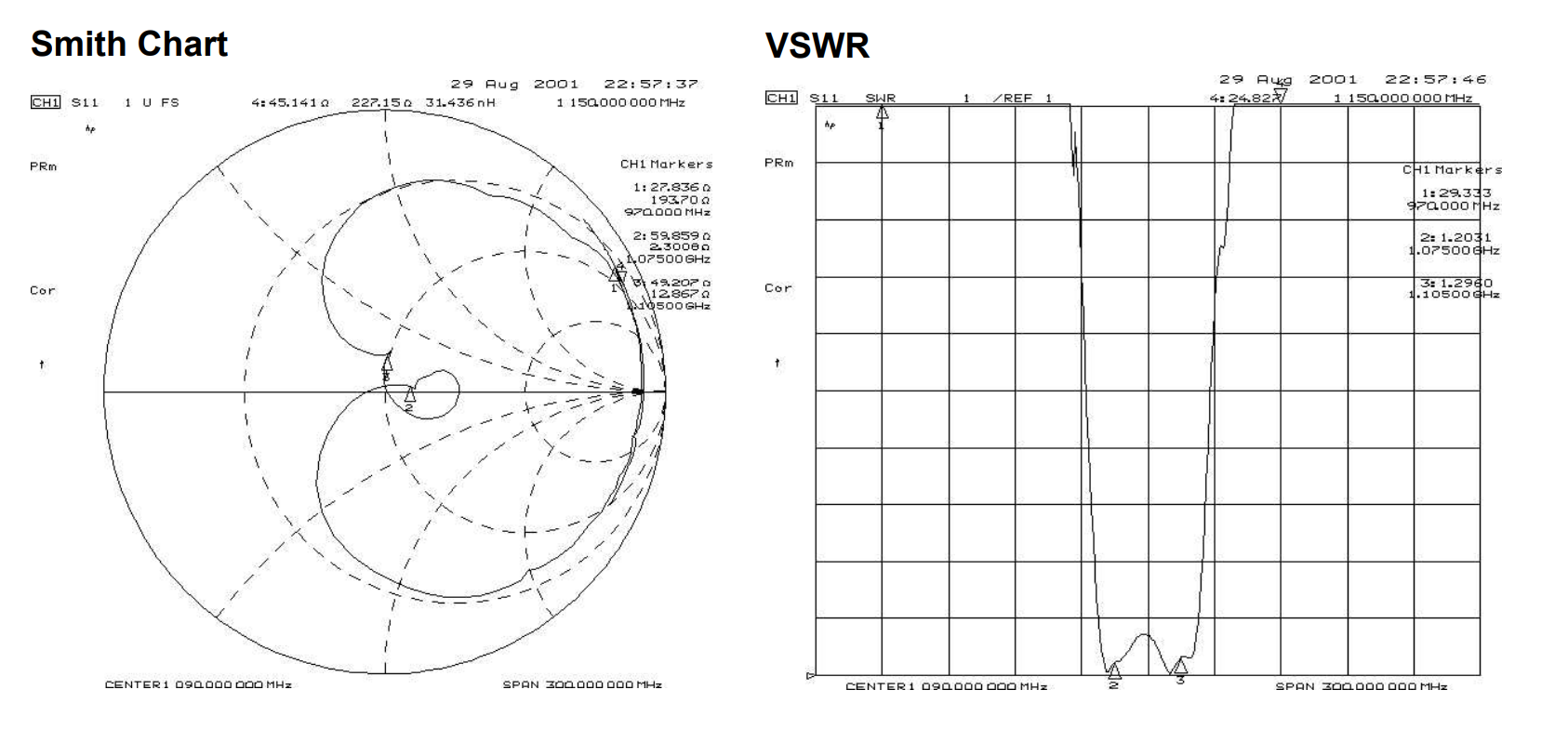
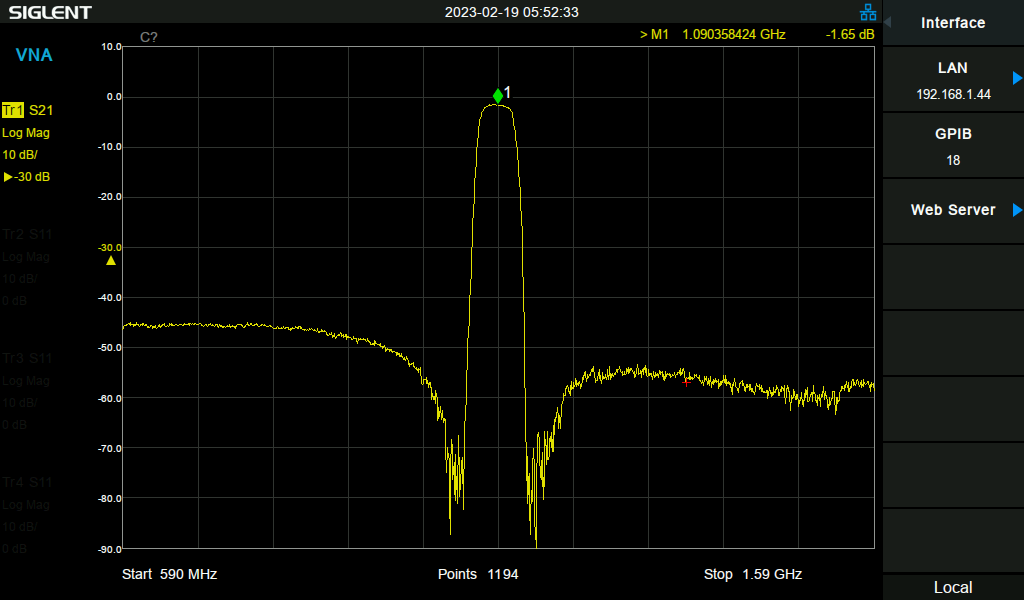
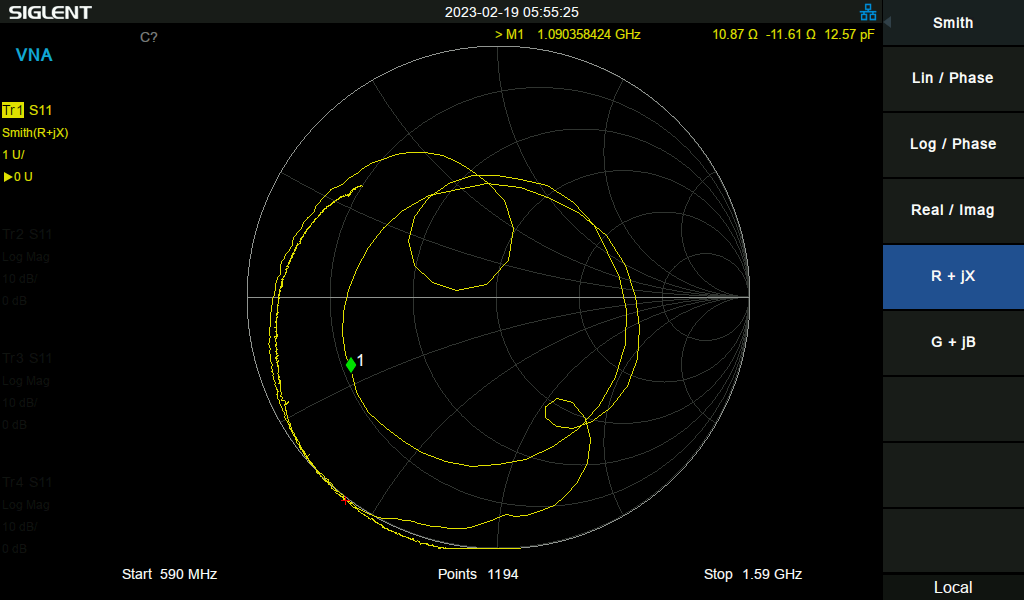
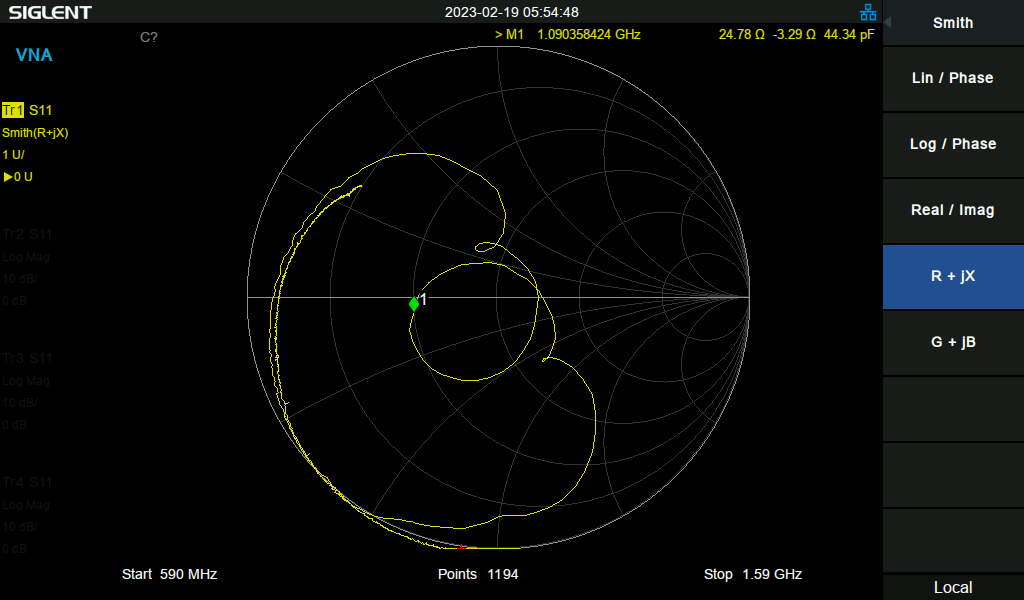
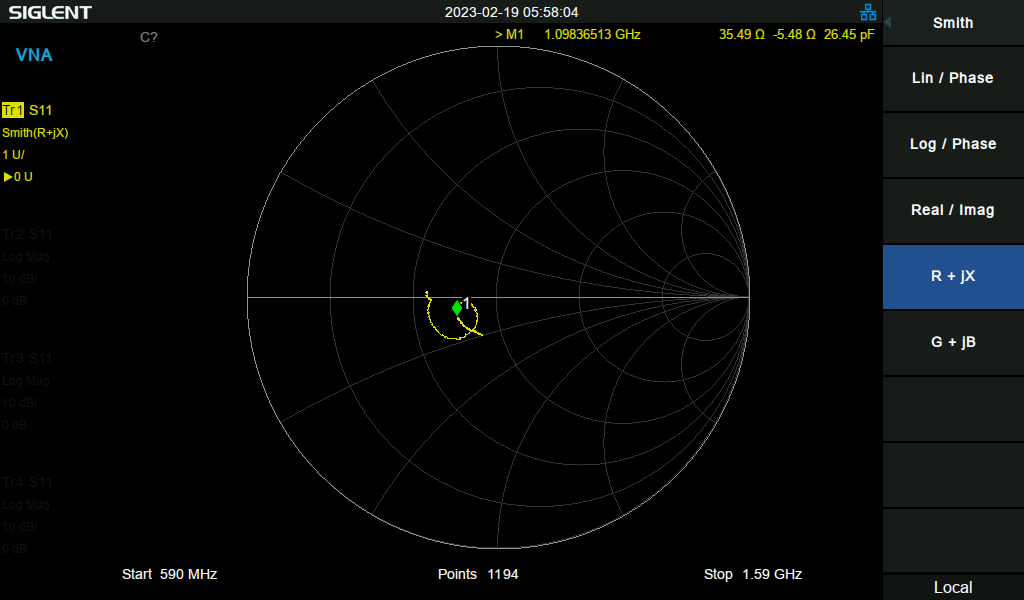
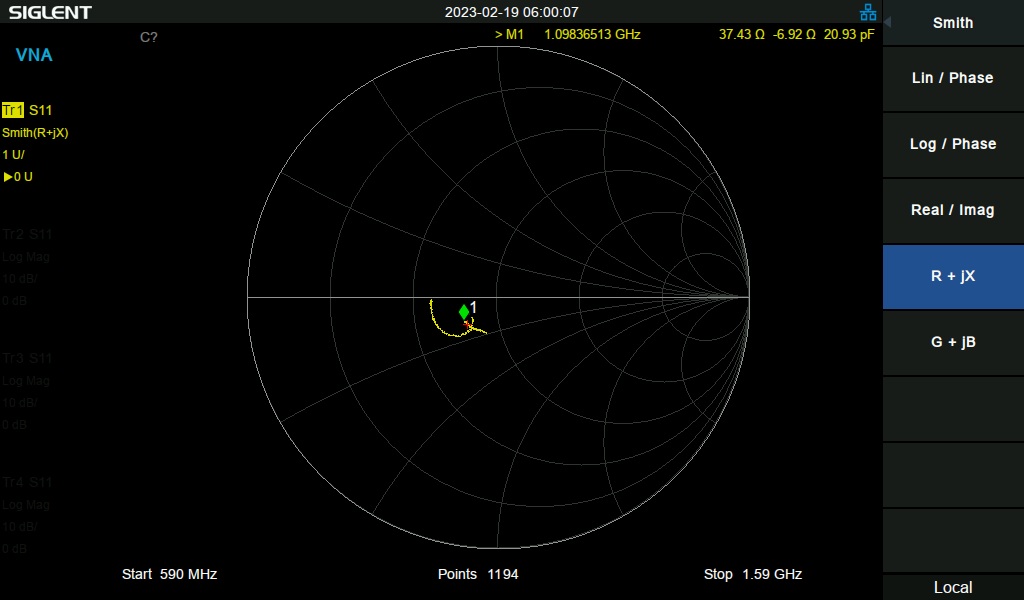
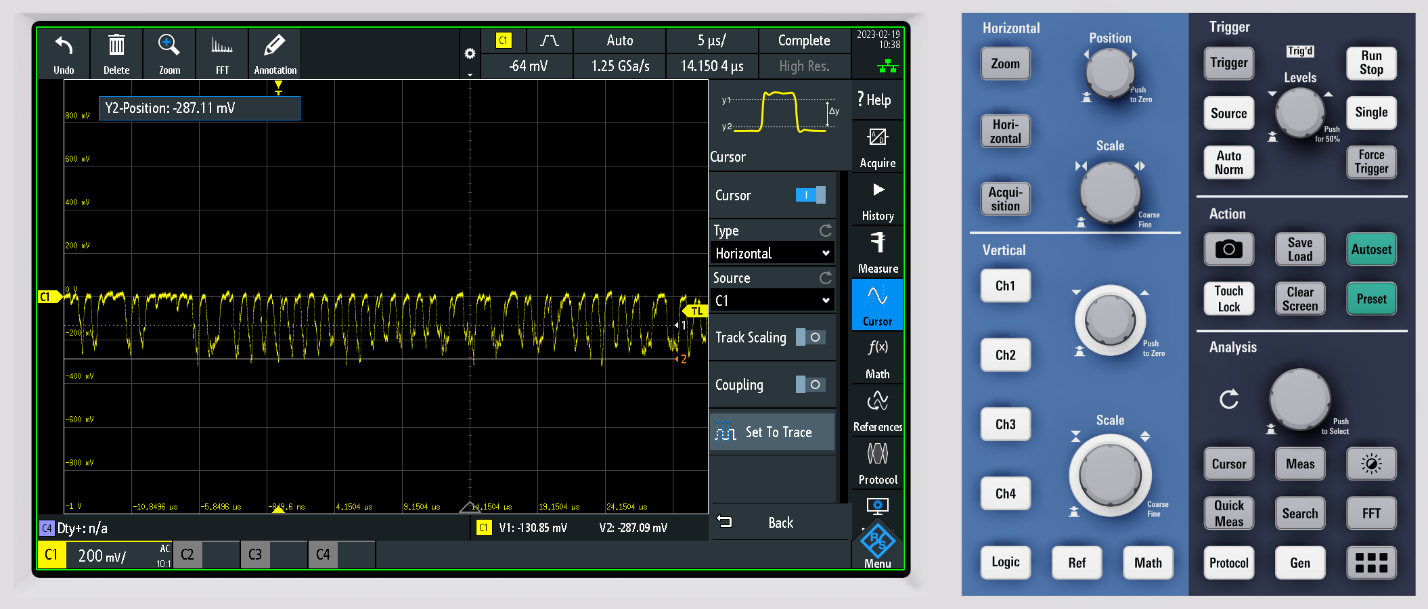
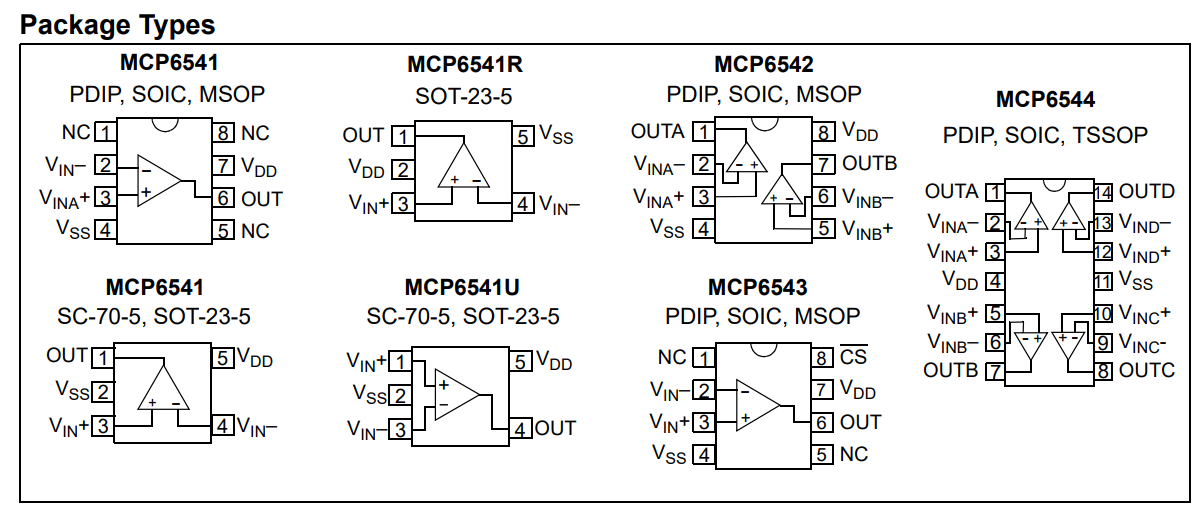
Design Notebook

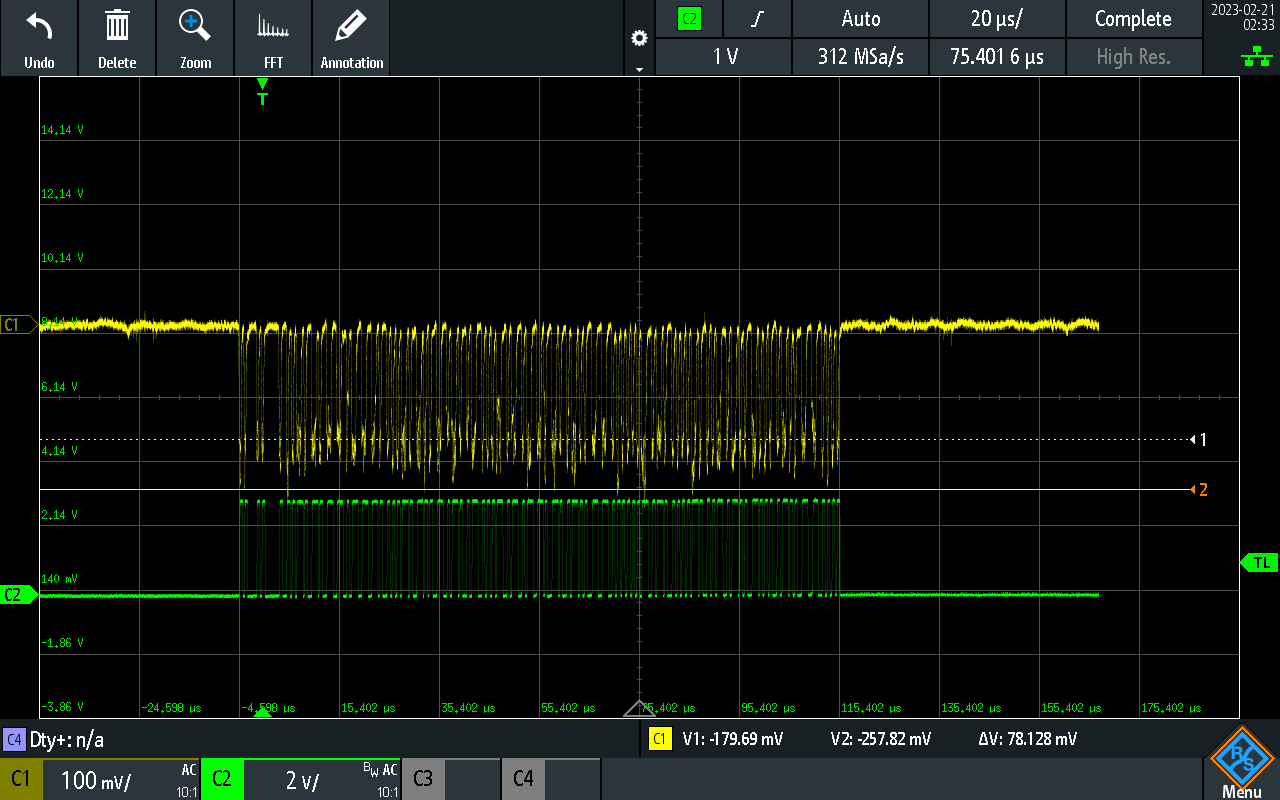
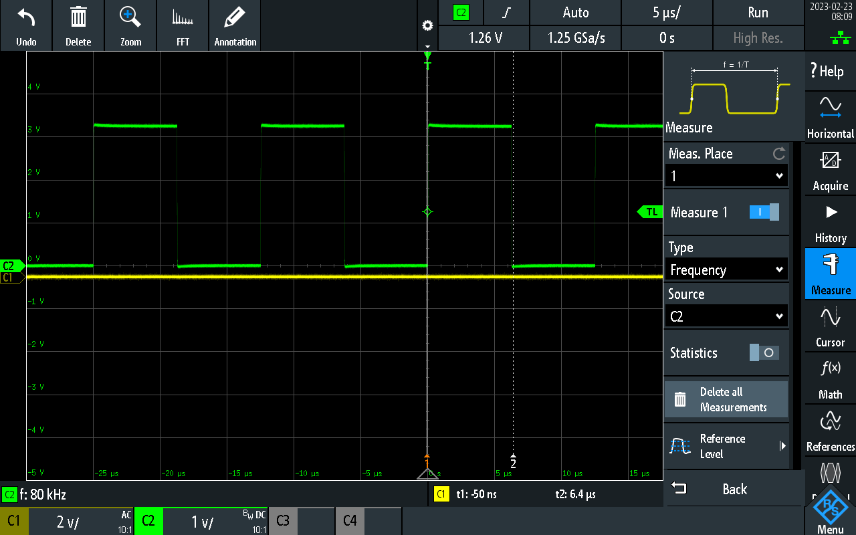
# 2023-02-18 RF Bringup

* Proto RF board bringup
  + SF2312D SAW Filter Datasheet
    - 
  + RF board S21 frequency response with 30dB attenuator on output
    - 
  + RF board S11 input match (powered off)
    - 
  + RF board S11 input match (powered on)
    - 
* Proto detector board bringup
  + Input match without power
    - 
  + Input match with power
    - 
* Decoding ADS-B Messages
  + 
* Notes for next PCB Rev
  + ADS8314 has internal coupling cap, can ditch 39pF cap on input to RF detector board.

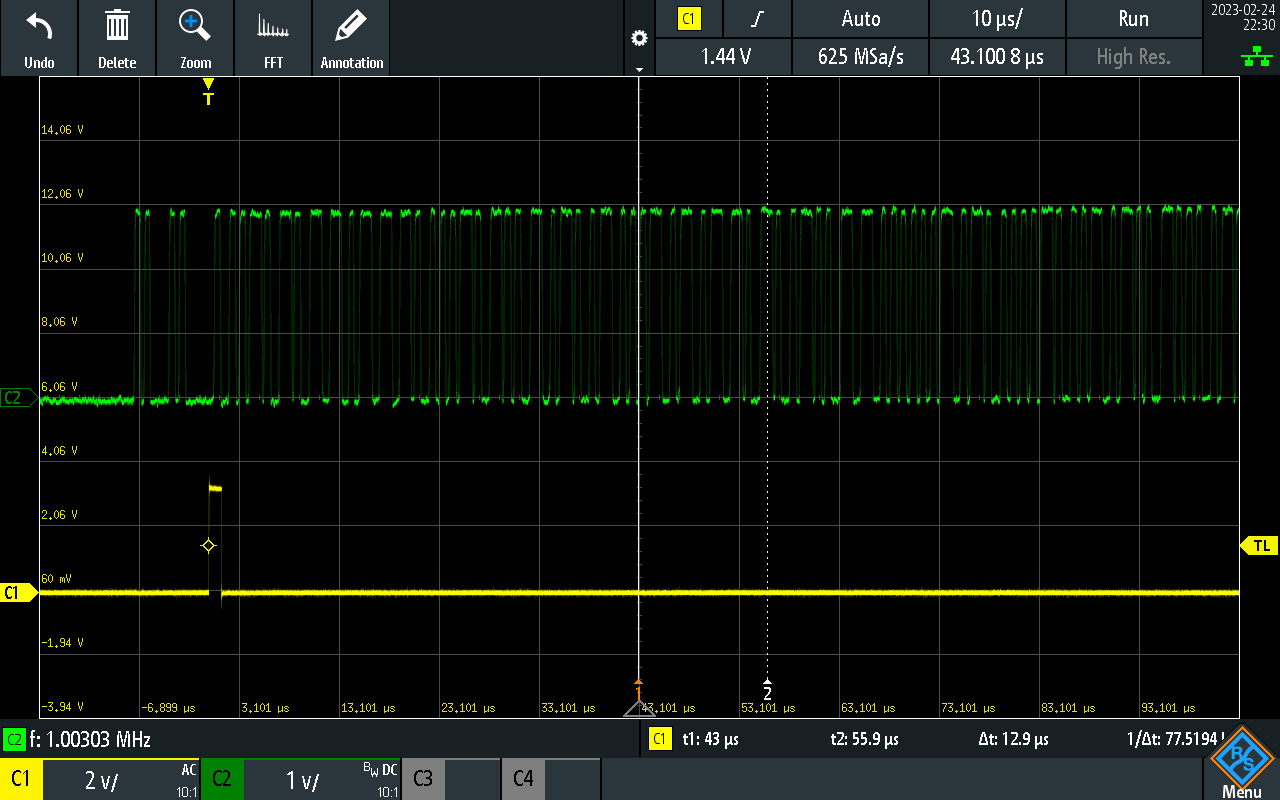
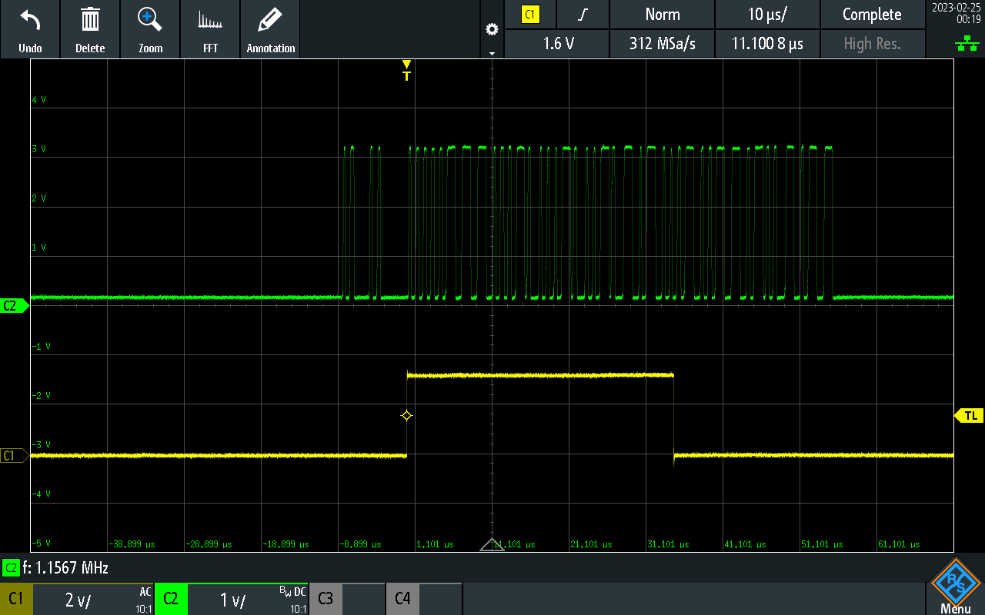
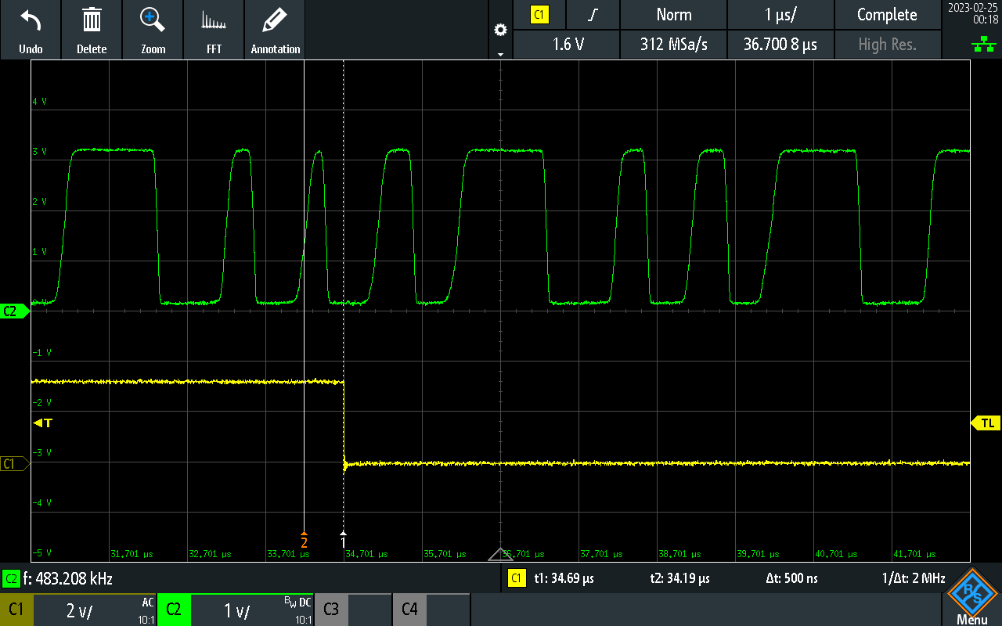
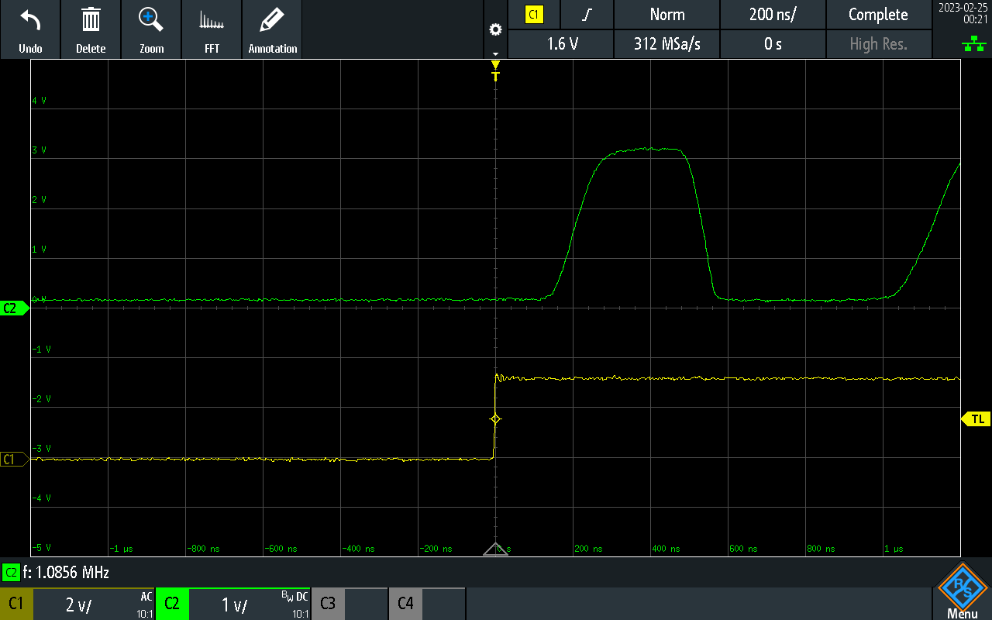
## 2023-02-19 MCU Board Bringup

* Notes for next time
  + Remember to look for unconnected in the PCBNew DRC! Boy.
    - 2N7002 MOSFET gate was not connected.
    - Comparator output was not connected to Pi Pico.
  + MCP5941U is a DIFFERENT footprint than the MCP6541! I got the MCP6541U, which is pin compatible with the LMV331 but is different than the MCP6541.
    - 
  + MCP6541 is too slow to decode ADS-B at 1MHz. The propagation time is 4us! This results in only a partial slew out of the comparator when the input signal has made the full pulse. Not good.
  + Switched comparator to LMV331 (~400ns propagation time, slowish but maybe fast enough). Need to add an output pullup resistor since the output of the LMV331 is open collector.

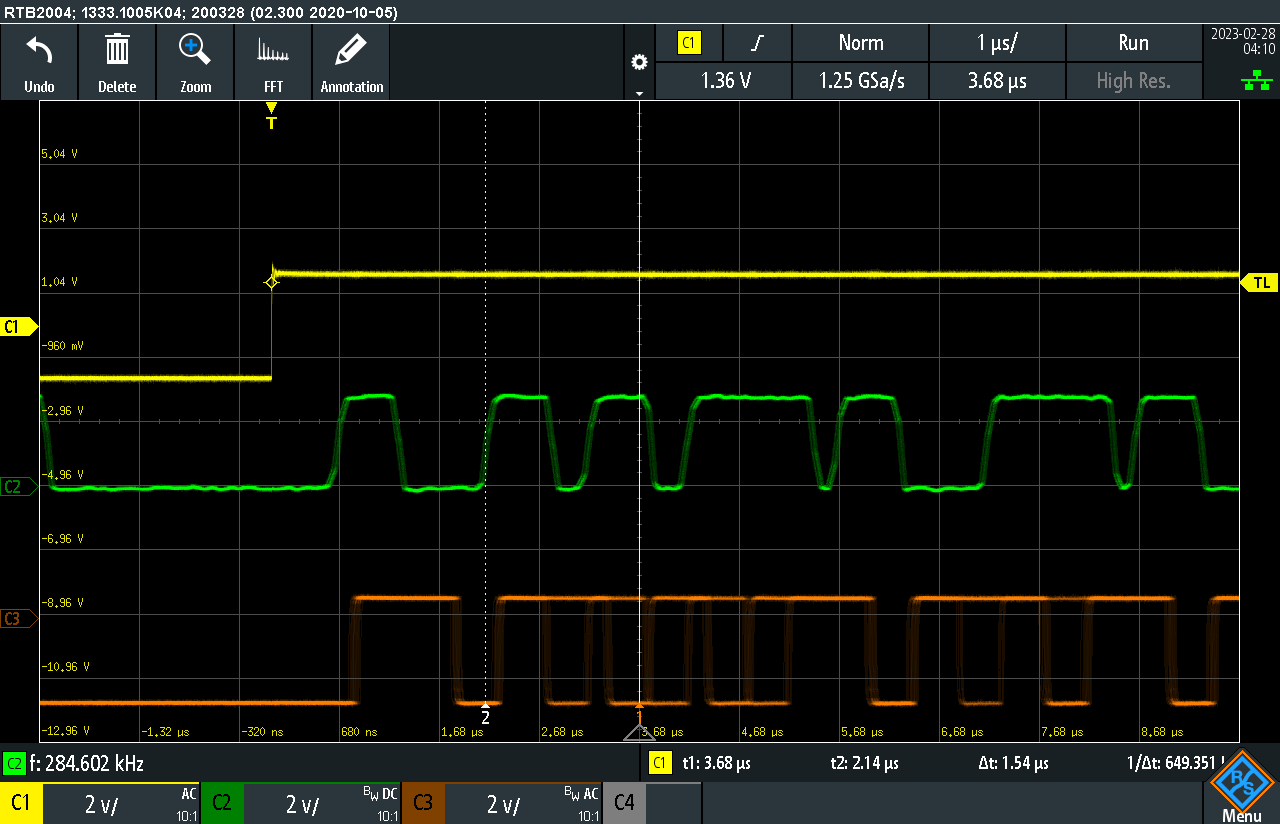
## 2023-02-20 Decoding ADS-B?

* Got a nice ADS-B packet!
  + 
  + Downsampled to 1:20 sample ratio to use with arbitrary function generator. 312Msps/20 = 15.6Msps
  + Replicating ADS-B waveform from comparator output with arbitrary function generator.
* CANRP2040 project has [really good examples and notes](https://github.com/KevinOConnor/can2040/blob/master/docs/Code_Overview.md) about using RP2040 PIO for decoding stuff at 1MHz.
* [RP2040 Datasheet](https://datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf)
* [C++ PIO Tutorial](https://www.digikey.com/en/maker/projects/raspberry-pi-pico-and-rp2040-cc-part-3-how-to-use-pio/123ff7700bc547c79a504858c1bd8110)
* Bringing up PIO capture program.
  + Testing with 200 cycle “blink” pulses, PIO clock set to 16MHz.
    - 
    - 200\*80kHz = 16MHz. Not bad!
* Got into some really weird error loops with the debugger when trying to debug PIO. Some troubleshooting tips that may or may not make a difference:
  + Move the SWD cable away from the RF boards / put the cable below the ground planes.
  + Power cycle by turning off the power supply for a full 10 seconds, then reset the board with the debugger and reconnect.
  + Wait until the debugger is done thinking before clicking any buttons.
  + Disable adaptive clocking, set it to something low like 4000kHz (this is not done by default)!
  + This is it!! Launch vs. [attach](https://www.electrorules.com/vscode-cortex-debug-launch-configurations/)! Since I am running an external debugger session, I need to attach to it instead of launching a new one. Now all the issues make sense (things working the first time but not the second, all of the breakpoints being set over each other, etc).

## 2023-02-24 More Decoding ADS-B

* Successful preamble detection on a recorded message!
  + 
  + Used PIO to detect a preamble and set a GPIO HI right after a valid one was found.
* Looking for end of ADS-B message.
  + Problems with looking for 3 sequential LO samples at 0.5bit intervals.
    - 
      * False end of packet.
    - 
      * Misalignment with sample interval causes a LO to be read on a rising edge, providing three sequential LO samples.
    - 
      * Packet read interval starts before first bit. This shouldn’t happen, maybe it’s not waiting for packet first edge to start read interval.

## 2023-02-27 Decoder Work

* Can’t put PULSES net next to the RUN pin (reset, active low). Data pulses trigger the microcontroller reset and cause the device to restart intermittently.
* Added a sideset debug output to the decode PIO program.
  + 
    - Yellow line is decode interval, green line is data, orange line is PIO decode timings. Very jittery! Something is wrong with the state machine and this is causing decodes of the test message to be inconsistent.
* Trying to get delay timing correct.
  + 
    - Want to sample polarity around 6ns for a rising edge to see if next bit will be a 1. This corresponds to around 7 clock cycles at 12MHz.