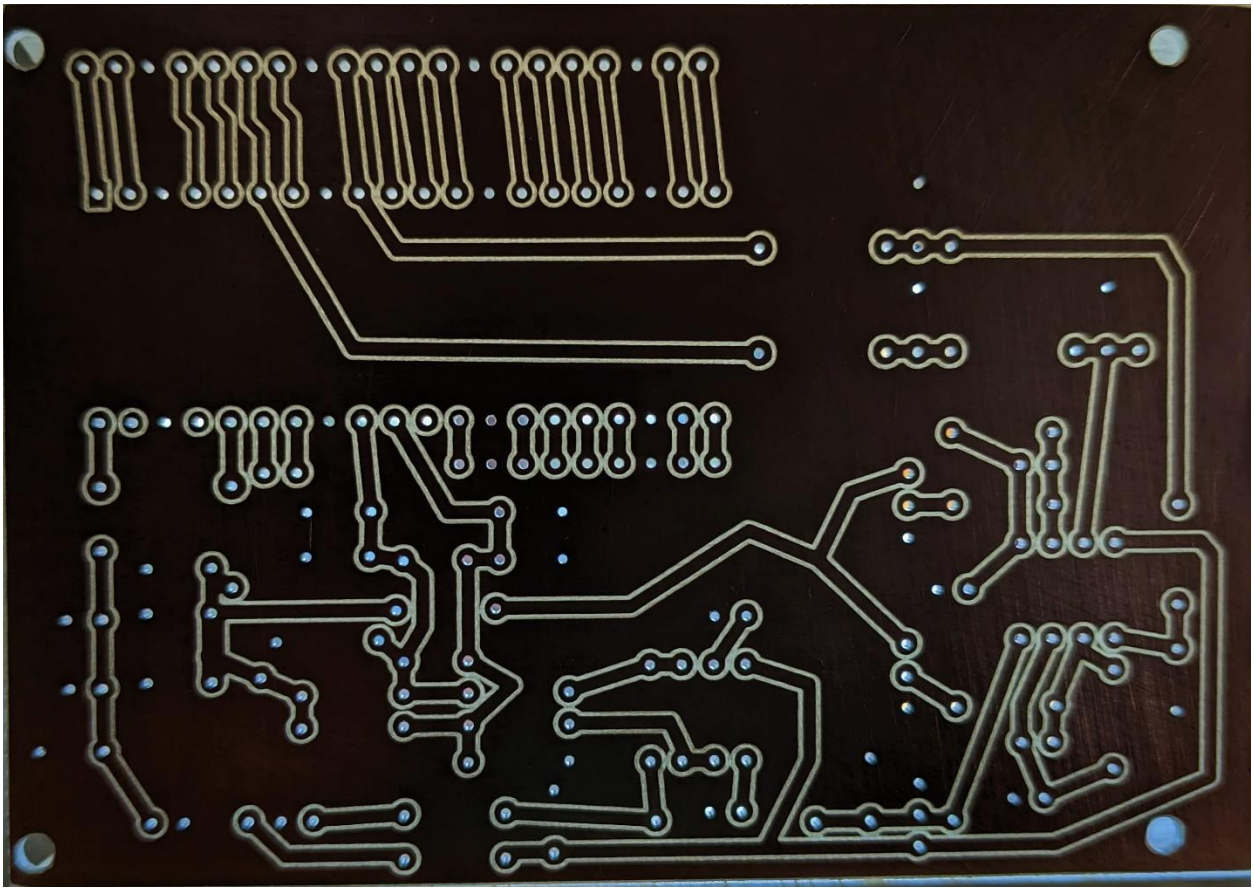
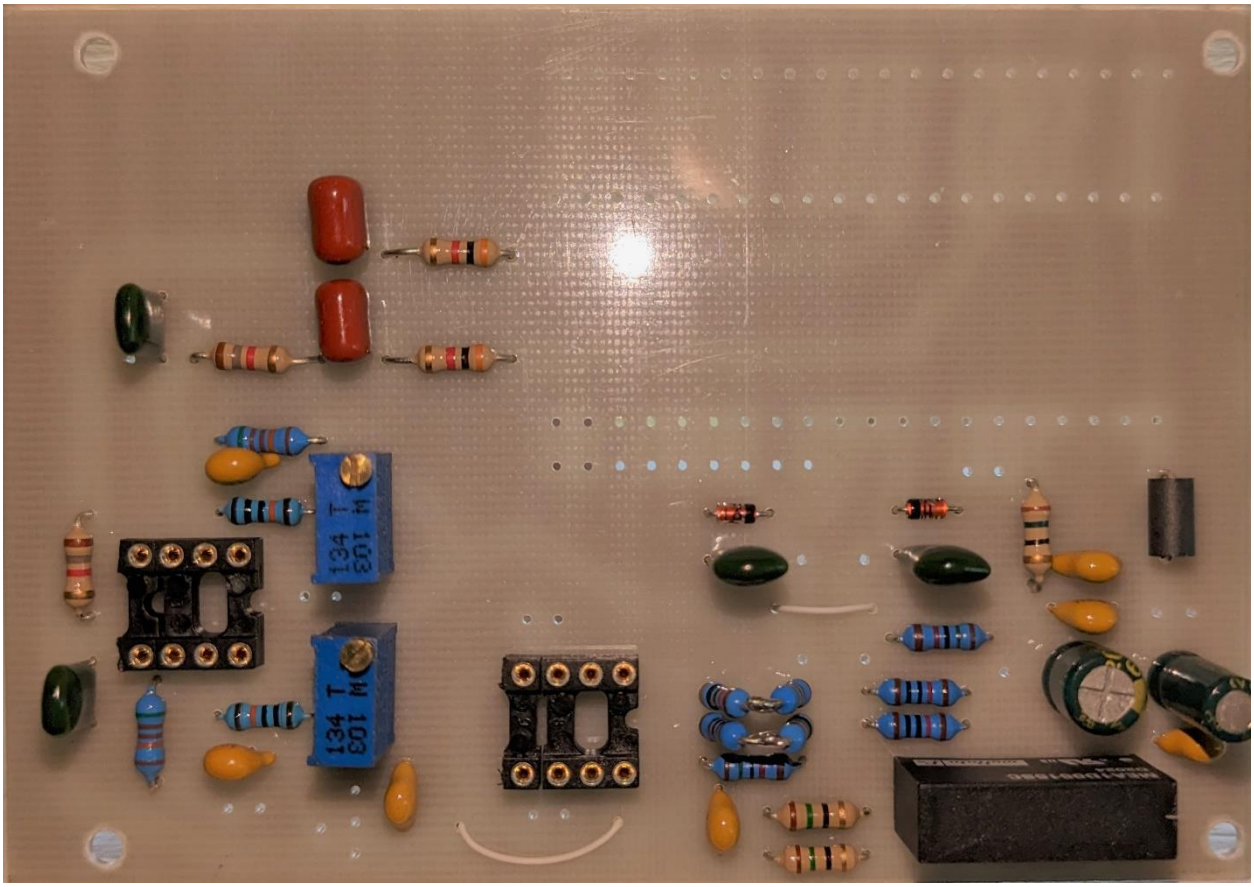
Some Building Tips



After milling and cleaning the board, use backlighting to check for burrs shorting across the milled copper traces.



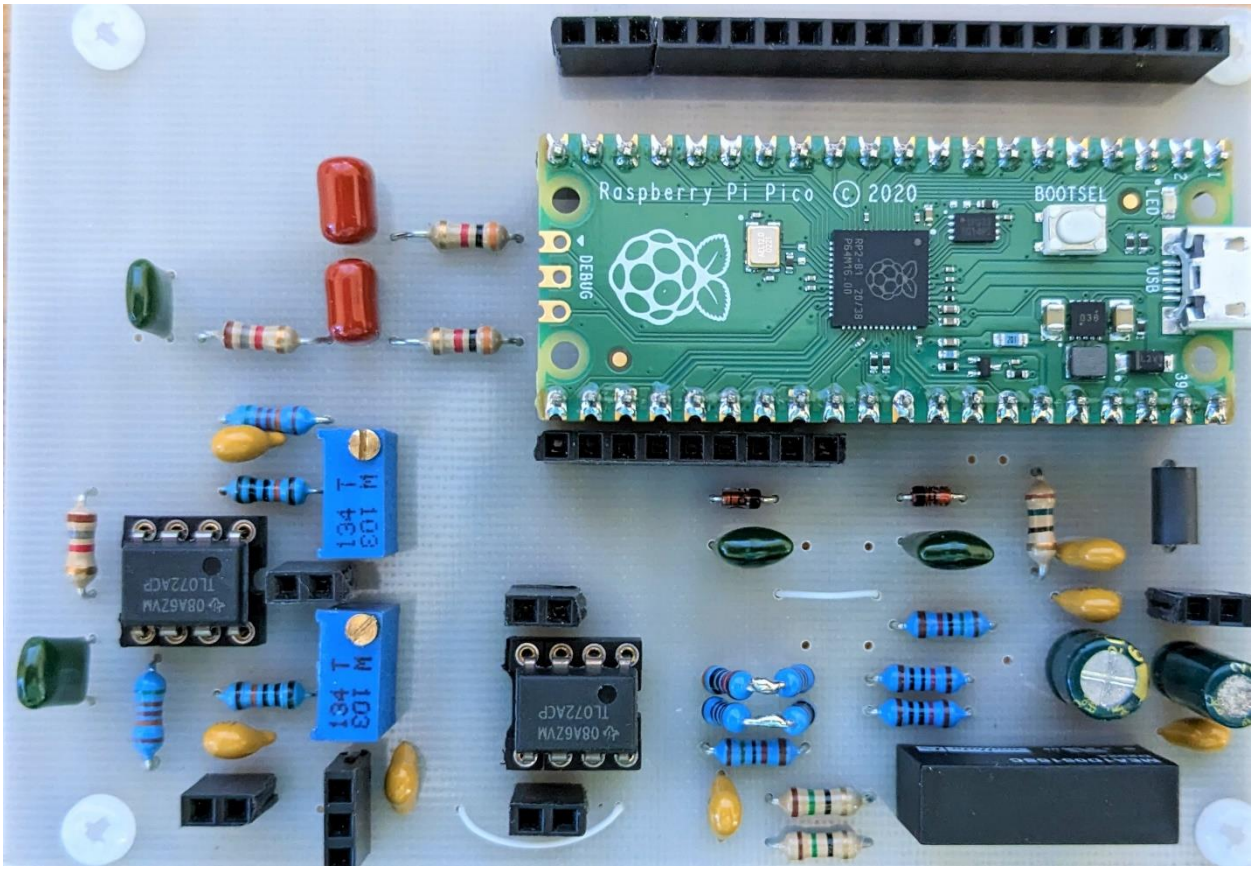
The board loaded with components (less female connector strips).

Incorrect 1.8K resistors were installed for the 18K resistors. DAC noise was excessive and this tipped me off to the wrong parts. You may also note the 8-pin machine tooled sockets are made of cut sections of 6-pin sockets. That's what happens when you run out of the correct parts.

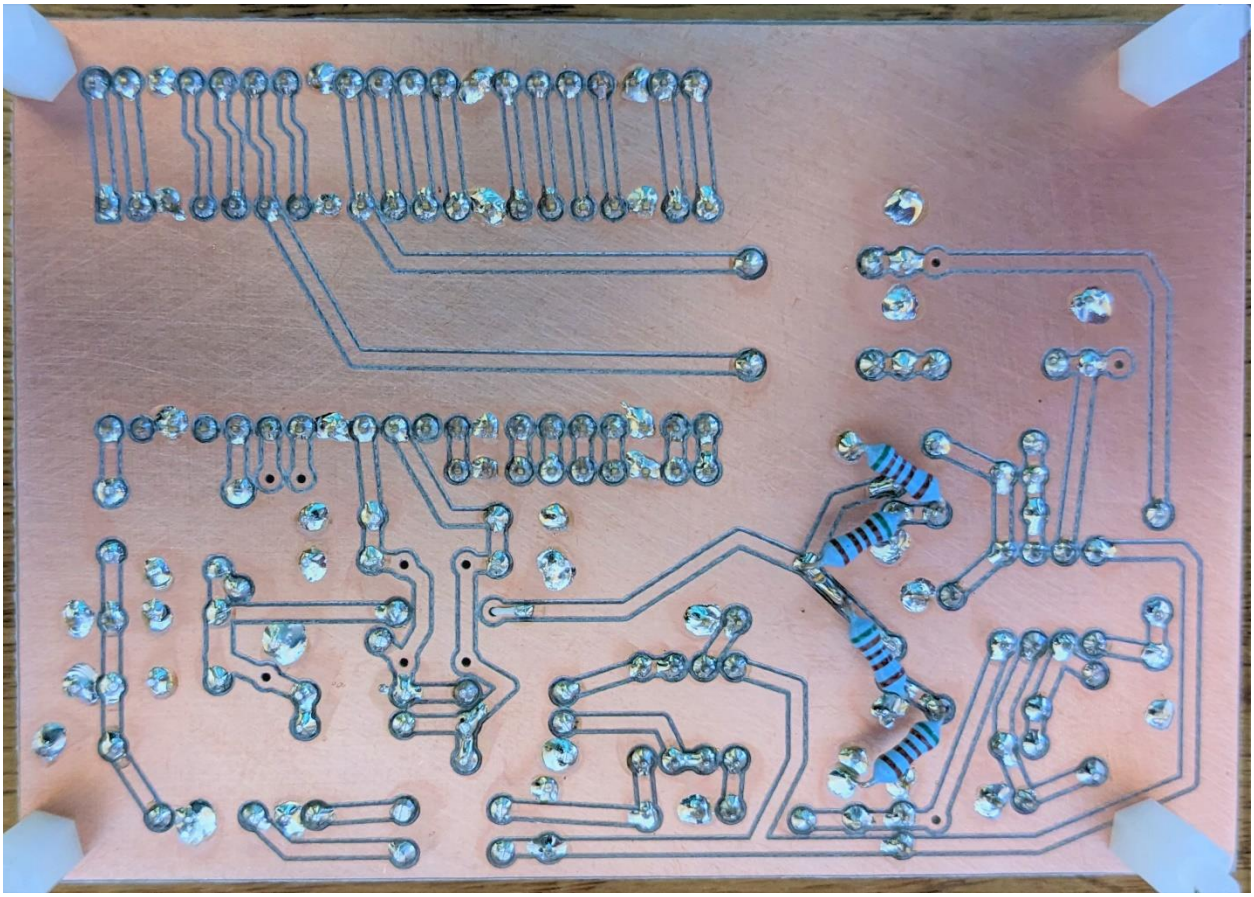
You may note the 5K pots are 10K pots. Another parts shortage. The fix is shown on following pages.

The 31K resistors are actually 30K and 1K resistors connected in series. You could even use a 27K and a 3.9K set of resistors.

Don't forget the two jumper wires.



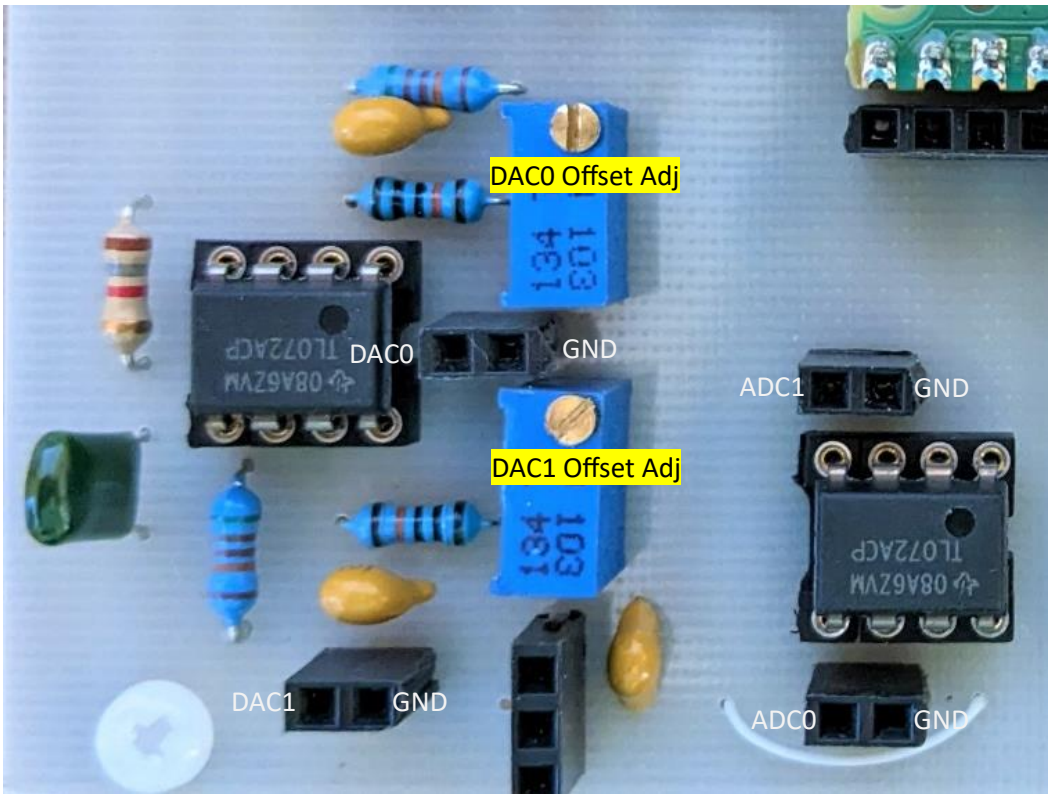
The fully loaded board. The female connector strips are cut and trimmed from 40-pin stock. Use nylon screws and standoffs for board attachment.



The fix for using a 10K trimmer pot is to solder a 5.1K resistor from the wiper to ground and a 5.1K resistor from the wiper to the 3.3V trace. If you only had 10K trimmers, you could just use them as is. The DAC voltages would be lower by about 2.5%.

Note that the nylon standoff at the bottom right corner bridges ground and the -15V trace. Using a metal standoff would result in a short. You could also drill a hole farther away from the trace.

Looks like I missed a two pin female strip for pins 34 and 35. Always inspect the card for solder bridges, incorrect parts and missing parts.



Once everything looks good, connect a USB cable to the board and your computer. Drop the **Pico_block_compiler_i.uf2** file into the Pico file folder. Use Device Manager to find your serial port number. Then open a connection to the serial port in your terminal app or use simpleCRT.exe.

Ground inputs ADC0 and ADC1.

Measure the voltage on the DAC0 pin. Using the DAC0 offset potentiometer, set the voltage to zero. Measure the voltage on the DAC1 pin. Using the DAC1 offset potentiometer, set the voltage to zero.

Type in the following program:

```
clr
dac in ch0
dac in ch1
adc ch0 a
adc ch1 b
prt a b
end
set dt 0.2
set avg 128
set max 10000
set ch1 1
set in 0
set fmt 3
```

With the ADC inputs grounded, use the cal command to zero the offset. Then use the run command to start reading the ADCs. The program will print the ADC0 and ADC1 readings every 0.2 seconds.

Hit <Esc> to stop the program.

Use the set command to change the DAC voltages:

```
set in 10

run
```

This will change both DACs to around +10V. You can connect the ADC0 input to the DAC1 output and the ADC1 input to the DAC0 output. The printed voltages will be close to the set value. The voltages will not be exact but should be around ±1% of the correct value.

Try some of the example programs in the *block_programming_pico.pdf* document.