

## Scopy - Oscilloscope and Logic Analyzer

# Analog Front-End Design 2

*If you have suggestions for improvements or would like to share your own designs for a Scopy analog front end then please head to the [Scopy forum](#).*

For this design we'll support a single voltage range of -1 to 1. This is similar to Design 1 and requires two extra resistors to set the required gain (Design 1 only attenuated the signal. Overall gain was not required). Given that the magnitude of our input voltage range is 2 (-1V to 1V) and the voltage range of the Pico ADC is 3.3 (0V to 3.3V) then the required gain is 1.65.

Even though we're amplifying the signal we still need the voltage divider in front of the op-amp so that the output can be level shifted (see Design 1 for an explanation of level shifting and Vref).

You'll see from the schematic below that we're giving the Rf and Rg pairs of resistors the same values. This simplifies the calculation of the required values of Rf and Rg and also the value of Vref.

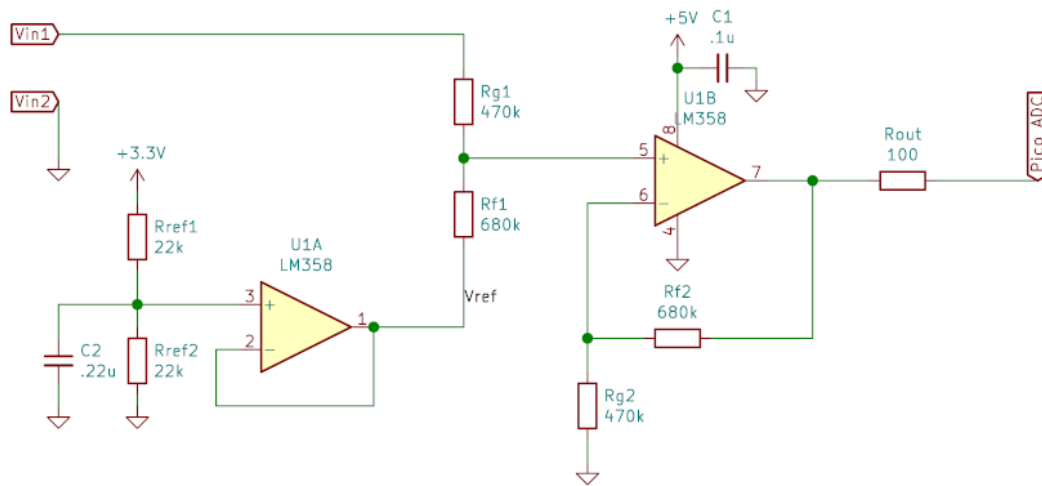
In this configuration the gain is simply  $R_f/R_g$  and the output is level shifted by Vref. So for our input voltage range of -1V to 1V,  $R_f/R_g$  is 1.65. Applying that to our input range gives -1.65V to 1.65V. The output voltage thus needs to be level shifted by 1.65V and so we set Vref to 1.65V.

For this design we won't be too precise when choosing the resistor values, let's try 680k and 470k for Rf and Rg respectively. The ratio of these values is 1.45 and not the 1.65 that we wanted but that's OK if we're happy with the reduced gain.

To generate the required Vref of 1.65V we can just use a voltage divider between 3.3V and GND with equal values for resistors Rref1 and Rref2. The actual value you choose doesn't matter too much (I don't think) but I've chosen a fairly high value of 22k to keep the current low.

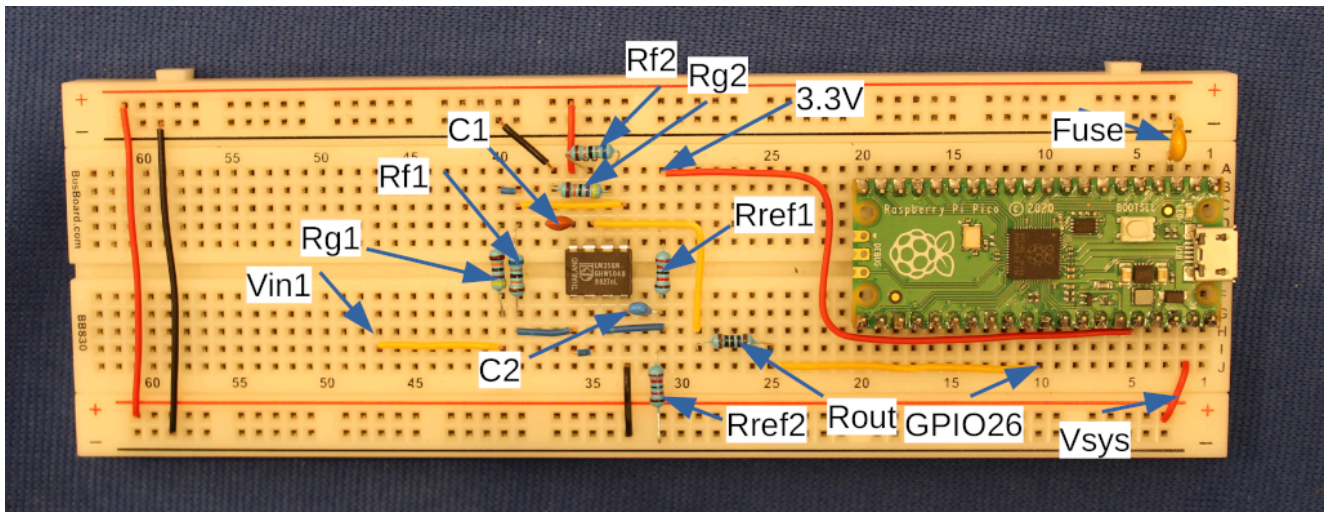
*Please note that input voltages outside of the -1V to 1V range can result in voltages at the RP2040 ADC pins that are higher than the maximum rating of 3.3V.*

## Schematic



## Construction

Assemble the circuit as per the schematic and breadboard image below.



*Caution. When the LM358 non-inverting input is floating the output might be a higher voltage than the maximum recommended voltage of the ADC pins ( $V_{IODD}+300\text{mV}$  according to the Pico datasheet). To prevent possible damage to the Pico you might want to only connect the ADC pin to the front end once your signal source is connected to  $V_{in}$ . Having said that, I've applied voltages of over 4V to the ADC pins without any apparent damage to the Pico (with a 100R current limiting resistor in series). Future designs will add over-voltage protection so that you won't have to worry about this.*

## Calibration

Once you've assembled the circuit make sure you haven't done anything stupid like shorting  $V_{sys}$  to GND (when I did that it resulted in the insulation on the jumper wire melting but luckily no damage to the Pico or my phone. Now I use a low current resettable fuse between the Pico breadboard power rail).

Go to the voltage range settings in the app (to get there tap the channel badge at the bottom of the screen to bring up the channel menu and then tap Settings and Voltage Range(s)) and enter -1 and 1 for the Min and Max voltages respectively (for the default range).

Chances are that the voltage displayed in app won't be exactly the same as the voltage applied to Vin. This is because of a combination of factors including:

- we didn't use resistors that would give us a gain of exactly 1.65
- the actual values of the resistors are probably not exactly the same as the nominal value
- the input voltage offset of the op-amp

To fix this, follow the instructions for [vertical calibration](#).

*The author makes no warranty, representation or guarantees regarding the suitability of this design for any particular purpose. Nor does the author assume any liability arising out its use and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages.*

---

## See Also

[Documentation Index](#)

[Scopy on GitHub](#)

[Using the App](#)

[Scopy Forum & Support](#)

[FHDM Store](#)

---

## Scopy - Oscilloscope and Logic Analyzer

Scopy - Oscilloscope and Logic Analyzer



[fhdm-dev](#)

Scopy is an oscilloscope and logic analyzer powered by your Android phone/tablet and Raspberry Pi Pico or Pico W.