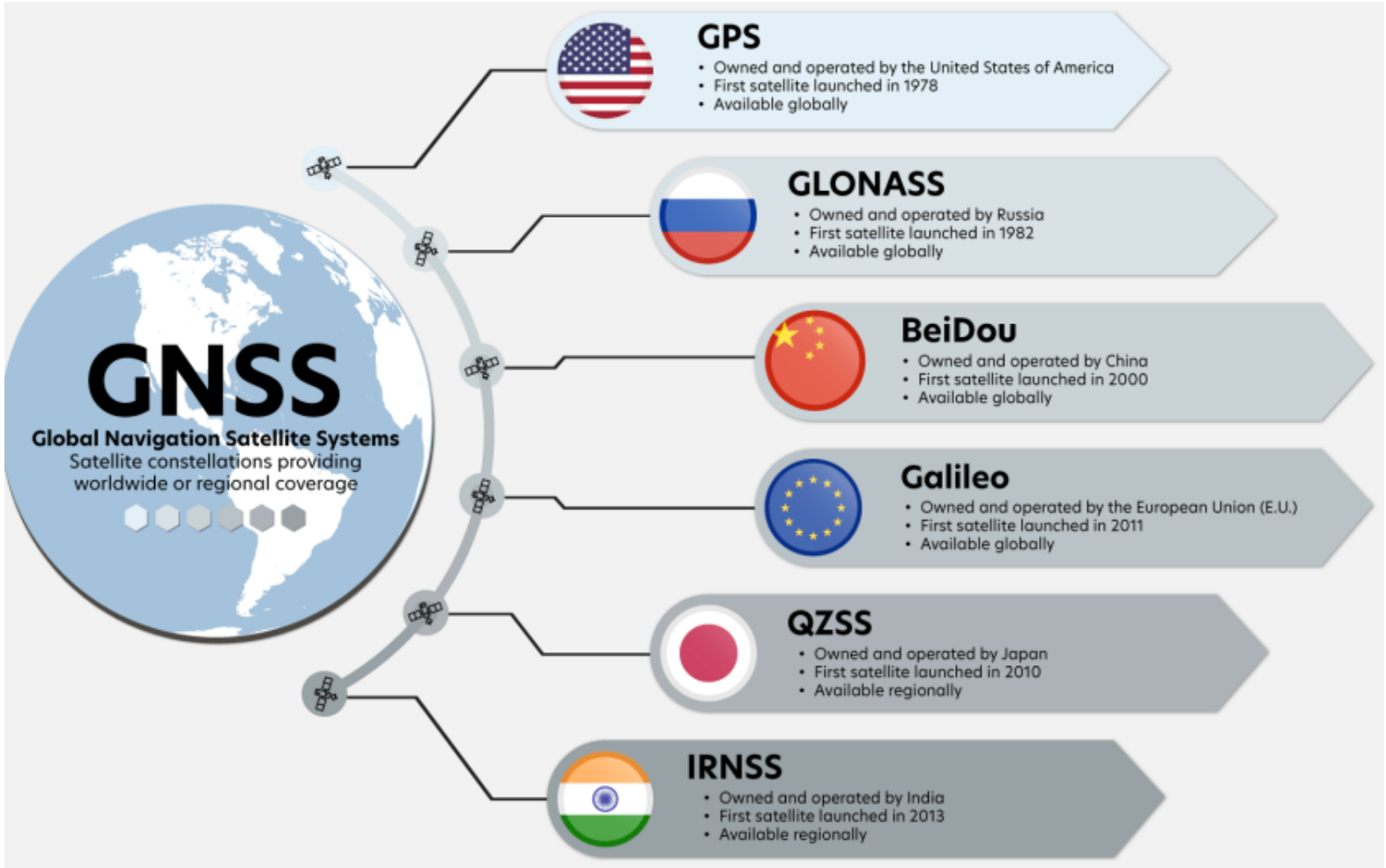
gr-pocketsdr: Bringing PocketSDR Front-Ends to GNU Radio and GNSS-SDR Ecosystem

Minhaj Uddin Ahmad
Ph.D. Student | mahmad12@crimson.ua.edu
Connected and Automated Mobility Lab @ UA

GNU Radio Conference 2026

THE UNIVERSITY OF ALABAMA® | Alabama Transportation Institute

Global Navigation Satellite System



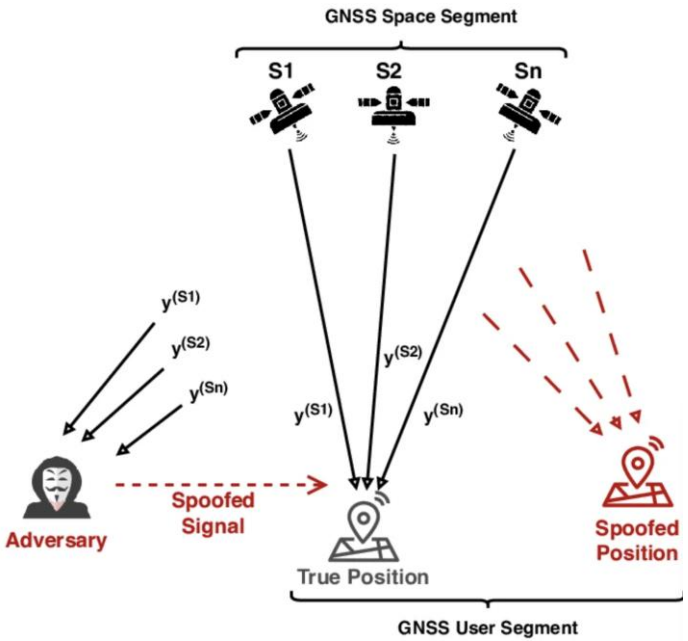
https://eos-gnss.com/wp-content/uploads/2023/04/GNSS-Infographic-1-1024x576.png

GNSS satellite signals are open for anyone / any device to receiver, for free, anywhere on earth.

With an SDR receiver, you can receiver the signals and process it in your laptop, turning your laptop into a GNSS receiver. This allows researchers to develop and test new algorithms and signal processing techniques.

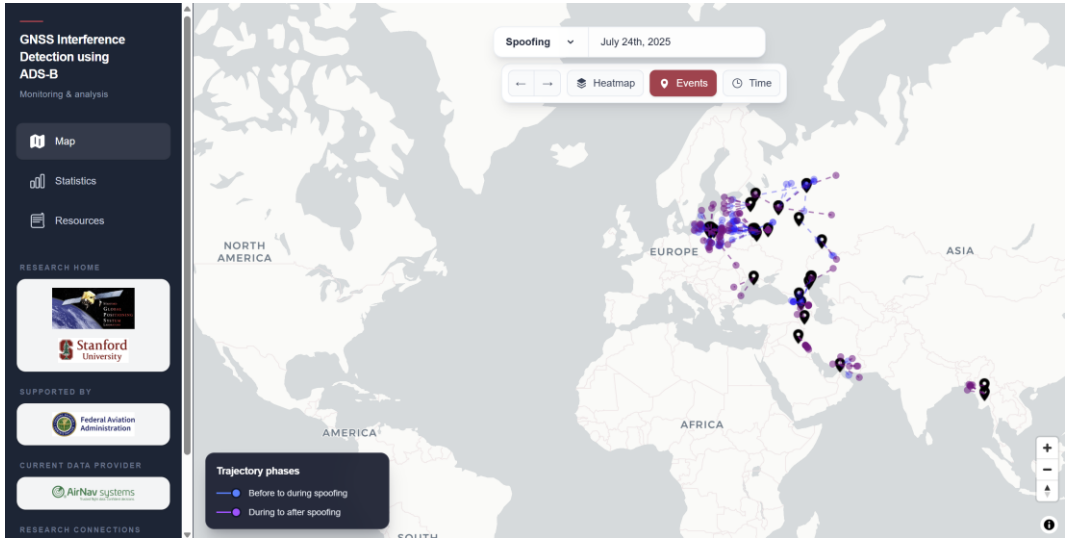
Since GNSS signals are open, without any protection mechanism (mostly), there are safety and security concerns too, requiring active research in this area.

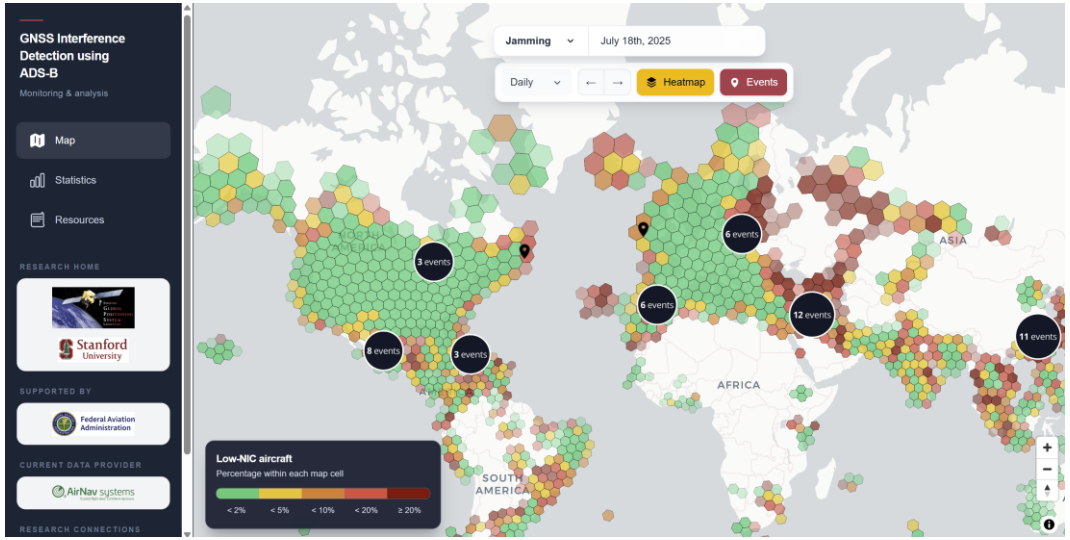
We need more research for GNSS resiliency



Spoofing concept

https://www.researchgate.net/publication/382288564_CLPDI-based_Fast_Secure_Code_Estimation_and_Replay_Attacks_on_GNSS_Signals





Broad hardware support makes research accessible

Commodity SDR Hardware

PocketSDR

RTL-SDR

HackRF

USRP

LimeSDR

bladeRF

PlutoSDR

Interface

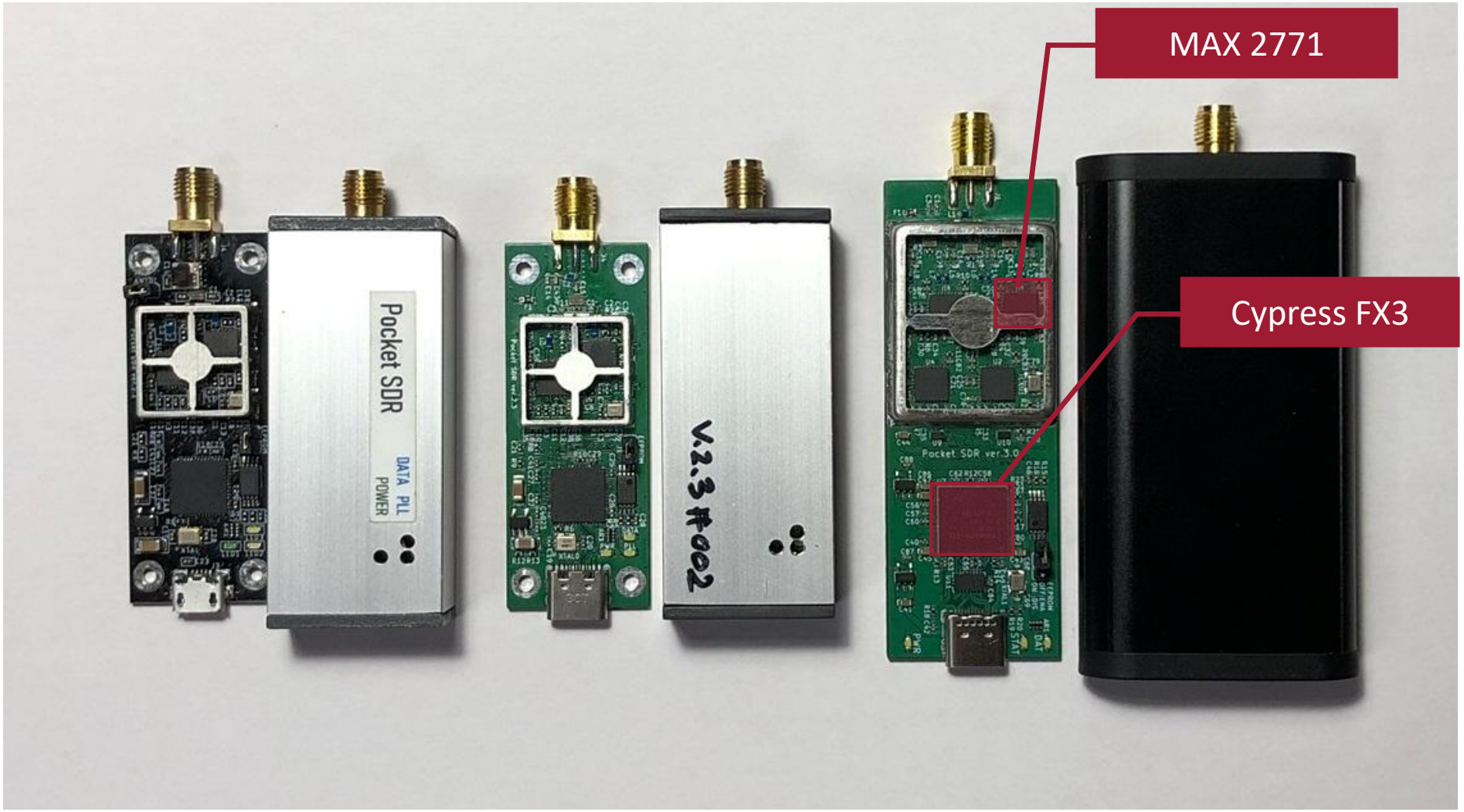
Processing Software

GNU Radio

GNSS-SDR

RTKLIB & Custom

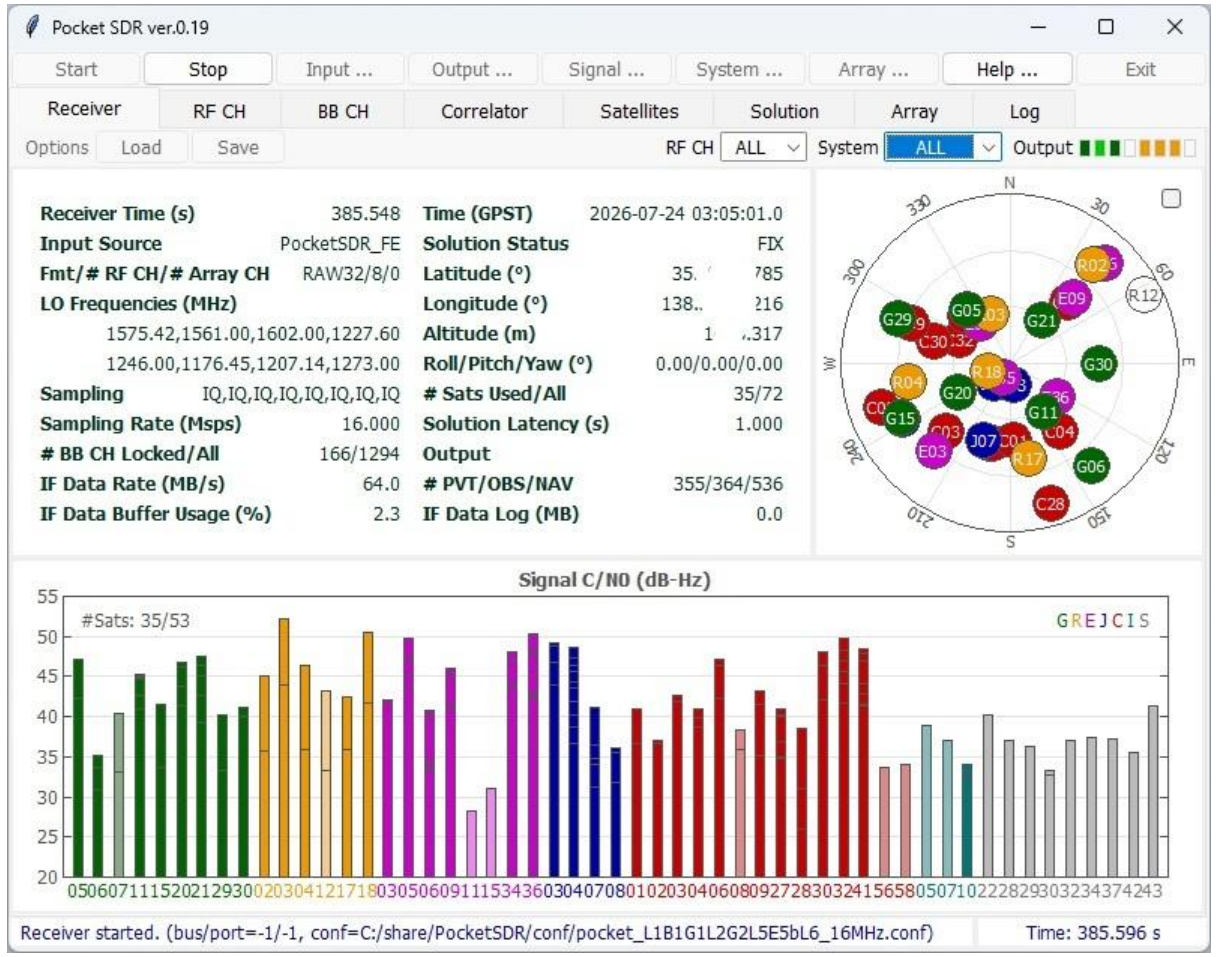
PocketSDR



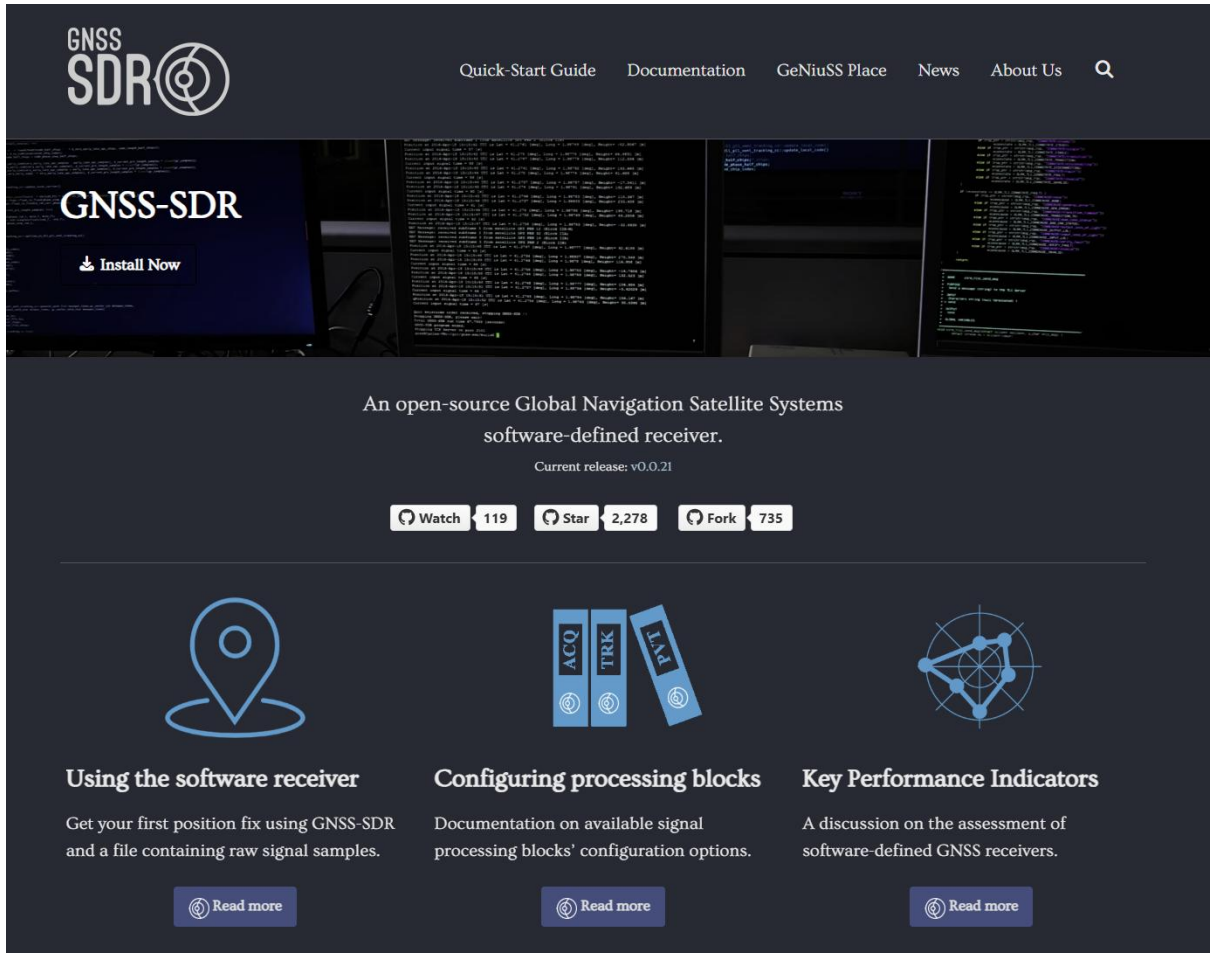
https://github.com/tomojitakasu/PocketSDR

https://www.datagnss.com/products/pocketsdr-gnss-receiver

General Purpose GNSS Processing

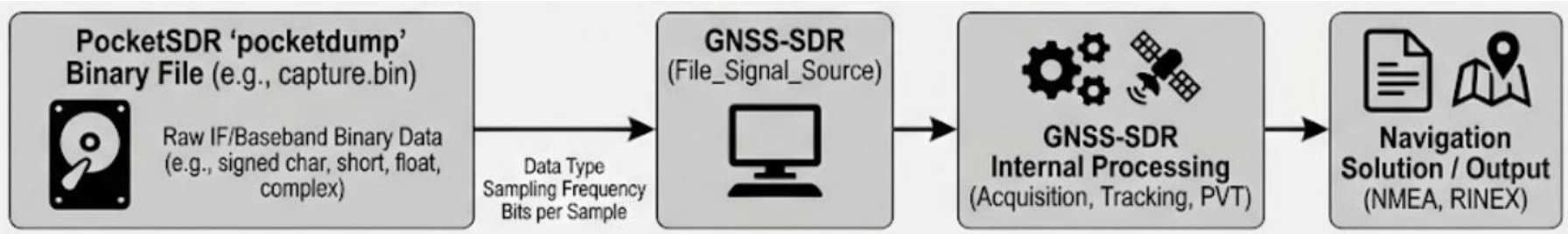


https://github.com/tomojitakasu/PocketSDR

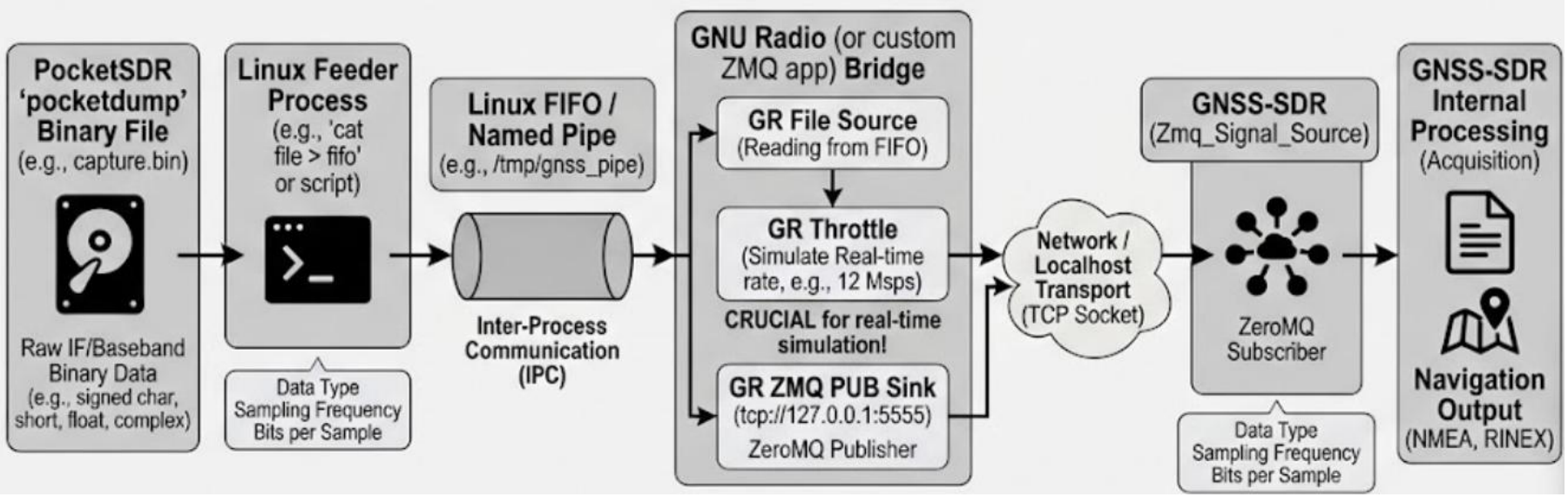


General Purpose - https://gnss-sdr.org/

Traditional workflow

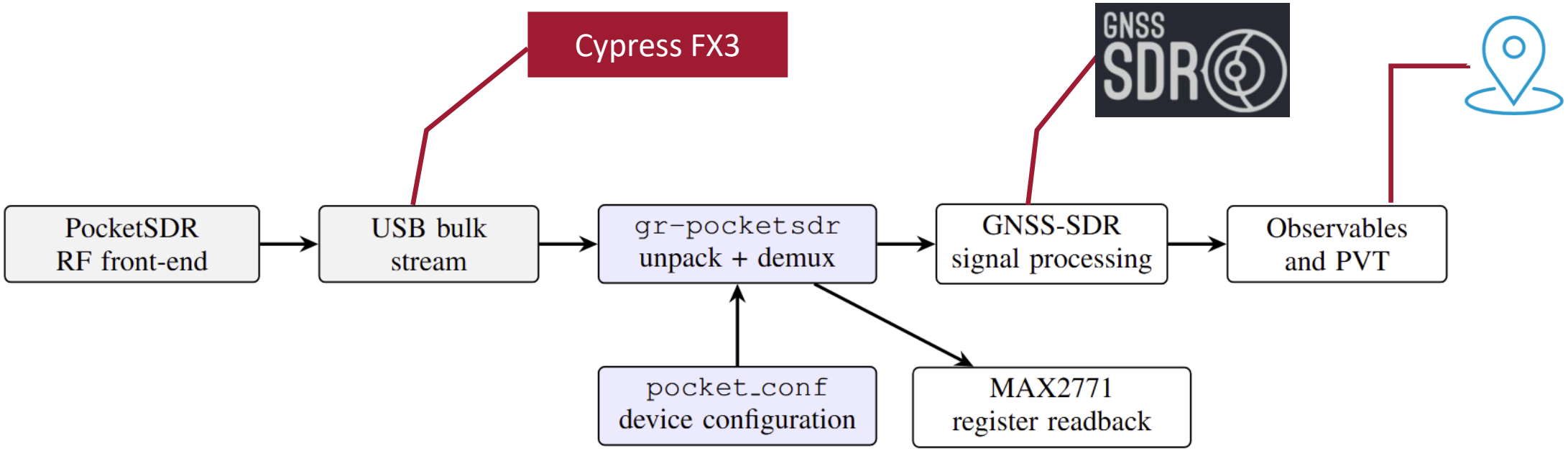


Offline file post-processing



Workaround for real-time processing

Building the bridge for live reception



Lets compare the gr-pocketsdr vs. ZMQ workaround

	Pocket_SDR_Signal_Source	ZMQ_Signal_Source
Total CPU	**178.9 %**	**205.1 %**
Latency, median	320.6 ms	334.8 ms
Latency, worst case	374.9 ms	679.7 ms
Lag spread (jitter)	58 ms	236 ms
Wire rate, 1 channel	16 MB/s	64 MB/s
Reports lost samples	yes	no

Medians differ by 14 ms — worst cases differ by 305 ms

Application – Lets test the PocketSDR clock

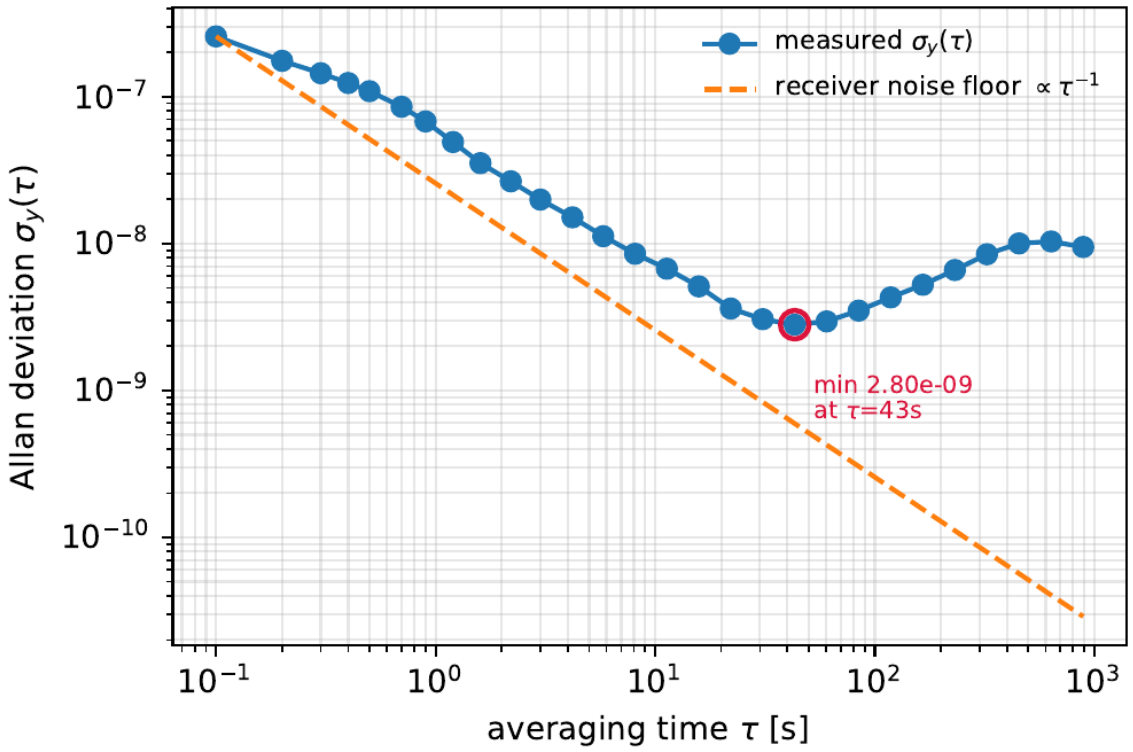
Clock offset = TCXO phase error vs GPS time

37.6 min unbroken fix, 22 584 epochs at 10 Hz

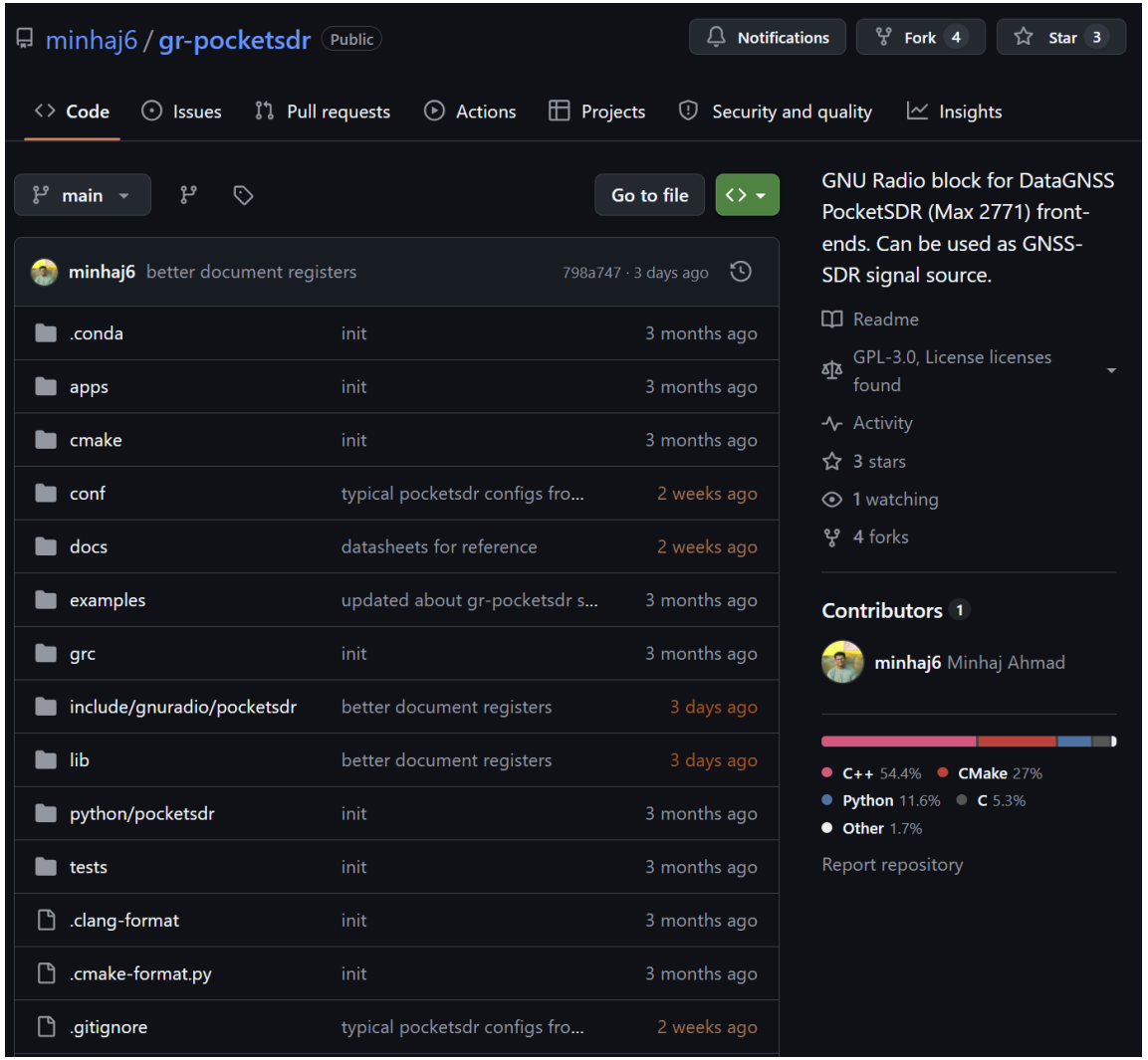
Frequency offset: −0.265 ppm

Stability floor 2.8×10^{-9} at τ = 43 s

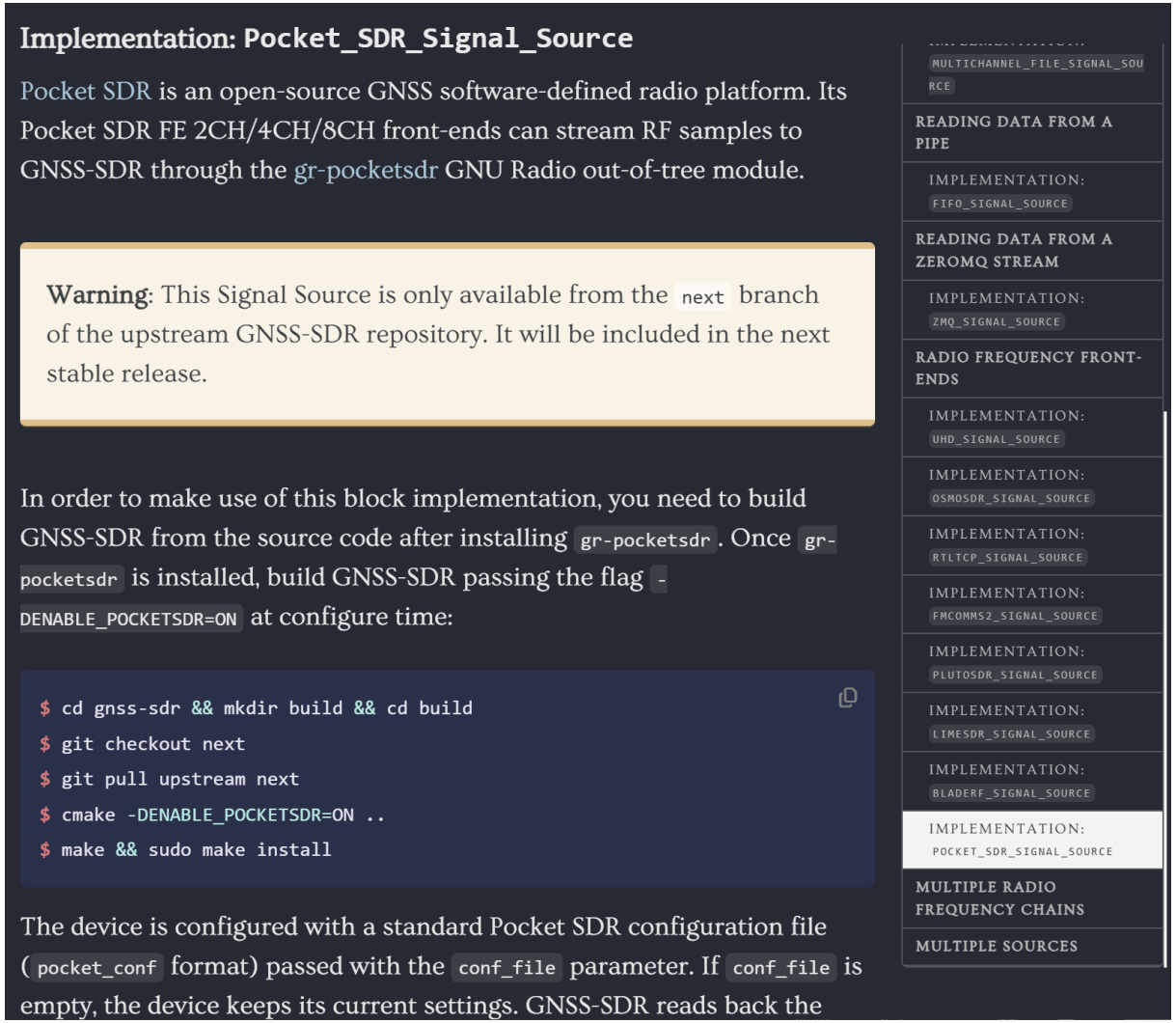
Above the receiver noise floor throughout



Open-source!



Source code available at GitHub



Merged with GNSS-SDR upstream, development branch

Conclusions and future work

It works! (few quirks to work out)

14.6 % less CPU than ZMQ workaround, 305 ms better at worst case jitter

4× less data transmission compared to ZMQ workaround

Get reports of ring-buffer overruns

Multi-frequency (GPS L1, L2, L5, Galileo) testing is open for future work

Thank You

Questions?

Supported by the National Center for Transportation Cybersecurity and Resiliency (TraCR), a U.S. DOT National University Transportation Center, and by NSF Award #2340456.

Contact: Minhaj Uddin Ahmad | mahmad12@crimson.ua.edu