ECE422 Project: Antenna and Radio for VHF Communications

1 Introduction

In ECE422, you will complete a project in a team to build the antenna and RF front-end for a VHF radio system. This radio will be used to receive a variety of terrestrial and satellite transmissions in the VHF band, including:

1. Broadcasts from commercial- and government-operated stations;
2. Aviation-band transmissions, including voice and ACARS (Aircraft Communications Addressing and Reporting Systems) traffic;
3. Amateur radio transmissions, including both terrestrial and satellite.

The radio itself comprises:

1. An antenna tuned for the VHF band, designed with the gain for the targeted application in mind;
2. The RF front-end of the radio, to improve selectivity and sensitivity in links involving large distances;
3. The radio itself, which will receive, downconvert, and digitize the received signal;
4. A PC, which takes digitized samples from the radio to perform demodulation and, in the case of digital signals, decoding.

The first two items, the antenna and the front-end, are the focal points of the project, as you will design and/or assemble them. The remaining components are provided, and you will learn how to use them to test and use your completed radio system. This guide will work through major component of the radio and related activities for using each of them.

You can complete the project in three ECE422 labs, as well as outside of lab time. There are 3 regularly schedule PRA times set aside for project work, where you can attend the lab and have TA/staff support.

2 Your ECE422 Radio Kit

You should have received a kit that contains some of the items you need to complete the project. The contents of the kit are:

- An unpopulated printed circuit board (PCB) for the radio front-end

- Electronic components for building the radio front-end (see Section 2.2)
- A NooElec software-defined radio (SDR) dongle
- A coaxial cable for connecting the SDR to the PCB

Additional parts and hardware required for the assembly of the antenna are provided in the GB347 lab are are not included in the kit. Also, to perform measurements of the antenna, you will need access to the vector network analyzers in GB347.

2.1 SDR

The SDR is a core part of the receiver. It is a USB peripheral that implements a superheterodyne receiver. A block diagram of the NooElec SDR you are using is shown in Figure 1. As seen in the figure, the SDR comprises two major chips: the Rafael R820T2 tuner, and the Realtek RTL2832U demodulator.

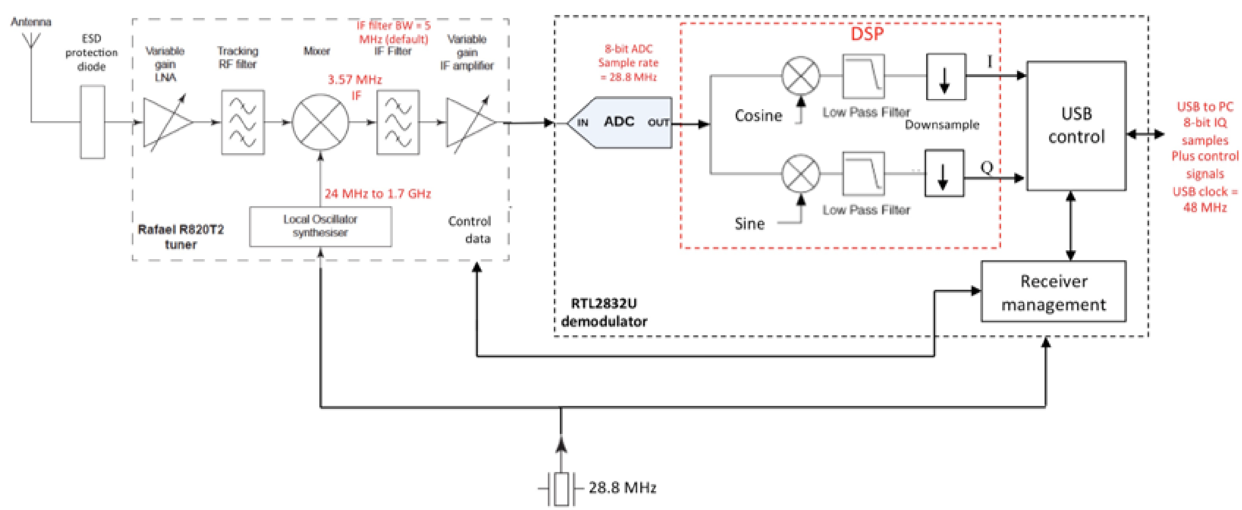

Figure 1: NooElec SDR block diagram

The R820T2 is the analog portion of the SDR. An incoming signal from the antenna is amplified using a variable-gain low-noise amplifier (LNA) and filtered using a bandpass filter. Both of these components are digitally controlled: the gain of the LNA can be programmed, and the centre frequency of the RF filter can be shifted electronically (though its overall bandwidth is quite wide). The subsequent signal is downconverted using a mixer to an intermediate frequency centred at 3.57 MHz. The use of an intermediate frequency (which is not at baseband) is the reason this radio is called a *superheterodyne receiver*.

The mixer is supplied with a local oscillator (LO) signal provided by a digitally controlled synthesizer, which can produce a frequency f_{LO} anywhere between 25 and 1750 MHz. This means that an RF signal at approximately the same frequency ($f_{RF} = f_{IF} + f_{LO}$) can be practically tuned to by the SDR. Since the mixer produces sum and difference products, the desired difference product at the intermediate frequency is filtered by a bandpass filter, which rejects the sum product.

This IF signal is amplified further using another variable-gain amplifier and passed to the RTL2832U. This chip digitally samples the IF using high-speed analog-to-digital converters. The signal is now in the digital domain, and processed using digital signal processing techniques. We can see two parallel mixers resembling the mixer in the R820T2, except that these mixers are realized computationally using digital signal processing (DSP). They are also feed their own LO signals supplied by a numerically controlled oscillator, and the two LO signals are 90° out of phase with each other (cosine and sine), realizing a *quadrature receiver* that outputs the *in-phase* (I) and quadrature (Q) signals that can be used for demodulation. These are baseband signals, and they can be downsampled to a lower sample rate since their highest frequency component is much lower than that in the IF signal. These digitally downsampled signals are passed on to the PC over a universal serial bus (USB) interface. While this chip is called a demodulator, the actual demodulation (converting the I/Q signals to an human- or computer-intelligible signal) is done on the PC. Each I and Q sample is represented using 8 bits (16 bits total.)

The last part visible in the block diagram is a 28.8 MHz crystal, which acts as a central clock for both chips.

On its own, the SDR is very powerful and versatile. However, the wideband nature of the SDR and the fact that the variable-gain LNA has at most 50 dB of gain, mean that it may not be sufficiently sensitive for weak signals, such as those used for space communications. Thus, we will be adding another stage to the "front" of this radio in the form of a front-end module, discussed next.

2.2 RF Front-End Module

The PCB in your kit implements the front-end module for the SDR. Your kit should contain the items listed in Table 1.

A schematic of the PCB is shown in Figure 2. Reading the schematic from left-to-right, we observe that the module functions as follows:

1. A *balun* is formed by T1, which converts the balanced antenna input (assuming you are using a balanced antenna) to an unbalanced signal compatible with the coaxial cable and planar transmission lines used on the PCB.

2. J1 is an SMA coaxial port that allows the antenna to be directly connected to test equipment or the SDR itself. It is a test port for when you are designing the antenna, and can also be used if you don't want additional filtering and amplification applied to the antenna signal before connecting it to the SDR.

3. The antenna signal is fed to a 3rd order Chebyshev filter formed by C1–C7 and L1–L3. This filter realizes a filter response with 3 dB cutoff points at 130 MHz and 160 MHz, which centres the filter around the amateur 2 m radio band from $144 - 148$ MHz. The passband ripple of the filter is designed to be 1 dB. The main purpose of this filter is to prevent signals outside the passband from entering the radio and, importantly, saturating the receiver given the additional amplification being provided by the front-end module. For

Type	Designator	Description
Magnetics	T1	BN-61-2402 binocular core
	L1–L3	100 nH inductor
	L4, L5	1 μH inductor
Passives	R1	150 Ω resistor
	R2	22 Ω resistor
	R3	470 Ω resistor
	C1, C2, C6, C7	5.6 pF capacitor
	C3, C5	1.8 pF capacitor
	C4	9.1 pF capacitor
	C8	330 pF capacitor
	C9	0.01 μF capacitor
Semiconductors	U1	26 dB MMIC amplifier
	D2	Power LED
Connectors	J1	SMA jack
	J2	SMA jack, board end launch
Hardware	–	PCB
	–	#6-32 screws, nuts

Table 1: Contents of the PCB kit

example, FM broadcast signals are very strong, and once amplified by the module, would likely overwhelm the receiver, making receiving other weaker signals impossible or difficult.

4. The amplifier is realized from a monolithic RFIC, a PGA-103+ from Mini-Circuits. This is an integrated amplifier providing approximately 26 dB in the band of interest. It also has a relatively low noise figure of 0.5 dB, though the overall noise figure of the front-end will be degraded by the fact that the filter precedes the amplifier. The amplifier is made unconditionally stable by the network comprising L5, R1, and C8. The amplifier is supplied DC power at the output port via a bias tee formed by L4 and C9: $+$5V is available from the SDR via the RF input port.

5. J2 supplies the filtered and amplified signal to the SDR receiver or test equipment.

6. D2 provides a visual indication of whether DC power is being received from the SDR or other biasing arrangement.

3 Antenna

You will be responsible for designing two antennas for the project:

1. A half-wave dipole for initial terrestrial radio testing; and

2. A Yagi-Uda antenna, a type of parasitically excited linear antenna array, for providing the high gain needed for satellite experiments.

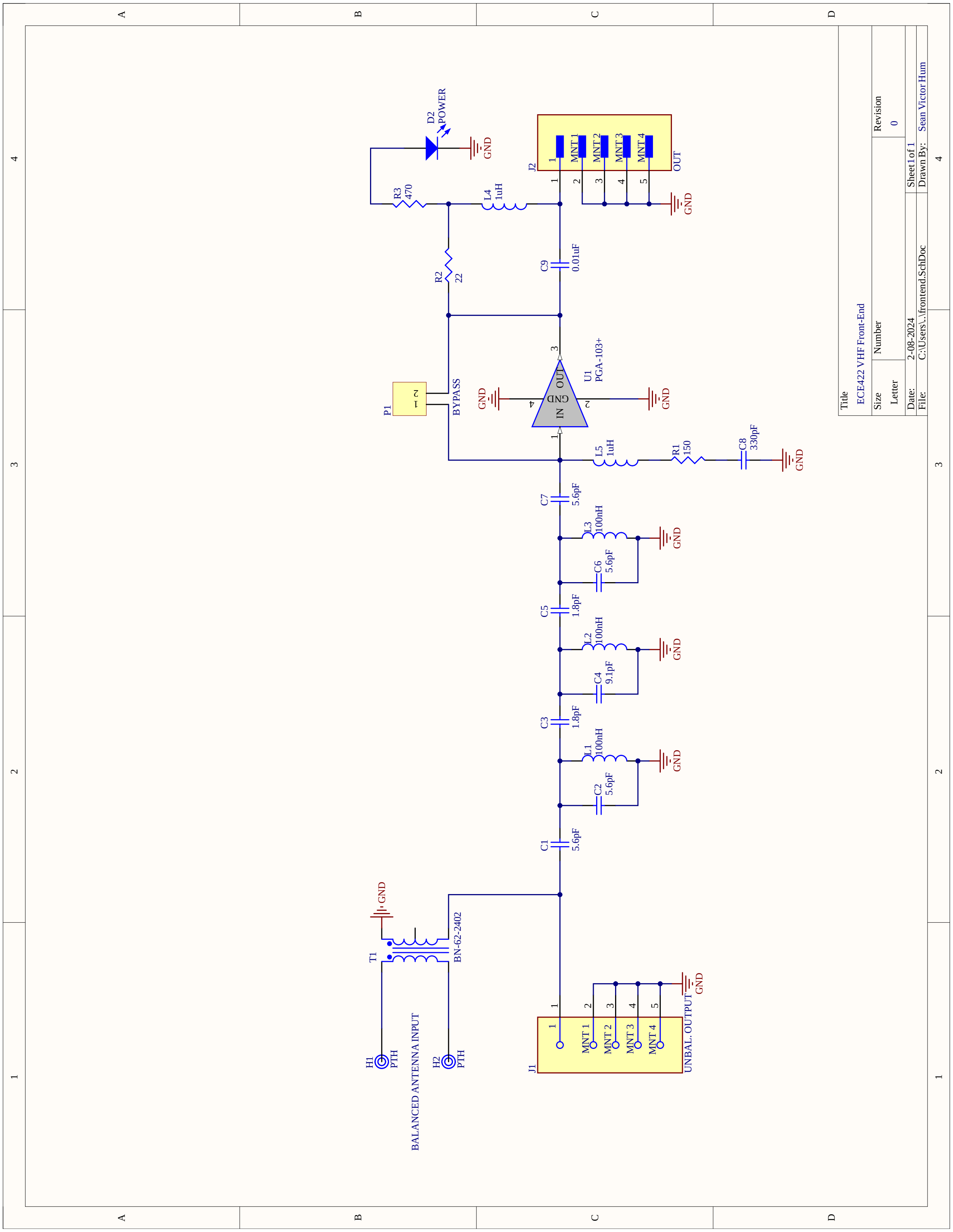

Figure 2: Schematic diagram of RF front-end module

3.1 Half-Wave Dipole

The half-wave dipole is fairly straightforward to construct. You will construct it from $1/2$" (12.7 mm) copper tubing, which will be mounted to a supporting structure and fed at the centre. Detailed instructions for cutting the tubing will be provided during the labs.

As discussed in the lectures, half-wave dipoles are *balanced antennas*, yet the transmission lines we will use to connect it to the radio and test equipment are coaxial transmission lines, which are *unbalanced*. Hence a *balun* is needed. A transformer balun (T1) is integrated into the PCB for the front-end receiver, to allow balanced antennas to be used with it.

3.2 Yagi-Uda Antenna Array

The Yagi-Uda is a type of linear antenna array designed to produce an end-fire beam, and is illustrated in Figure 3. Maximum gain is developed in the $+x$ direction. The distinguishing feature of a Yagi-Uda antenna is that only one antenna element, a half-wave dipole (labelled *D* in Figure 3), needs to be excited. The remaining elements, which are placed near the driven dipole, are *parasitically excited* by the dipole and these dipoles do not require connections to the RF source, and are essentially short-circuited at their centres. These parasitic elements are placed and sized in such a way that the correct amplitude and phase of the required currents are induced on them, producing the end-fire response that is the goal of the design. The elements are referred to as the reflector (*R*) and directors (*D1*, *D2*, *D3*) in the antenna. It is possible to have several reflectors and numerous directors comprising a Yagi-Uda antenna, but the design you will pursue will be smaller to keep the construction manageable. Elements are mounted along a physical boom that supports the elements, that is oriented along the x-axis.

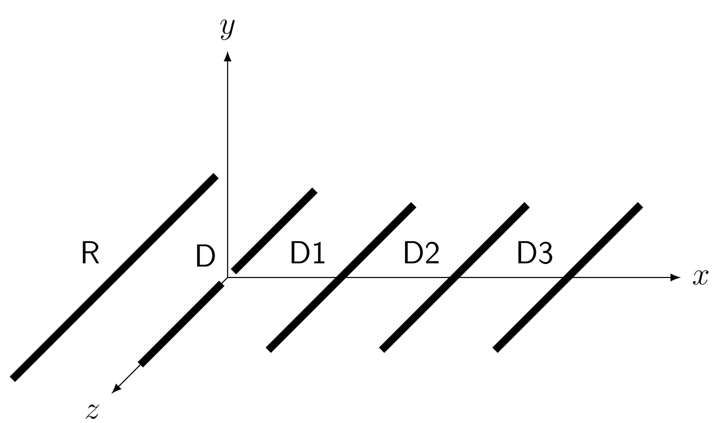

Figure 3: Yagi-Uda antenna

Details of the design of the Yagi-Uda array will be provided in one of the case study lectures. Useful design tables exist for determining the spacing and size of the reflectors and directors, depending on the final gain that is desired from the antenna. See the following link for an example:

`https://antenna-theory.com/antennas/travelling/yagi.php`

4 Recommended Procedures

There are 3 PRA sessions devoted to building your radio system. It is recommended that you carry out the activities per session as follows.

4.1 Session 1: Half-wave Dipole Construction and Antenna/SDR Testing

For this lab, your goal should be to build and tune the half-wave dipole for your receiver. After doing some basic measurements to confirm the dipole is tuned correctly, you will connect it to your SDR and try to pick up some terrestrial broadcast signals. This is an important step, because you will also calibrate the frequency synthesizer in the SDR by doing so.

4.2 Balun Assembly

T1 is wound on a BN-61-2402 binocular core included in your kit. It has a primary and secondary consisting of 2 bifilar turns. Using magnet wire provided to you in the lab, twist it into a bifilar-style wire. There are many ways you could do this. You could clamp one end and twist the other using a very low-speed drill. Or, simply twist the wire by hand. The number of twists is not critical. Your twisted wire should resemble that shown in Figure 4.

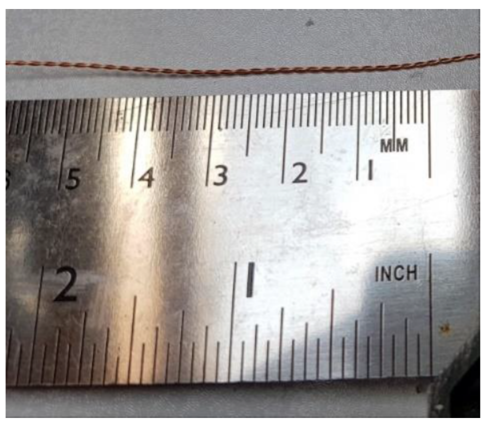

Figure 4: Bifilar wire [figure courtesy Hans Summers]

Wind 2 turns of this wire on the core. One "turn" means that the wire goes through both holes, so that the wire come out the same side of the core. Leave enough excess wire so that both ends are a few centimetres long. Cut the end of the wire where you folded the wire over and untwist both wire-pairs, so that you have four wire-ends to work with.

Next, you need to determine which wires are which. Begin by scraping the enamel off the wire ends, using a sharp knife or sandpaper. Using a multimeter to check continuity. One wire will be the primary of transformer, while the other will be the secondary. Since the turns ratio is 1, it does not matter which you use for which.

Refer to Figure 5 for the PCB artwork associated with the balun. The primary wire needs to be inserted into holes 4 and 5 on T1. The secondary should be inserted into holes 1 and 3. Nothing will be placed in the other holes (labelled "2") in the T1 footprint.

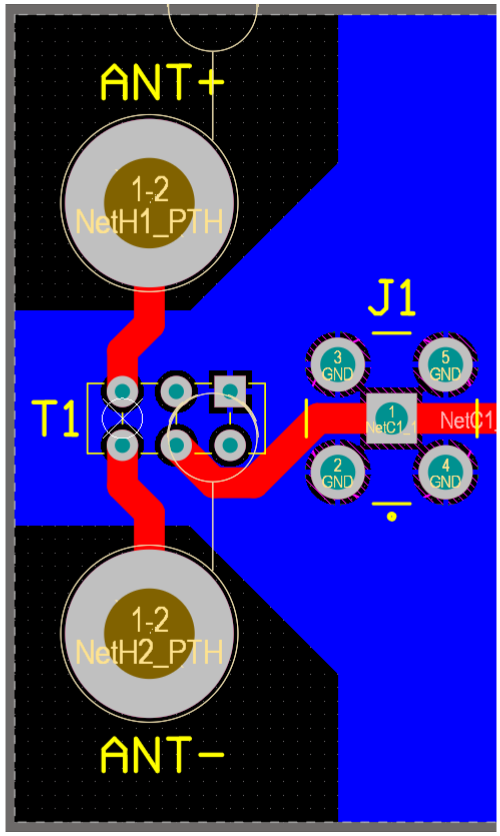

Figure 5: Balun PCB artwork

Ensuring that the wire enamel has been removed, solder your balun into place. You may need to apply heat for longer than usual in order to burn away any remaining enamel in order to make a good connection. A frequent problem is that you will think the balun is soldered in place but in fact there was enamel remaining on the wire, so that there is not actually an electrical connection between the wire and the board at the soldered location. Check continuity: there should be continuity between the two antenna "terminals" on the PCB labelled "ANT+" and "ANT-". There should also be continuity between the centre pin of the J1 footprint and the ground plane (back) of the board. Do not continue until you have achieved continuity for both these cases.

Solder the SMA connector J1 in place. The rest of the board does not need to be populated at this time.

4.3 Half-wave Dipole Assembly

Here are detailed steps for the assembly of the antenna:

1. Determine the lengths of the two halves of your half-wave dipole, and increase them by approximately 10% for margin.

2. With the help of the lab manager, cut two copper pipes to length and secure them to a wooden boom using the fasteners provided. Include fasteners right at the inner ends of the dipole-halves so that you can attach wire feedlines there.

3. Assemble and solder the balun to the PCB using the procedure outlined below.

4. Solder J1 to the PCB.

5. Calibrate a VNA for a one-port measurement in the $100 - 200$ MHz range.

6. Cut a pair of wires and use them to connect the PCB antenna terminals to the antenna. For the PCB, there are holes into which screws and nuts can be inserted to secure the wire to the PCB. On the antenna side, it should be possible to wrap wire around the fasteners holding the dipole halves in place.

7. Measure the antenna's input reflection coefficient and note the location of the resonance. Cut back the dipole symmetrically until the resonance is centred around 142 MHz.

Record your antenna dimensions, and also the measured S11 to a file. Include these in your report.

It is recommended that for your very first tests with the radio, that you populate only the balun portion of the radio receiver PCB, and keep the rest of the board unpopulated except for J1.

4.4 SDR Testing

For this part of the lab, you will connect the half-wave dipole you constructed to the SDR peripheral, in order to become useful with its operation. This step is also required in order to *calibrate* the SDR.

Calibration is required, because the local oscillators used on the SDR do not have infinite precision, which means that the radio will not tune precisely to the exact frequency you will request in the software. The error is not very large, but is significant enough to make it difficult to properly tune in to specific frequencies if it is not corrected for. Let us say the actual measured frequency is f_m, while the desired signal frequency is f_s. Then,

$$f_m = f_s \pm f_e, \tag{1}$$

where f_e is the frequency error of the device. The *frequency accuracy* is usually specified in *parts per million (ppm)* relative to the signal frequency. For example, if you are trying to synthesize a 100 MHz signal, and the frequency accuracy of the device is 25 ppm, the frequency accuracy of the synthesizer will be

$$f_m = 100 \text{ MHz} \pm \frac{25}{1,000,000} \cdot 100 \text{ MHz} = 100 \text{ MHz} \pm 2.5 \text{ kHz}. \tag{2}$$

Since the actual frequency may deviate by $\pm 2,500$ Hz relative to the frequency we want, this could lead to tuning errors when trying to tune to a signal precisely at 100 MHz. The actual frequency error will be unique to the SDR that you are using. Our goal is to determine this frequency error and systematically tell the SDR software to correct for it.

This can be done by tuning the SDR to a known reference signal. However, this is difficult to do, because we can't just use other signal sources in the labs as references, because they too could be producing their own frequency errors. We need a very well calibrated frequency standard, such as an atomic clock comprised of cesium oscillators and hydrogen masers, similar to what is used on GNSS satellites.

Luckily, terrestrial radio stations exist whose sole purpose is to be an international time or frequency standard. These include WWV, WWVH (origin in the United States), and CHU (in Canada), but these operate at shortwave frequencies far from where we are using the SDR.

A practical option is to use broadcast radio stations in the VHF band, which should be sufficiently accurate for our purposes. We can use commercial broadcast stations for this purpose, but this will not be very accurate, so we will use Canada's national Weatheradio network for fine-tuning. Weatheradio Canada transmitters broadcast using one of seven frequencies: 162.400, 162.425, 162.450, 162.475, 162.500, 162.525, or 162.550 MHz, all of which lie within the passband of our radio system. Toronto's Weatheradio station broadcasts a narrowband FM signal at 162.400 MHz.

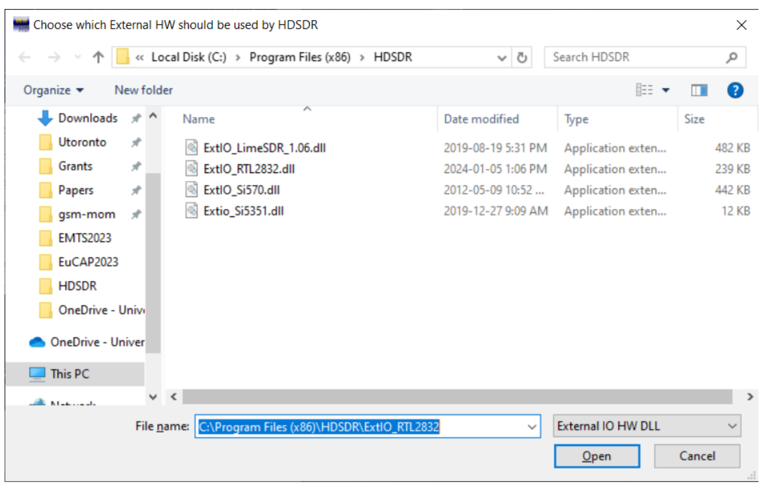

Figure 6: Dialog box for DLL file

For initial testing of the SDR, we will use the half-wave dipole you have fashioned and connect it to the SDR, without requiring the rest of the front-end module to be completed. Only the balun T1 and SMA connector J1 need to be populated on the board.

WARNING: The SDR provides DC on the RF input connector, which is used later to bias the amplifier in RF front-end module. At this stage, we are just using the balun portion of the board. **If you connect the SDR directly to J1, the DC from the SDR will be shorted to ground by T1 and the SDR may be damaged.** To alleviate this, a bias tee (which you will use in later labs) should be inserted between J1 and the SDR, to AC couple the SDR's input to the antenna. See Figure 7.

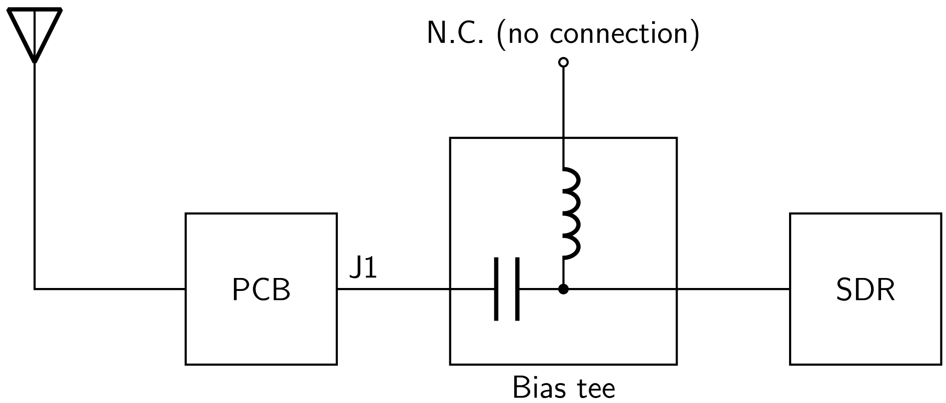

Figure 7: Usage of bias tee to AC couple the SDR to the antenna

The procedure is as follows:

1. Connect your completed half-wave dipole to one of the SMA terminals on the bias tee using a male/male SMA cable or adapter. It does not matter which terminal on the bias tee you use.
2. Connect the SDR to the other terminal of the bias tee using a male/male SMA cable.
3. Connect your USB SDR to one of the lab stations in GB347. It should automatically recognize the device.
4. Open the HDSDR application from the Start Menu. If the program prompts you for a DLL file, please choose the `ExtIO_RTL2832.dll` file as shown in Figure 6.
5. Ensure you choose a sound device to list to the demodulated output in HDSDR. You can do this by selecting the Soundcard [F5] menu and choosing an appropriate sound device under the "RX Output to" section.
6. HDSDR will open a waterfall display similar to that shown in Figure 8. The usage of HDSDR will be demonstrated in the lab, but using the mouse you can easily control the SDR to tune in to specific frequencies. Important SDR control parameters can be specified by pressing F8 or pressing the `SDR-Device [F8]` button. This will reveal a small dialog box as shown in Figure 8.
   - Automatic gain control (*AGC*) for both the tuner and RTL device should be disabled for now. This allows you to control the gain of the tuner's LNA using the slider.
   - The *Sample Rate* controls how much frequency bandwidth can be shown at a time. Larger sample rates allow a wider band of frequencies to be displayed, at the cost of higher noise in the signal (lower signal-to-noise ratio).
   - Importantly, the *Frequency Correction* allows you to specify a correction to the SDR frequency. If, by setting the frequency of the SDR, it is tuning too high in frequency, **a larger ppm value** can be used to correct this.

7. It is easiest to calibrate the radio on very strong signals to start, such as FM radio broadcast signals. These are good choices because they are easy to receive, but not great choices because the bandwidth of the signals is very large (100 kHz) and hence it is difficult to finely calibrate the radio on such a wideband signal – we'd prefer to use a narrowband signal, ideally a sinusoid, to tune the radio. Let's begin by tuning the radio to 99.1 MHz, which is CBC Radio One in Toronto. Set the LO frequency to exactly 99.1 MHz.

8. Since the FM radio stations use wideband FM (WFM) rather than narrowband FM (NBFM), you will have to press the `Bandwidth [F6]` button and choose a wider demodulation bandwidth to hear it correctly. Select 192,000 kHz.

9. You will likely notice that the voice-band signal from CBC is not centred at 99.1 MHz, but some other frequency, due to the inaccuracy of the synthesizer in the SDR. You can try tuning to that frequency using the mouse to first confirm that it is in fact CBC. Then tune back to 99.1 MHz once you have confirmed it is the correct station.

10. Press F8 and adjust the Frequency Correction value until the voice-band signal is centred approximately at 99.1 MHz as desired. Write down the ppm value!

11. Next, tune to the Weatheradio station at 162.400 MHz. Switch back to narrowband FM using a 12000 kHz bandwidth in the bandwidth window.

12. Similar to the case before, you should observe that the Weatheradio station is not precisely at 162.400 MHz. We can correct the remaining frequency error in the same way as we did for CBC, but since the signal is narrowband, we can get a much more accurate ppm value. Write it down!

The calibration of your SDR is now complete. Since HDSDR cannot remember the ppm value for your SDR, you should record it somewhere easy to access, as you will have to input this correction every time you use your SDR.

You can use the remainder of the lab the browse around the VHF spectrum with your antenna and receiver! Do some research and make observations on other interesting stations you find in your final report. Include screenshots from HDSDR.

4.5 Session 2: Terrestrial Radio Transmissions and Yagi-Uda Antenna Assembly

Once you have tuned your half-wave dipole and calibrated your SDR, you are in a position to use them together to receive and decode some interesting digital radio transmissions. We can then make the antenna more directive, by transforming it into a Yagi-Uda antenna, to increase its sensitivity for more challenging radio links tackled later in the project.

4.5.1 ACARS Reception

NOTE: It may be more productive to complete this activity first, before building the Yagi-Uda antenna discussed in Section 4.5.2. The reason is that ACARS will work best if you use a non-directional antenna to receive these signals, since they originate from aircraft that may be dispersed

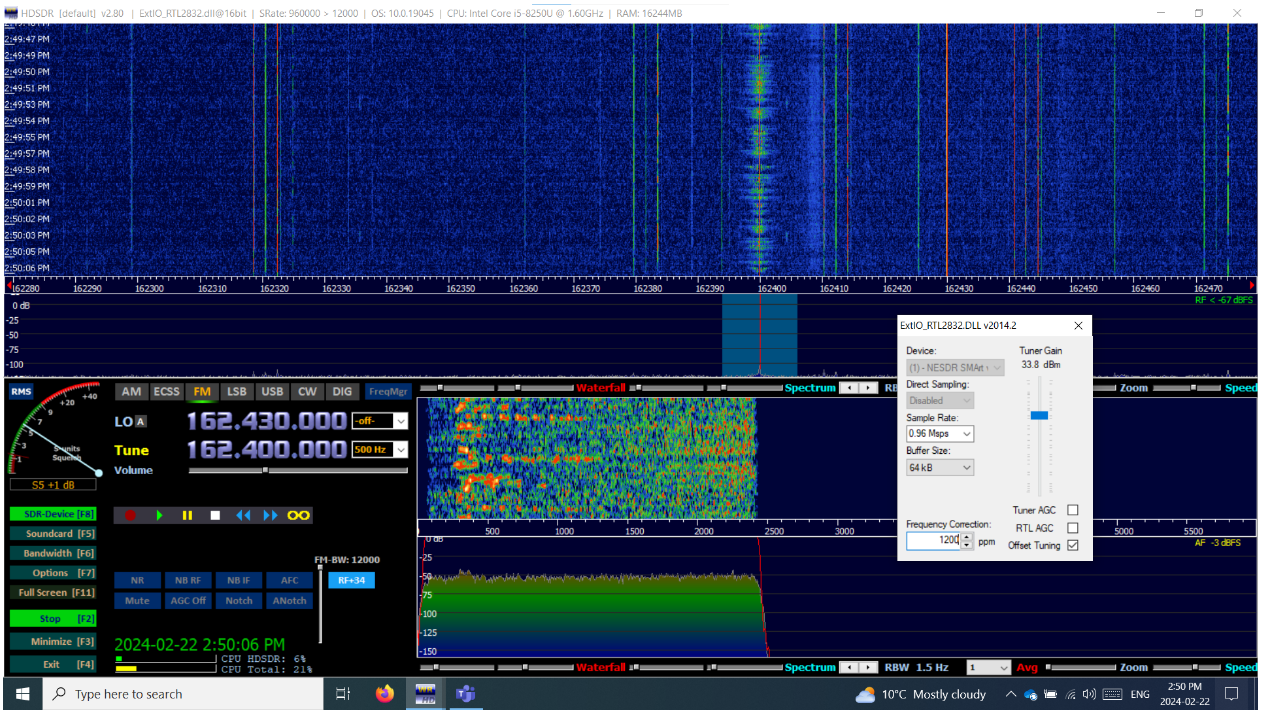

Figure 8: Main waterfall window with SDR control window

over a wide angular region around the antenna. The Yagi-Uda can be used, but then it is best to aim it towards an airport (Billy Bishop or Pearson).

This part requires that the dipole be decently tuned and your SDR peripheral calibrated. We will use them together to receive Aircraft Communications Addressing and Reporting System (ACARS) traffic, which is used by civilian aviation authorities for sending and receiving important information related to aircraft. ACARS messages are sent by both aircraft and ground stations on airports in a digital format (binary information) to provide:

1. Air traffic control messages, especially for requesting or providing clearances;
2. Aeronautical operational control messages; and
3. Airline administrative control messages.

Messages are broadcast on several bands, including the VHF band. Some ACARS links are purely terrestrial, while others can involve satellites. We will be exploring terrestrial VHF ACARS links with your radio.

ACARS messages are transmitted digitally at 2400 symbols/second using a modulation scheme known as minimum shift keying (MSK), a type of continuous-phase frequency-shift keying which maps different symbols to frequencies. The channel bandwidth is only 3 kHz, so you can easily "listen" to ACARS messages since the waveforms are in the audible range. Obviously, a digital decoder is needed to make sense of the actual signal. We will use a program called Black Cat

ACARS to decode the audio signals demodulated by HDSDR and decode them into intelligible messages.

Commonly used ACARS frequencies are:

- 131.475 MHz (used by Air Canada)
- 131.55 MHz (ARINC, especially at Billy Bishop airport)
- 129.125 MHz (used by ARINC)
- 136.975 MHz (used by ARINC)
- 136.850 MHz (used by SITACOM)

You can easily find and locate ACARS traffic by tuning your SDR around 131 MHz in HDSDR and choosing the bandwidth / sample rate so that you can monitor the entire spectrum between about 130-140 MHz. You can also focus on one frequency (recommended); the 131.475 MHz channel used by Air Canada is very active. Watch for short bursts of traffic on the waterfall centred at the frequencies listed above. Centre your demodulation window on one of these signals and set the receive bandwidth to about 5 kHz, and the demodulator to AM. The waterfall should resemble something like that shown in Figure 9.

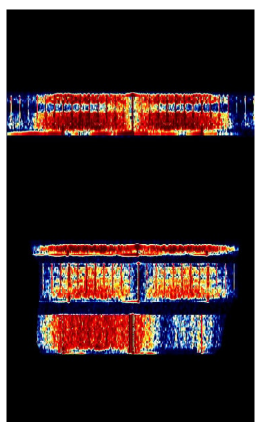

Figure 9: ACARS waterfall [source: www.sigidwiki.com]

While you can list to the signal from the speaker, to decode it, we will need an audio device that can accept the audio output you are listening to from HDSDR. For example, if you are using a laptop, it may have a microphone input jack. In the GB347 lab, USB sound cards are provided that you can plug into the workstations there to provide a sound card with an audio input. Plug in the USB sound card to an available USB port.

To decode the ACARS signal, you will need an appropriate ACARS decoding program that will take the demodulated audio signal you've received, and do the appropriate MSK demodulation of the frequency tones to yield a digital message. On Windows, this can be accomplished using a program called Black Cat ACARS, which can be downloaded from the following link:

https://blackcatsystems.com/software/black_cat_acars_decoder.html

In the ZIP file from the download, you can run the Black Cat ACARS executable, which will present you with a screen such as that shown in Figure 10. Under the Sound Input drop-down menu, select the USB audio device you have just plugged in to the PC.

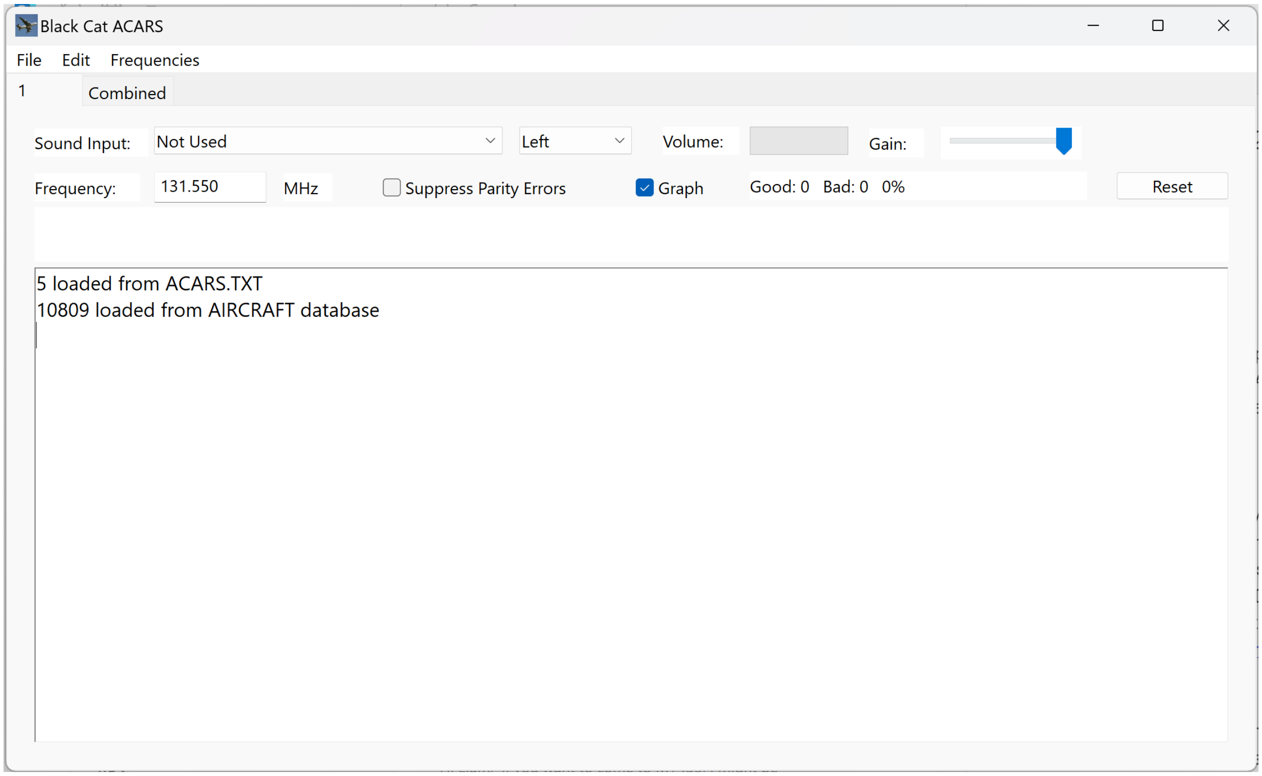

Figure 10: Black Cat ACARS window

To collect the audio output from HDSDR, use a 3.5mm stereo cable for the connection. Connect one end to the headphone/speaker output on the PC, and the other end to the microphone input of the USB sound card you are using. There is a volume indicator on the same line as the Sound Input item in ACARS, which should become animated if there is an audible signal being presented at the mic input.

If you are properly tuned to an ACARS frequency, and a valid ACARS signal is received, Black Cat ACARS should output the decoded message in the text window in the bottom half of the application window. For example, you may see a signal resembling this:

```
.C-FYJG  H1 50   [19 Jan 2024   16:35:44][ACA Airbus A319 0670] (131.475)
D44A RV1658 #DFBA04/A31904,1,1/CCC-FYJG,JAN19,213513,CYYZ,KTPA,1658/C105,
70131,4000,00,0100,0,0010,00,X/CEN112,01918,159,249,6043,267,C73006/EC731975,
00494,00559,00168,D3/EE731765,75591,03555,
```

The ACARS message format is standardized, and we will highlight a few features here.

On the first line, the `.C-FYJG` indicates this message is related to an aircraft with registration / tail number C-FYJG (this registration is usually visible on the fuselage of the aircraft). Next, the `H1` designator indicates the message type, which here refers to a message "to or from the terminal". These messages are used to transmit various flight and/or engine performance data as well as navigation and other parameters. Different airlines appear to use different formats and codes.

The remaining part of the first line is created by Black Cat ACARS: the message is time stamped, and the program has decoded part of the message to indicate the details of the aircraft: an ACA (Air Canada) Airbus A319, receiving on frequency 131.475 MHz.

The second line shows a few more features, but most of the information is not human-readable and requires an additional program to format the information being shown. It is some sort of report including the flight number `RV1658`, which is an Air Canada Rouge flight number. Nevertheless, using this program successfully shows how we can demodulate and decode digital information to produce messages used every day by the aviation industry.

Log some of the aircraft you are able to hear (based on the first line of the ACARS transmissions) successfully with your radio, in your report.

4.5.2 Yagi-Uda Antenna Assembly

Once you have tested the radio with non-directional radio signals in Section 4.5.1, you can modify your half-wave dipole design into a Yagi-Uda antenna. While this could be done before attempting the experiment in Section 4.5.1, doing so will require you to point your antenna towards ACARS transmitters and therefore know their location, since a Yagi-Uda antenna is much more directional than the half-wave dipole. Thus, while it might be useful for ground segment antennas (i.e., you can point your Yagi-Uda antenna towards Pearson International Airport, or Billy Bishop airport), it might make it more difficult to receive transmissions from aircraft (which would require you to know where they are in the sky).

To build your Yagi-Uda antenna, you can simply add elements to the boom that you assembled you half-wave dipole to in Section 3.1. You should use the Yagi-Uda antenna calculator discussed in Section 3.2 to aid in the determination of the element lengths and distances from the driven element. Remember that all the parasitic elements are shorted at their centre, meaning that you do not need to cut two halves as you did for the half-wave dipole. You can mount them to the boom using the same fasteners you used for the half-wave dipole.

When you are finished constructing your Yagi, first measure the input reflection coefficient to ensure the antenna is still decently matched. You may observe a frequency shift and narrowing of the response due to the introduction of the parasitic elements, which is normal. As long as the antenna's reflection coefficient magnitude is -6 dB or lower (≤ -10 dB preferable), your antenna will work well.

Optional: you can attempt to roughly measure the E-plane pattern of the antenna using the procedure of Lab 1 with your antenna as the antenna under test. By using another group's half-wave dipole or Yagi as a stationary fixed receiver, and placing your own Yagi on a tripod and rotating it, it is possible to measure the rough pattern of the antenna to verify that it is producing the desired directional response.

4.6 Session 3: Radio Front-End Assembly

In this session, you will complete the radio by soldering together the front-end module PCB discussed in Section 2.2, and characterizing it on a vector network analyzer.

The purpose of the front-end module is two-fold:

1. It provides additional gain in the form of a low-noise amplifier (LNA) at the beginning of the signal chain. As you will learn in class, by using an LNA with low noise figure at the beginning of the radio chain, we can increase the signal level at the output with minimal impact on the signal-to-noise ratio. This extra gain can be useful for picking up very weak signals from satellites, which is the next goal for your SDR. There is not sufficient gain in the SDR on its own to reliable close the link budget without the LNA.
2. It provides bandpass filtering in the 130-160 MHz band. This is important, because strong FM broadcast signals can easily overwhelm our now-very sensitive receiver (specifically, the amplifiers in the SDR). If any of these become saturated because of strong incident signals, the amplifiers will not function properly and prevent realizing the gain we need in the 144-148 MHz band.

4.6.1 PCB Assembly

The recommended order of assembly for the PCB is as follows:

1. U1, the surface-mount MMIC amplifier, should be soldered first. This is preferably done with solder paste (available in GB347) but can alternatively be done using conventional wire solder if you are careful.
   - Using a small applicator, place some solder paste on the three pads associated with U1.
   - Using a pair of tweezers, place U1 on the pads.
   - Using a soldering iron, touch the tip to each of the pads to solder U1 in place.
2. The remaining through-hole components can be soldered together in any order.
3. It is recommended you solder J2 last.

NOTE: There is no component fitted in the place of P1. It is DNP (do not populate).

4.6.2 PCB Testing Using a VNA

To test that the PCB is working, a full two-port measurement of the S-parameters should be conducted.

1. Disconnect your Yagi-Uda antenna from the PCB.

2. Calibrate the network analyzer for a two-port measurement between $100 - 200$ MHz.
3. To power the RF board, you will need a bias tee. Use the same one you used as the DC block for antenna testing. Connect it as shown in Figure 11. You will need to configure and connect a $+5$V power supply to the bias tee. Set a current limit to around 200 mA in case something has gone wrong during the soldering.
4. With the PCB connected only to the power supply, turn on the power and ensure that the current draw is around 100 mA. The LED on the board should also illuminate. If it draws no current or draws too much current, power off the board and troubleshoot what may be wrong.
5. When the board is operational, connect it to the network analyzer and measure the S21 of the board.
6. The amplifier should develop around 26 dB of gain, while the filter should provide filtering with cutoff frequencies at around 130 MHz and 160 MHz. Record the S21 frequency response you measure to a file and reconcile with expectations in your report.

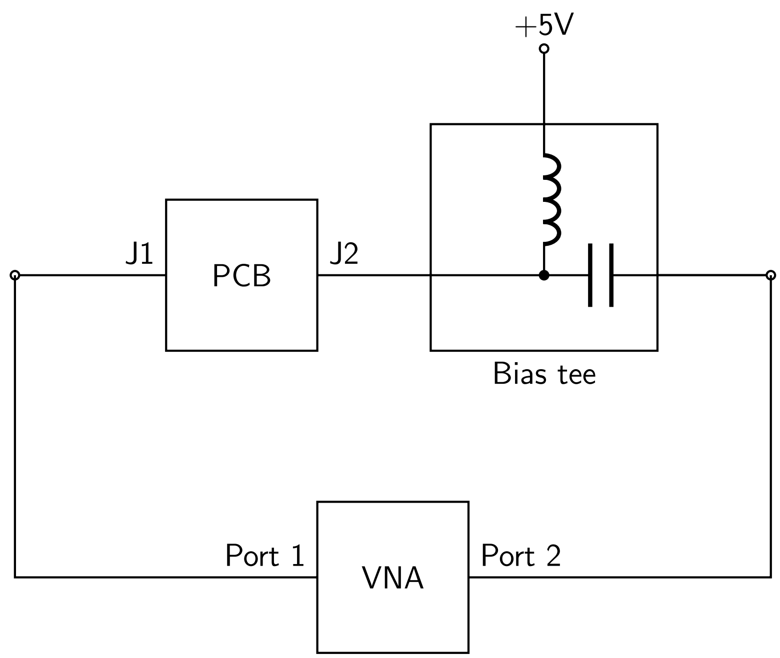

Figure 11: Usage of bias tee to power RF front-end for VNA characterization

4.6.3 Final PCB Testing With Your Antenna

With the amplifier in place, you can reconnect your dipole / Yagi-Uda antenna to the screw terminals of your PCB, and the signal from the antenna will be filtered, amplified, and delivered to J2. You can directly connector J2 to your SDR peripheral using an SMA cable. An on-board bias network on the SDR provides the $+5$V needed for the radio board, so when you plug your SDR into a USB port, the LED on your PCB should illuminate.

You should be able to test that everything is working by using your SDR to receive Weatheradio Canada or ACARS, as described in Sections 4.4 and 4.5.1. You will likely have to reduce the tuner gain due to the increase of gain provided by the external LNA. If you are able to receive these signals successfully, your completed project is ready to receive satellite signals, which will be discussed in the last experiment.