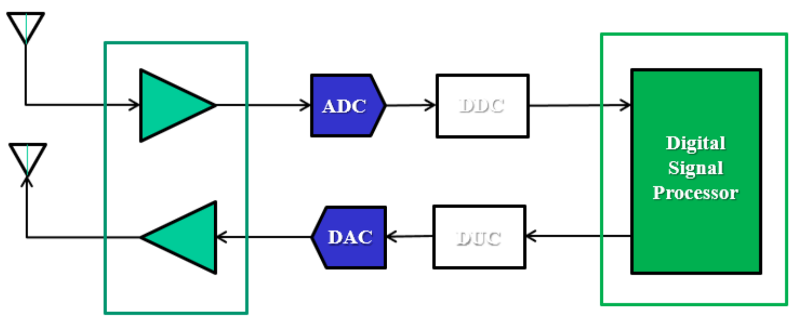
ECE4513 Digital Communications System Laboratory

Dennis Silage, PhD
silage@temple.edu

RTL Software Defined Radio

In this Laboratory you will have the hardware experience of using a software defined radio (SDR) which you can continue to use afterwards. The book *Software defined radio using MATLAB and Simulink* is on *Canvas* and can produce a *life-long experience* in this electrotechnology area after the Laboratory.

The SDR is a merger of digital signal processing and wideband radio hardware. The ideal SDR receiver consists of an antenna connected to an analog-to-digital converter (ADC) followed by a digital signal processing (DSP) system to extract the signal of interest. The ideal SDR transmitter consists of a DSP system where the information signal is inputted to a digital-to-analog converter (DAC) which directly interfaces with an antenna.



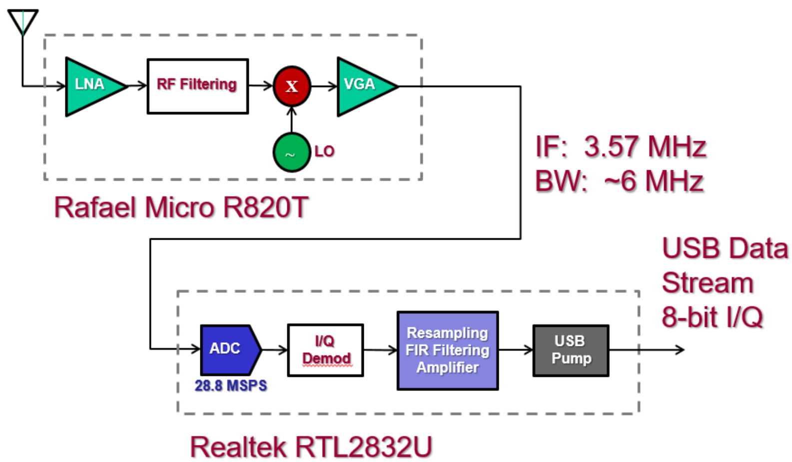
The DDC and DUB are a *digital down converter* and *digital up converter* which lowers and raises the carrier frequency of the received or transmitted signal. However, only a receive SDR is provided here.

The SDR hardware platform here is the RTL-SDR which has an active user community. The form factor of the RTL-SDR is similar to a large USB memory stick and referred to as the *RTL-SDR USB dongle*. The RTL-SDR dongle contains two ICs: the Raphael Micro R820T radio tuner and the Realtek RTL2832U which contains an 8-bit ADC and USB *data pump*.

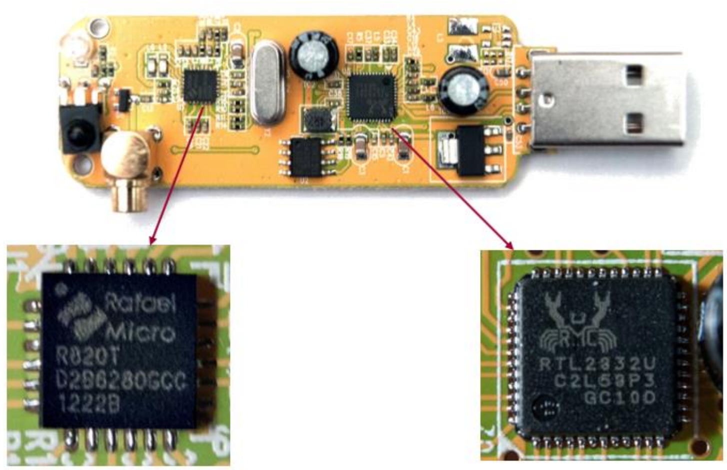
The R830T tuner is the radio frequency (RF) *front-end* for the SDR. A miniature coax connector for the antenna inputs the received signal to a low noise amplifier (LNA) providing a noise figure (NF) of about 3.5 dB. The tuning range of the R820T is 24 MHz to 1850 MHz. A frequency synthesizer inside the R820T tuner generates a local oscillator (LO) signal which is responsible for *down converting* the received radio frequency (RF) signal to an intermediate frequency (IF) for processing.

The tuning resolution of the RTL-SDR is 1 Hz. Gain control is also provided, both at the LNA and at the output via a variable gain amplifier (VGA). An automatic gain control (AGC) is a signal strength sensing circuit and algorithm to feedback a control signal to the gain control of the LNA and VGA.

The RTL2832U is the DSP which includes additional filtering and down sampling of the IF signal provided by the R820T tuner. The ADC produces 8-bit inphase (I) and quadrature (Q) unsigned sample values. The USB pump converts the samples to signed 8-bit values and parallel I and Q streams as a USB interface to a PC.

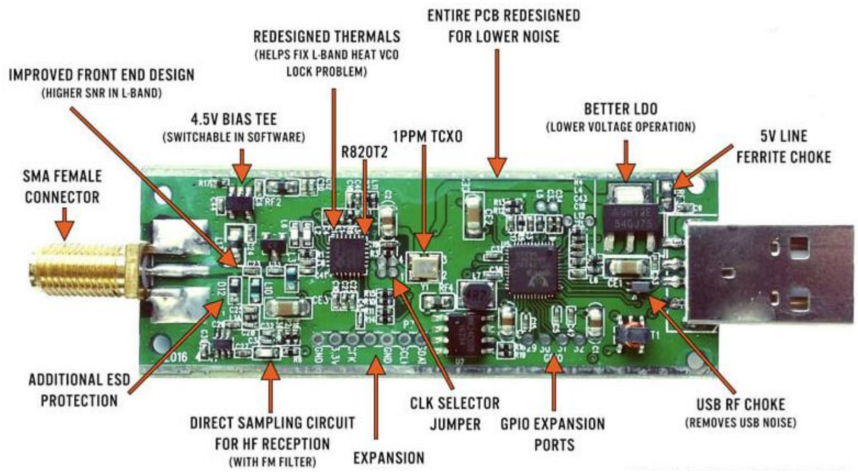


The original RTL-SDR dongle was basically configured as shown below.

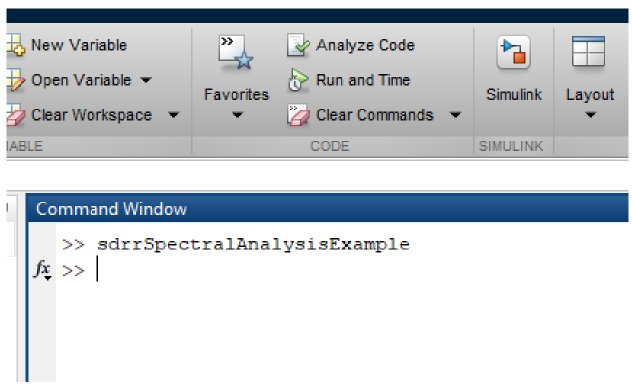


However, the Laboratory RTL_SDR dongle provided is improved as shown below. You will also have to install the *Simulink Communications Toolbox* support for and verify the performance of the RTL-SDR dongle as described in detail on the accompanying PowerPoint description *RTL-SDR Intro.pptx* on *Canvas.*

The Laboratory tasks are as follows and each are to be describe in detail and results shown in the Report.



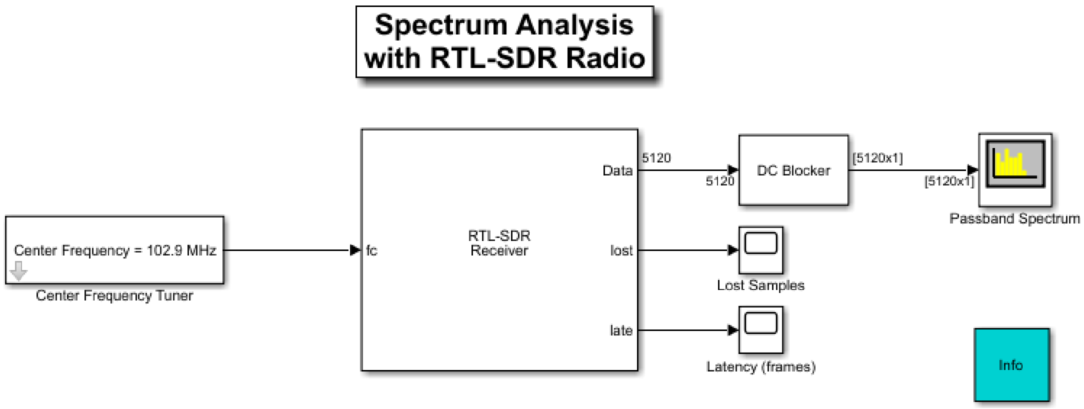
1. After proper installation and verification of the *Simulink Communications Toolbox* support for the RTL-SDR enter *sdrrSpectralAnalysisExample* in the MATLAB command window which launches the *Simulink Spectrum Analysis with RTL-SDR Radio* model.

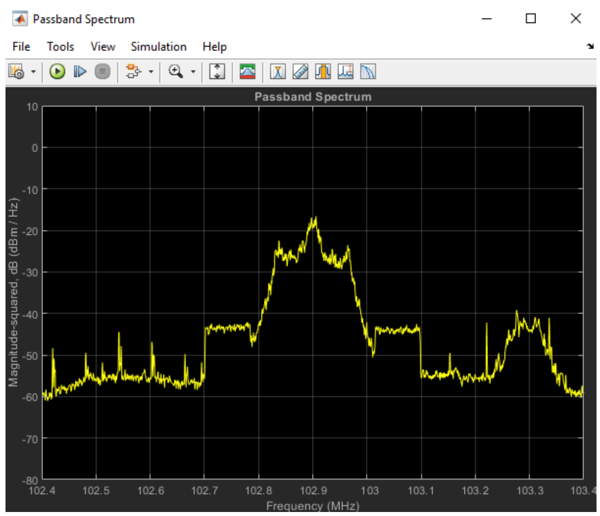


With adequate gain and a reasonable location, the large "rabbit ear" antenna provided should show a complete spectrum of an FM Broadcast station. The default gain is AGC but can be changed to a dialog or input port is warranted. The spectrum shown below is for WMGK with a center frequency of 102.9 MHz. Note the appearance of the digital modulation sidebands (rectangular pulses) as known as HD Radio.

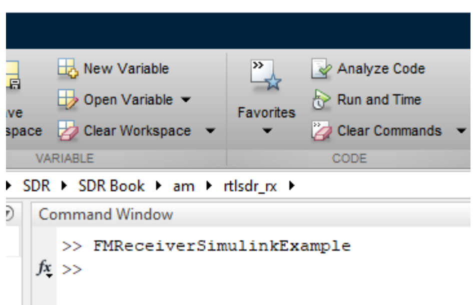
Set the center frequency to that of any another station in the FM broadcast range (88-108 MHz) and capture the spectrum for the Laboratory Report. For this FM broadcast station are digital sidebands present? Repeat this for five other stations. Are there some stations without a digital sideband? Perform a reference search and cite whether any of these stations are transmitting a digital sideband.

What is the nominal spacing for the center frequency of an FM broadcasting station? Research and describe the modulation and bandwidth used for both analog FM and HD Radio on the FM broadcast band.

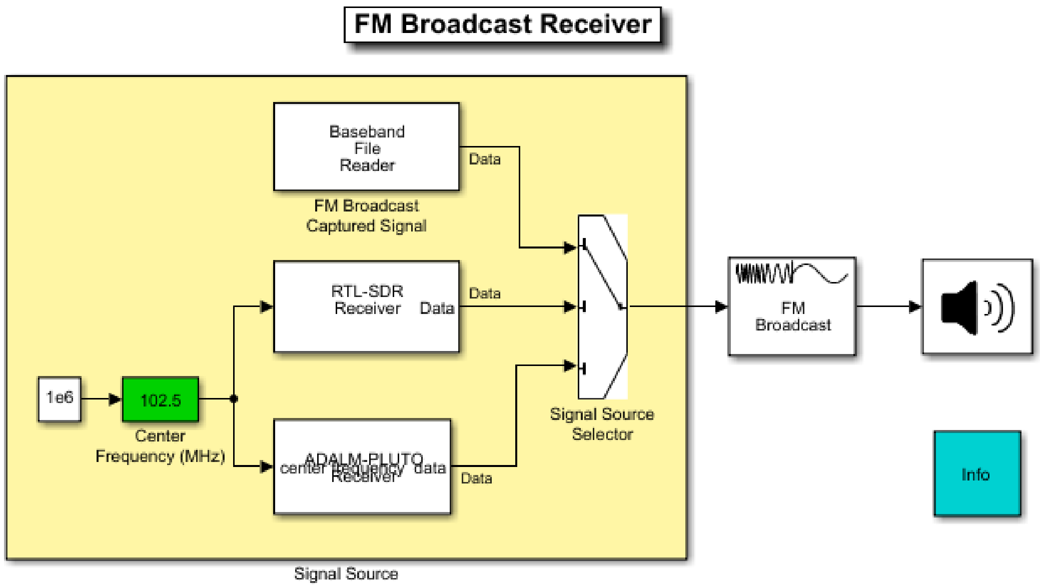




2. Stop the *sdrrSpectralAnalysisExample* and enter *FMReceiverSimulinkExample* in the MATLAB command window which launches the *FM Broadcast Receiver Model*. The Switch selects a prerecorded FM voice signal by default. Clicking on the Switch when the model is stopped selects the RTL-SDR receiver at the center frequency selected. This allows you to have an FM broadcast receiver on your PC using the RTL-SDR. The ADALM-PLUTO Receiver is another SDR and not available.

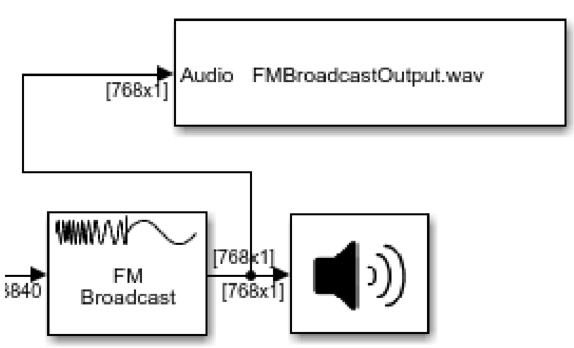


Change the center frequency higher from the nominal FM broadcasting frequency in steps of 5 kHz (0.005 MHz). At what offset frequency does the FM modulation become noticeably distorted? Modify the *FM Broadcast Receiver Model* by adding the *To Multimedia File* block from the *DSP Toolbox - Sinks* to write multimedia files (.avi), as shown below, to document the effect and be prepared to demonstrate the result. Use several different file names to differentiate the effect. Report this observation in the Laboratory Report.

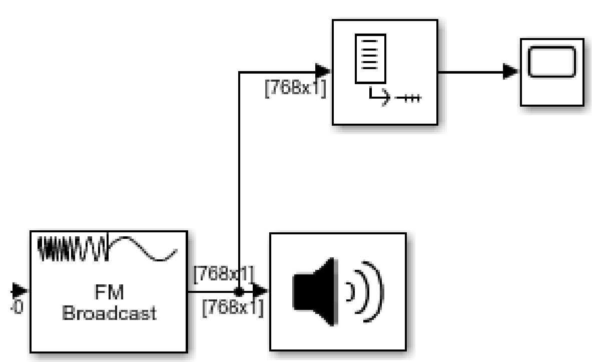


Next again modify the *FM Broadcast Receiver Model* to temporally plot the demodulated output as shown below. The *Unbuffer* block from the *DSP Toolbox – Signal Management – Buffers* converts the frame output of the FM Broadcast demodulator block to scalar samples to plot.

Stop the modified *FM Broadcast Receiver Model* and expand the temporal scope signal to demonstrate the characteristics of a typical voice or music signal. Research the signal expected. Does the observed signal agree with the expected result? Produce a temporal plot in the Laboratory report and describe the observation.



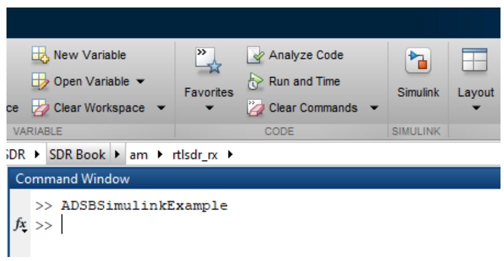
Multimedia file output



Temporal scope plot

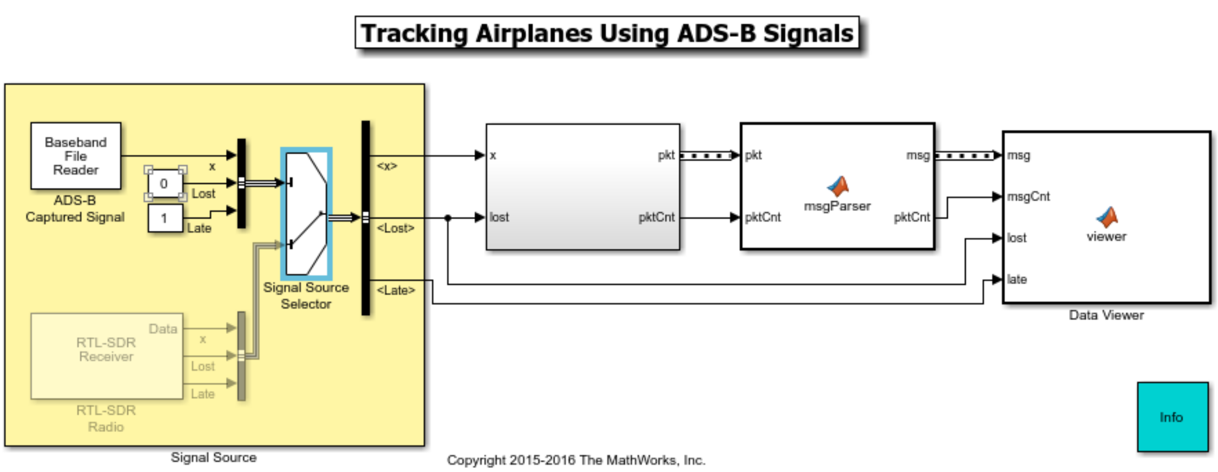
3. *Automatic Dependent Surveillance—Broadcast* (ADS–B) is a surveillance technology in which an aircraft determines its position via satellite navigation and periodically broadcasts it, enabling it to be tracked. The information can be received by air traffic control ground stations as a replacement for secondary surveillance radar, as no interrogation signal is needed from the ground. It can also be received by other aircraft to provide situational awareness and allow self-separation. ADS–B is *automatic* in that it requires no pilot or external input. It is *dependent* in that it depends on data from the aircraft's navigation system.

Enter *ADSBSimulinkExample* in the MATLAB command window which launches the *ADSB Receiver Model*. The Switch selects a prerecorded ADS-B signal by default. Clicking on the Switch when the model is stopped selects the RTL-SDR receiver at the center frequency selected in the model.

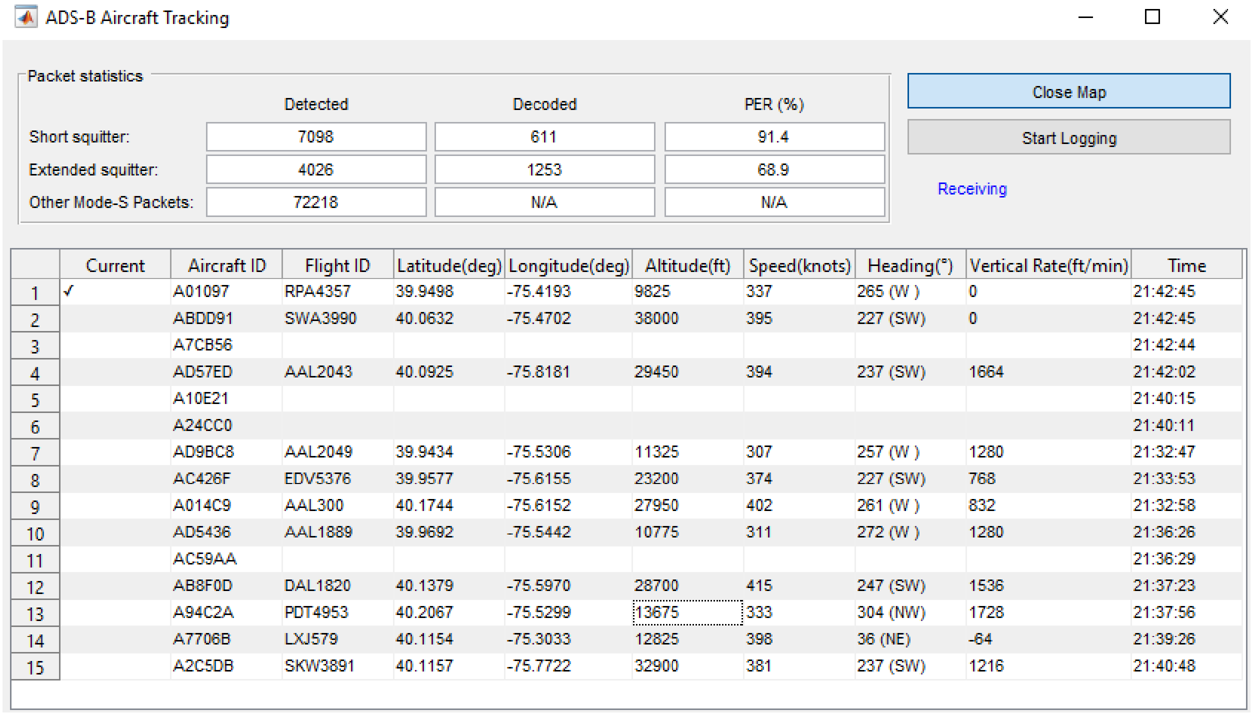


Run the ADS-B model and open (double-click) the *Data Viewer* which shows a list of current aircraft in your vicinity. The ADS-B model initially executes for 10 seconds but the simulation time can be extended to infinity by entering *inf.*

The *info* button opens the description of ADS-B Simulink simulation. What is the carrier frequency of ADS-B? What modulation method and data rate are used? Describe the use of *CRC* in the signal decoding. What is *squitter.* Show your *Data Viewer* and convert several of the aircraft latitude and longitude to a geographical location and their speed in knots to mph. The add-on Mapping Toolbox can produce a map of the aircraft locations. The aircraft shown are near my location in the western suburbs.



Data Viewer



ADS-B Aircraft Tracking

Packet statistics

	Detected	Decoded	PER (%)
Short squitter:	7098	611	91.4
Extended squitter:	4026	1253	68.9
Other Mode-S Packets:	72218	N/A	N/A

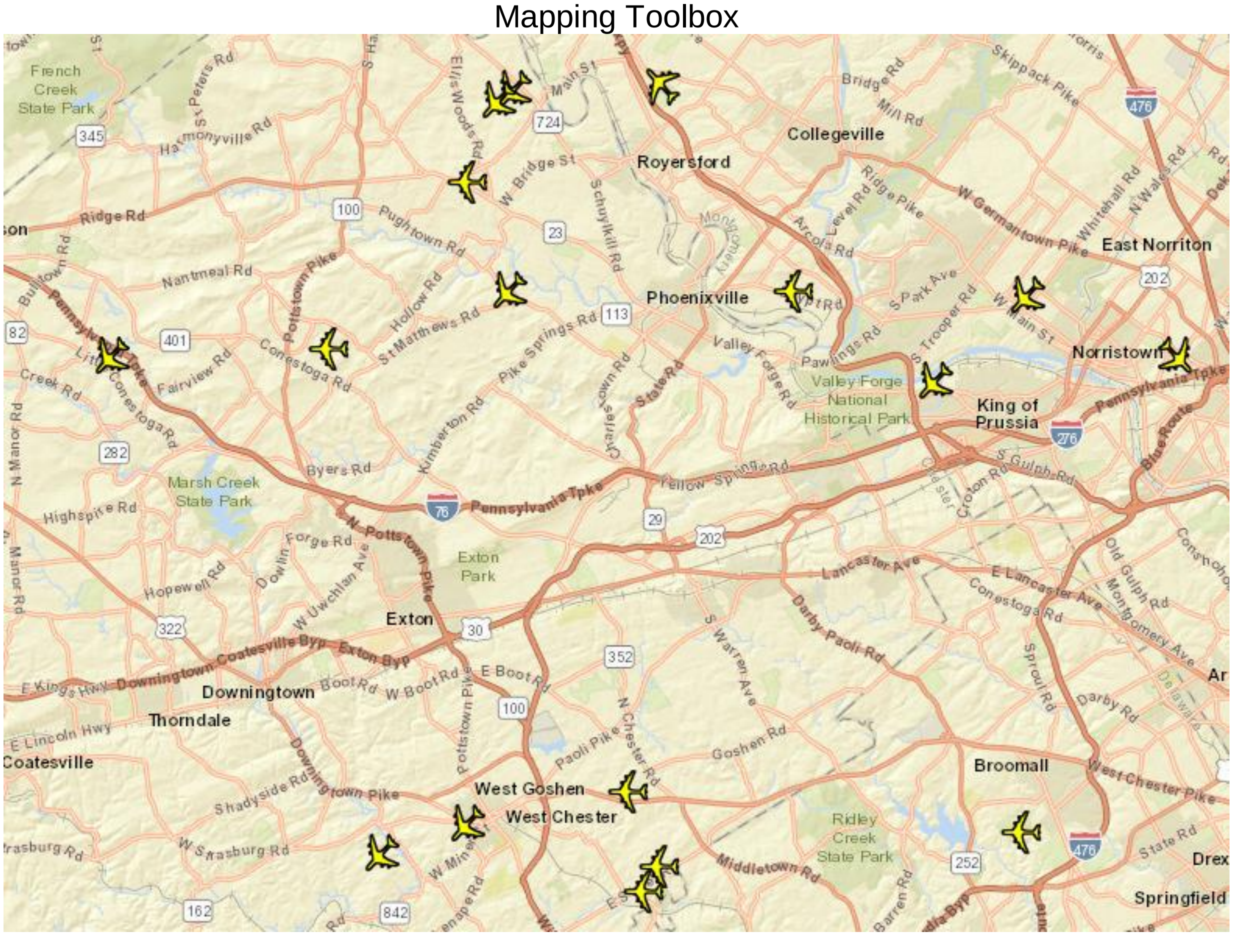
Close Map

Start Logging

Receiving

	Current	Aircraft ID	Flight ID	Latitude(deg)	Longitude(deg)	Altitude(ft)	Speed(knots)	Heading(°)	Vertical Rate(ft/min)	Time
1	✓	A01097	RPA4357	39.9498	-75.4193	9825	337	265 (W)	0	21:42:45
2		ABDD91	SWA3990	40.0632	-75.4702	38000	395	227 (SW)	0	21:42:45
3		A7CB56								21:42:44
4		AD57ED	AAL2043	40.0925	-75.8181	29450	394	237 (SW)	1664	21:42:02
5		A10E21								21:40:15
6		A24CC0								21:40:11
7		AD9BC8	AAL2049	39.9434	-75.5306	11325	307	257 (W)	1280	21:32:47
8		AC426F	EDV5376	39.9577	-75.6155	23200	374	227 (SW)	768	21:33:53
9		A014C9	AAL300	40.1744	-75.6152	27950	402	261 (W)	832	21:32:58
10		AD5436	AAL1889	39.9692	-75.5442	10775	311	272 (W)	1280	21:36:26
11		AC59AA								21:36:29
12		AB8F0D	DAL1820	40.1379	-75.5970	28700	415	247 (SW)	1536	21:37:23
13		A94C2A	PDT4953	40.2067	-75.5299	13675	333	304 (NW)	1728	21:37:56
14		A7706B	LXJ579	40.1154	-75.3033	12825	398	36 (NE)	-64	21:39:26
15		A2C5DB	SKW3891	40.1157	-75.7722	32900	381	237 (SW)	1216	21:40:48

Mapping Toolbox



This is a one week Laboratory for the week of April 15th and due no later than ***Monday*** April 22nd 11:59 PM with upload to Canvas. Note that the deadline is a Monday and not a Wednesday so that your results can be randomly demonstrated in the Laboratory recitation during the last full week of classes April 22rd.

Spring 2019