End of Summer Presentation and Demo
Unicorn Blue

Wi-Fi Detection

August 15, 2022

Presented by Kent & Daniel

Geon Technologies, LLC

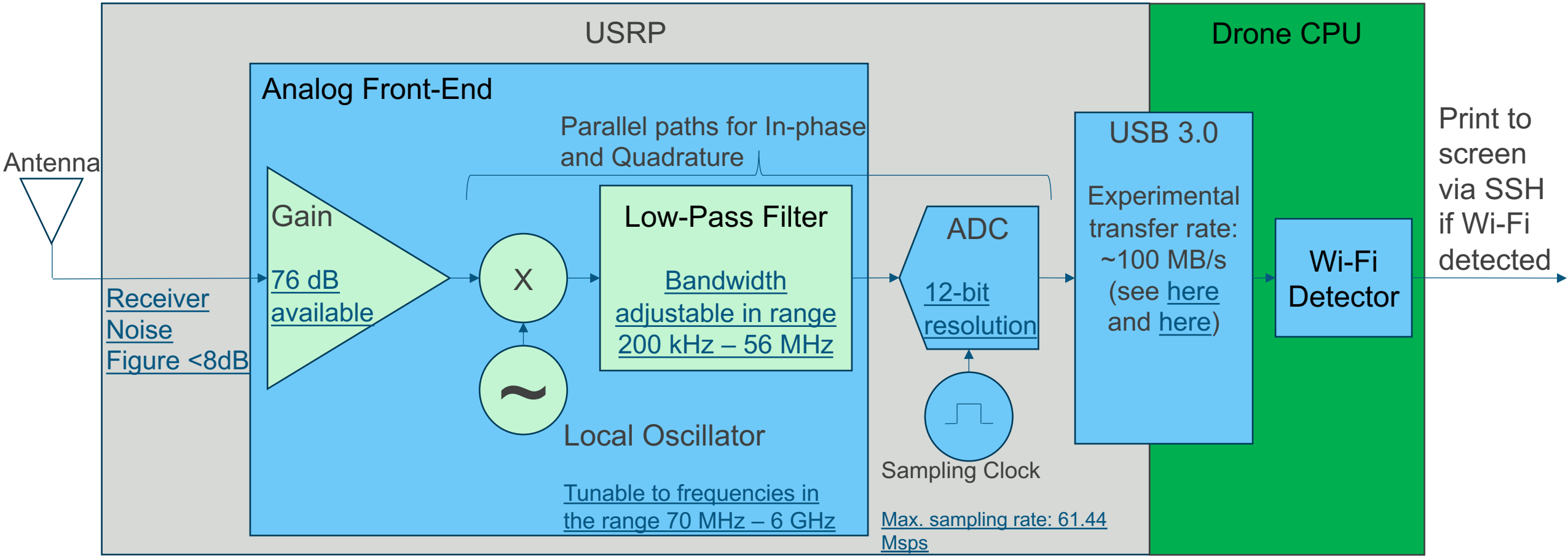
Outline

- Project overview
- System overview
- Detection algorithm
- System integration
- Performance
- Next Steps
- Experience Gained

Project Overview

- From Davis's Onboarding Presentation:
  - "Description: Develop an 802.11ac detector to be deployed on a Qualcomm RB5 drone."
  - "Benefits: Expand Geon's proprietary ES capabilities and readiness to secure funding for OpenCPI and other ES-related efforts."
- Bands at 2.4 GHz and 5 GHz.
- Goals
  - Baseline goal – achieved
    - Have program running on drone.
    - Turn on router to transmit OFDM Wi-Fi at known frequency.
    - See message printed to screen confirming Wi-Fi detected.
  - Stretch goal: SSID extraction – insufficient time
  - Stretch goal: multi-channel scanning – achieved with limitations
- Roles
  - Kent – System integrator
  - Daniel – Algorithm developer
  - Davis – Mentor

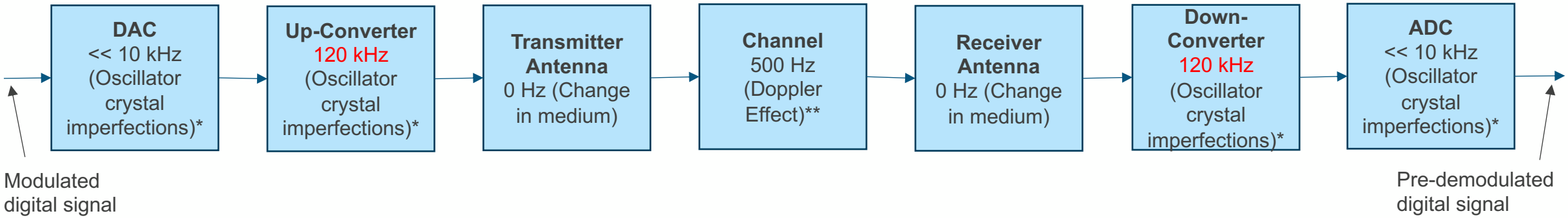
System Overview



Host and over-the-wire data types - complex-int16

Expected Carrier Frequency Offset (CFO)

Full Communication System with approximated maximum CFO contributions:



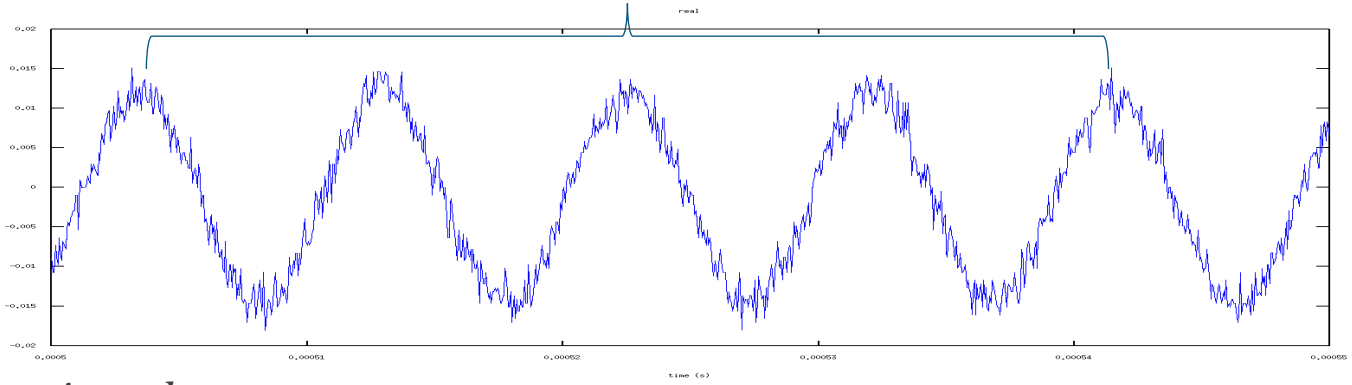
* Based on 20 ppm tolerance of a standard oscillator crystal, 6 GHz carrier frequency (upper end of the 5 GHz band), and 20 Msps DAC and ADC clock.
** Based on of 27 m/s speed of drone relative to transmitter. Equation retrieved from Young and Freedman, *University Physics with Modern Physics, 15th edition*, page 528.

- Approximated maximum CFO: 240 kHz
- Wi-Fi detection algorithm can handle CFO up to ~500 kHz.

Follow Up on PDR Streaming Test Concerns

- Supplied 5200.1 MHz sinusoid to USRP with signal generator
- Ran rx_samples_c.c (under copywrite)
- Plotted output
  - Residual well within maximum expected CFO of 240 kHz
  - Low SNR due to attenuation
- Tutorial

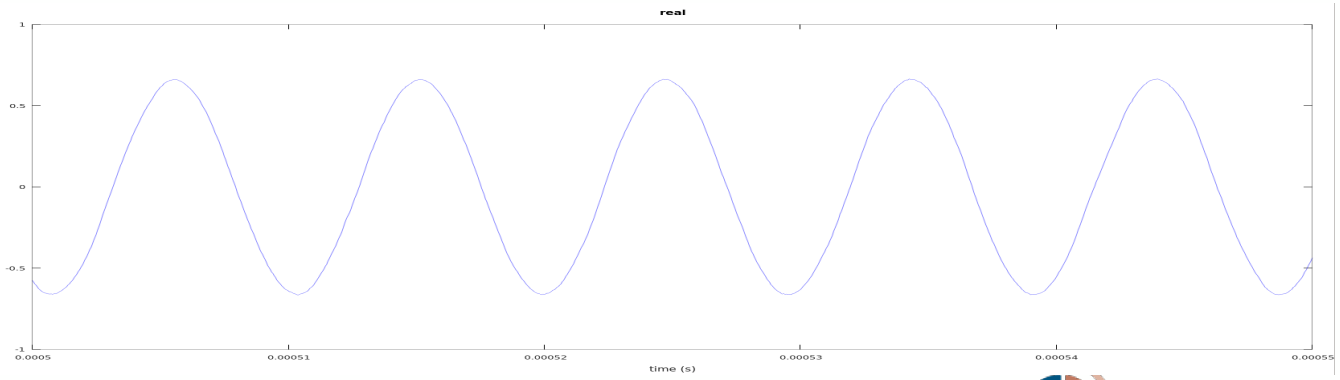
With attenuation: 4 cycles, 38.2 µs



$$\frac{4\ cycles}{38.2\ \mu s} = 0.105\ MHz \Rightarrow 5\ kHz\ residual$$

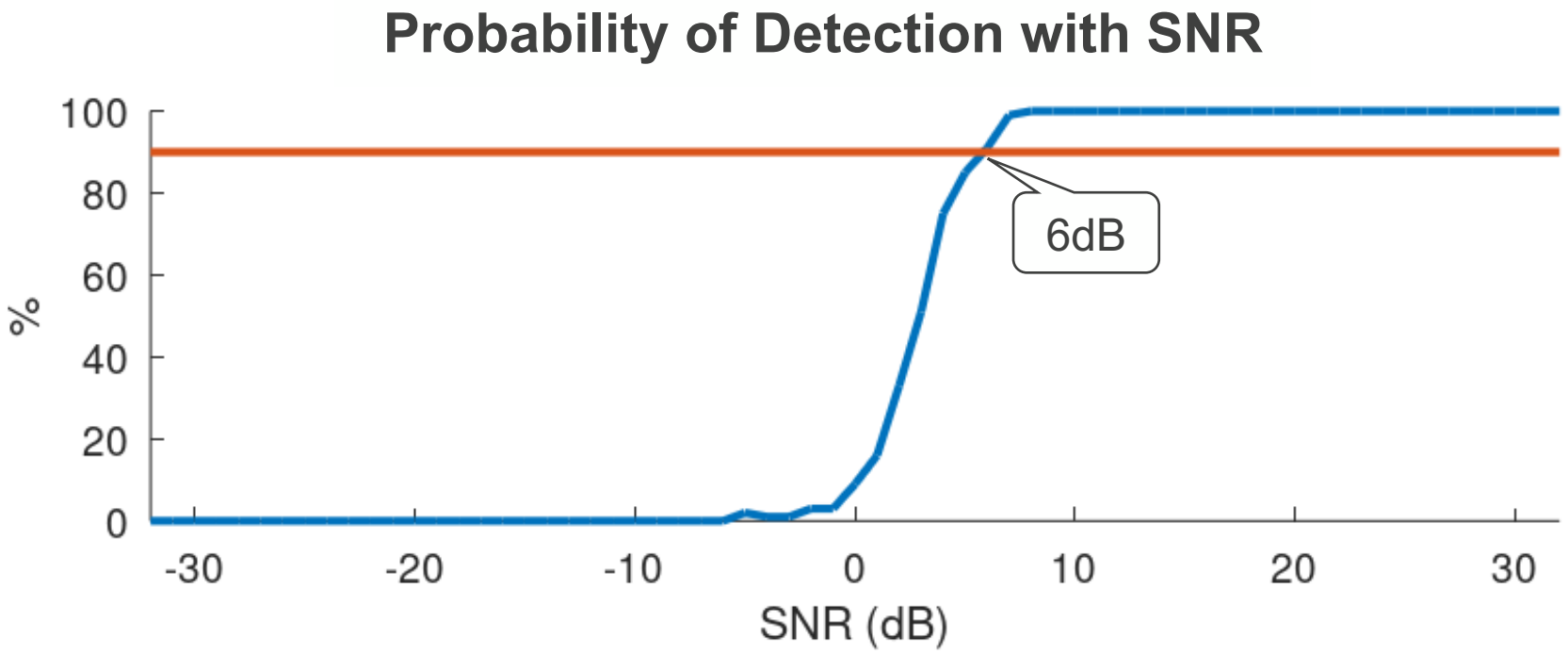
Approximate amplitude: 0.013

Without attenuation:



Approximate amplitude: 0.66

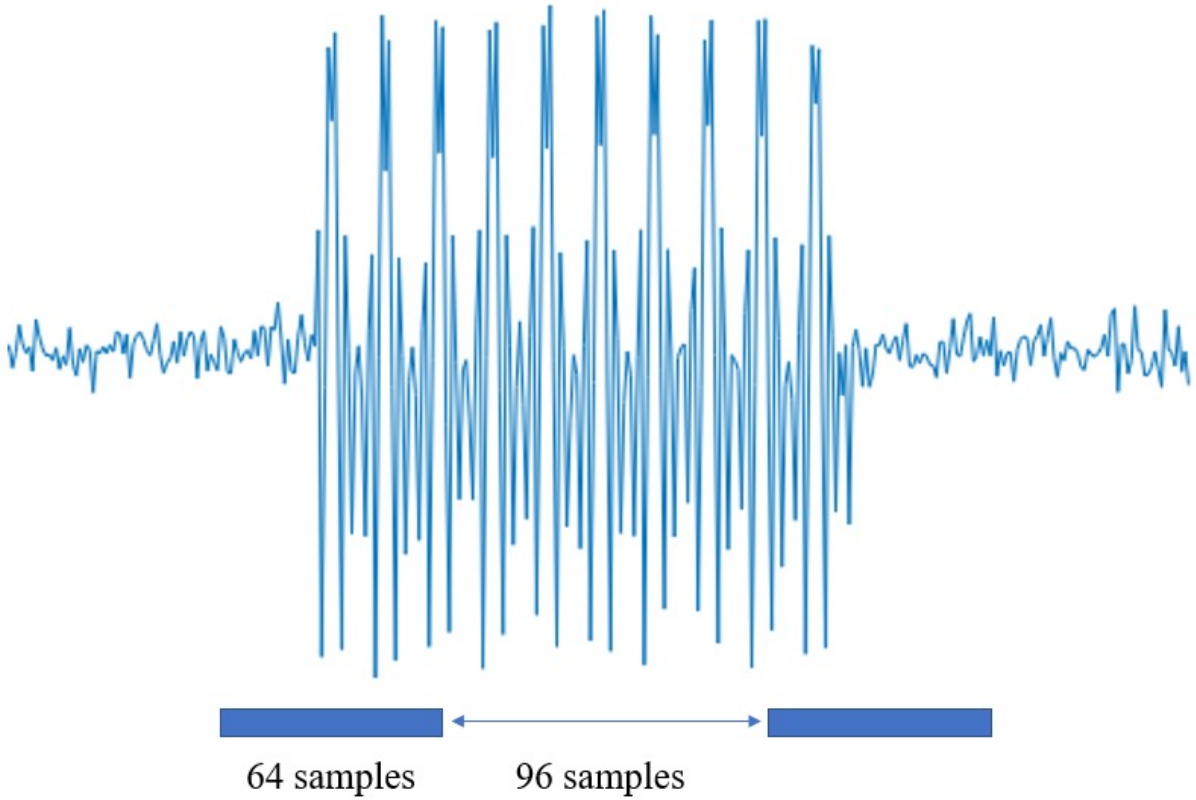
Signal to Noise Ratio Tolerance



Detector Algorithm

- Subsampling
- CFO Correction
- Cross Correlation

Subsampling

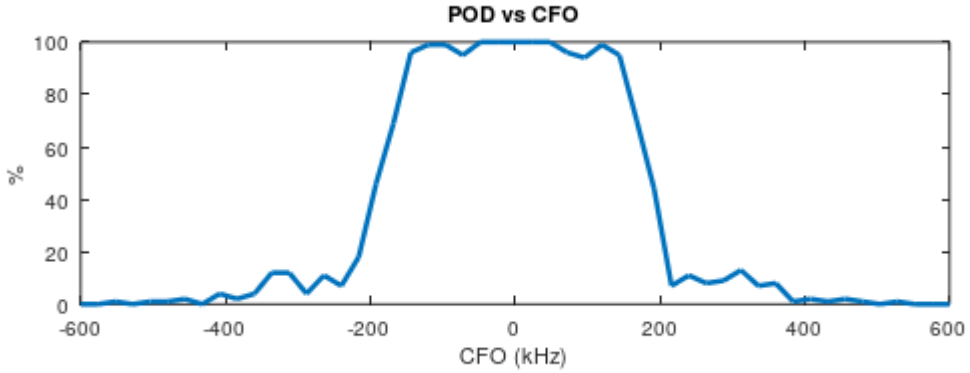


CFO Correction

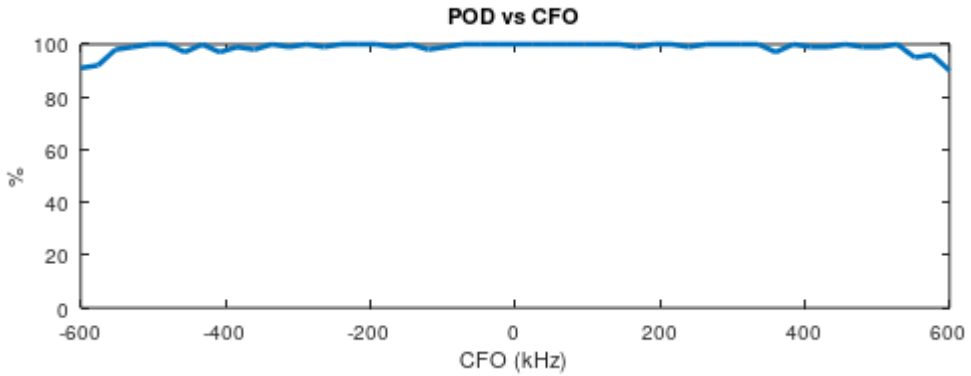
- Calculate angle difference between signal and lagged signal
- Perform moving average

CFO Correction

Probability of Detection without CFO Correction



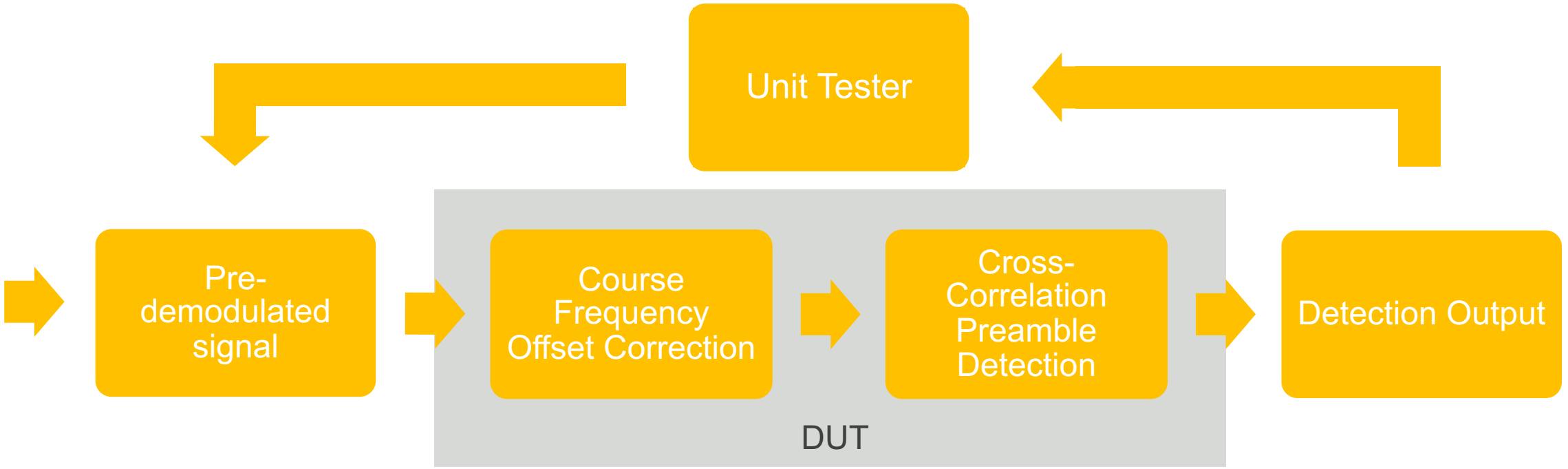
Probability of Detection with CFO Correction



Cross Correlation

- 16-sample chips
- Add chips together
- Take absolute value of output

Detector Flowchart



System Integration on the Drone

- Streaming data on the drone
  - Using docker container generated with buildx
  - Confirmed data streaming with 0.1 MHz sinusoid test
  - Tutorial
- Integrated Wi-Fi detection algorithm with data streaming
  - Use C Coder to Convert detection MATLAB scripts to C
  - Adapt rx_samples_c.c to call the detection algorithm and support common usage
  - Compile in Docker container on ARM with CMake
- Designed single-channel detector and multi-channel scanner

Data Overflow

- Encountered when running rx_samples_c.c and variations thereof
- Initially encountered on ARM on drone
- Solutions
  - Switched to complex-int16 CPU data type
  - Optimized size of receive buffer for transporting data from USRP
  - Optimized detection algorithm
  - Introduced multithreading
  - Switched to one time memory allocation
  - Oriented fan correctly
- Results
  - Data overflow very rare (Possibly once in several hour of active testing)
  - Detector algorithm keeps up with 20 Msps data streaming
  - Detection of Wi-Fi reported within 4 s of receiving preamble in single-channel detector

Single Channel Detector Performance

- False positivity rate: 0.0% in 1000 packet test
- Detection rate test
  - USRP on lab desk nearest kitchen
  - 2.4 GHz band
  - 1000 packets, 1 preamble/packet

Test	Router Location	Lab Doors	Detection Rate
1	Conference Room 1	Closed	81.5%
2	Conference Room 1	Open	98.0%
3	Same wall as door to Conference Room 2	Open	99.9%

Adjacent Channel Wi-Fi Detection

- Detector often identifies Wi-Fi preambles on adjacent channel as preambles on operating channel
- 2.4 GHz band more susceptible than 5 GHz band
- Attempted solution: Reduced low-pass filter bandwidth from 56 MHz to 16 MHz
- Results
  - Test for decreased detection probability on operating channel
    - Repeat Test 1
    - 56 MHz bandwidth: 81.5% detection rate
    - 16 MHz bandwidth: 81.6% detection rate
  - Test for increased immunity to adjacent channel detection in 2.4 GHz band
    - 1000 packets, router ~2 m from antenna, transmitting 1 preamble/packet on adjacent channel
    - 56 MHz bandwidth: 70.4% adjacent channel detection rate
    - 16 MHz bandwidth: 64.4% adjacent channel detection rate
  - Adjacent channel detection nearly eliminated in the 5 GHz band
- Improvement; further investigation necessary

Scanning Detector Performance

- Analyzes 1 cs of data from each channel
- Limitation: Will not detect every packet due to short dwell time
- Scans entire 5 GHz band in 7.3 s
- Scans North American 5 GHz channels (56 channels) in 2.5 s
- Demo

Next Steps

- Mitigate adjacent channel Wi-Fi detection
- Investigate more rigorous channel scanning mechanism – likely FPGA channelizer
- Implement SSID extraction
- Achieve DSSS Wi-Fi detection
- Consider how to handle bonded channels

Experienced Gained

- Linux, Git, Docker, UHD, Cmake
- Communication Systems
- Wi-Fi
- Enhanced programming skills: C, MATLAB
- Technical Documentation
- Troubleshooting and Problem Solving

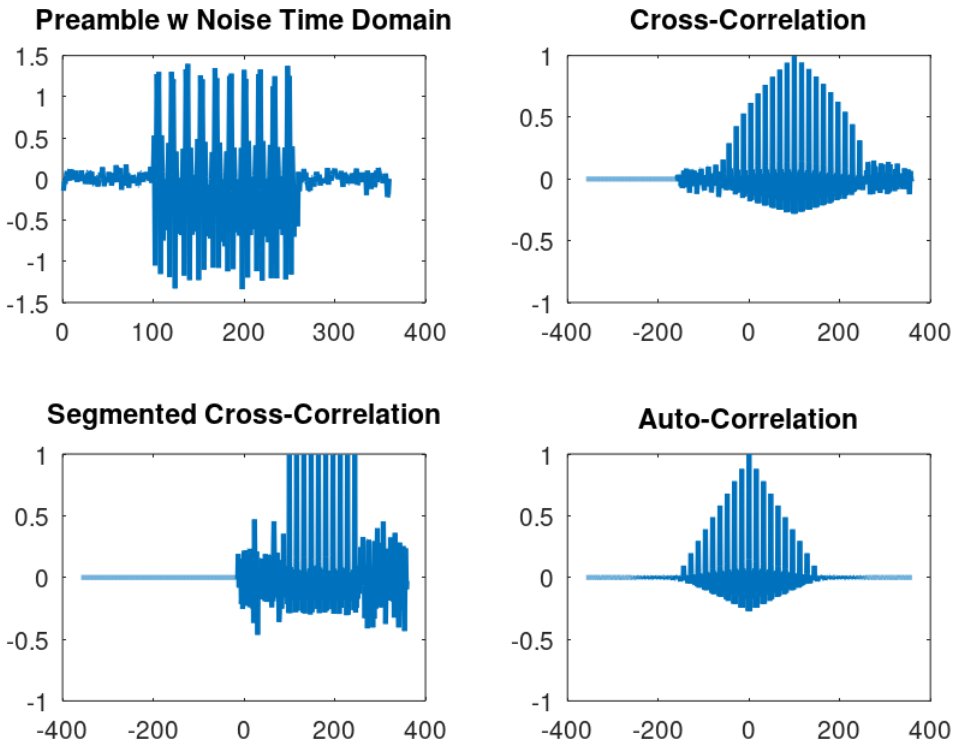
Summary

- Project: 802.11ac Wi-Fi detector for drone
- System
- Detector algorithm
- System integration
  - Data streaming on drone
  - Wi-Fi detector integration
  - Data overflow mitigation
- Performance
  - Single Channel Detector Statistics
  - Adjacent Channel Detection
  - Multi-channel scanner
- Next steps
- Experience gained
- Thank you Davis and Geon!

APPENDIX

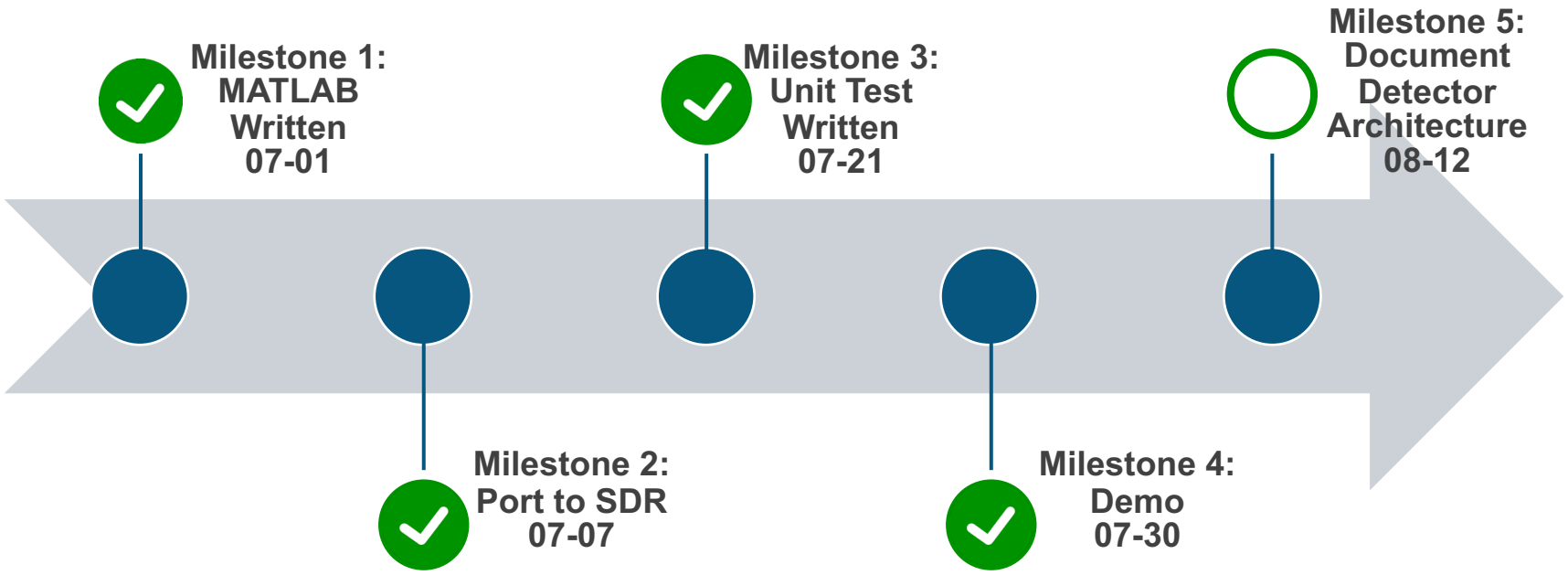
Algorithms Considered

- Cross-Correlation
- Segmented Cross-Correlation
- Auto-Correlation



Geon Technologies, LLC

Roadmap



Tentative Timeline

Week of	Monday	Tuesday	Wednesday	Thursday	Friday
7/11				Achieve cross compilation for ARM using CMake	
7/18	Develop real-time functionality – amplitude detector				Integrate Wi-Fi detection algorithm
7/25					

Introduction to Kent

- Pursuing B.S., Electrical Engineering
- Specializations
  - Communication and Signal Processing
  - Controls
- Hobbies
  - Steel-string acoustic guitar
  - Keeping and breeding fish
  - Enjoying the outdoors

A. James Clark School of Engineering

https://eng.umd.edu/branding-toolkit

Department of Electrical & Computer Engineering

https://cra.org/job/university-of-maryland-college-park-instructional-faculty-lecturer/instructional-faculty-lecturer-3/

College Park Scholars

Public Leadership

https://kentchesley.weebly.com/


geon
Technologies, LLC