LAB 6

Frequency Modulation

6.1 Objective

This laboratory exercise introduces frequency modulation. This lab exercise is a nice illustration of the utility of the software defined radio approach, since the algorithms for creating and demodulating FM in software are much simpler than those used in the traditional hardware approach.

6.2 Background

Frequency Modulation

Frequency modulation (FM) was introduced by E.A. Armstrong in the 1930's as an alternative to the AM commonly in use at the time for broadcasting. The advantage to frequency modulation is that, for a given transmitted power, the signal-to-noise ratio is much higher at the receiver output than it is for AM. The digital version of FM, frequency-shift keying, has been in use since an even earlier date.

In FM, the frequency of the carrier is modulated to follow the amplitude of the message signal. To be more specific, if $m(t)$ is a message signal with peak value m_p, then the *instantaneous frequency* $f(t)$ of the carrier is given by

$$f(t) = f_c + k_f m(t), \tag{1}$$

where f_c is the carrier frequency and k_f is a proportionality constant called the "frequency sensitivity." The term $k_f m(t)$ is called the "frequency deviation" of the instantaneous frequency from the carrier frequency, and the *peak frequency deviation* $\Delta f = k_f m_p$ is an important FM system parameter. Given the instantaneous frequency, we can find the total instantaneous angle $\theta(t)$ of the carrier by integrating the instantaneous frequency. That is,

$$\begin{aligned} \theta(t) &= 2\pi \int_0^t f(\alpha)\,d\alpha \\ &= 2\pi f_c t + 2\pi k_f \int_0^t m(\alpha)\,d\alpha + \theta(0). \end{aligned} \tag{2}$$

Since the initial angle $\theta(0)$ is of no consequence, we can simplify the equations by taking $\theta(0) = 0$. The transmitted FM signal is then given by

$$
\begin{aligned}
g(t) &= A_c \cos\left[\theta(t)\right] \\
&= A_c \cos\left[2\pi f_c t + 2\pi k_f \int_0^t m(\alpha)\,d\alpha\right] \\
&= A_c \cos\left[2\pi f_c t + 2\pi \Delta f \int_0^t \left[m(\alpha)/m_p\right] d\alpha\right].
\end{aligned}
\tag{3}
$$

Figure 1 shows a 2 kHz carrier frequency modulated by a 200 Hz sinusoidal message.

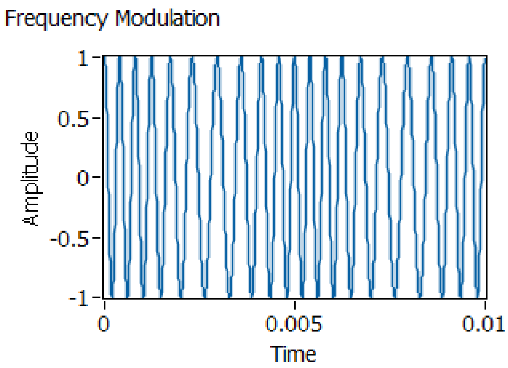

Figure 1. Frequency Modulated Signal

To create an FM signal using the USRP, the message signal is normalized to a peak value of one, multiplied by $2\pi\Delta f$ and integrated to give $2\pi\Delta f \int_0^t \left[m(\alpha)/m_p\right] d\alpha$. Next, the complex-valued signal $\tilde{g}(t)$ is formed, where

$$
\tilde{g}(t) = A_c e^{j2\pi\Delta f \int_0^t \left[m(\alpha)/m_p\right] d\alpha}. \tag{4}
$$

The complex-valued signal $\tilde{g}(t)$ is sent to the Write Tx Data function, and the USRP produces the FM signal.

The only tricky step in generating an FM signal is integrating the message. In discrete time we have

$$
\int_0^t x(\alpha)\,d\alpha \cong \sum_{k=0}^{n} x(nT)T, \tag{5}
$$

where T is the reciprocal of the IQ sample rate. If we write

$$
y[n] = \sum_{k=0}^{n} x(nT)T, \tag{6}
$$

then

$$y[n]-y[n-1]=\sum_{k=0}^{n}x(nT)T-\sum_{k=0}^{n-1}x(nT)T \\ = x(nT)T. \tag{7}$$

Equation (7) is the difference equation of an IIR filter. This filter can be implemented using the IIR Filter function found in the ExternalFiles folder. Use a "forward coefficients" array of $[T]$ and a "reverse coefficients" array of $[1 \quad -1]$.

Demodulation

FM demodulation is much easier to carry out using the USRP than it is using conventional hardware. The signal provided by the Fetch Rx Data function is $\tilde{r}(t)$ given by

$$\tilde{r}(t)=A_r e^{j2\pi\Delta f\int_0^t [m(\alpha)/m_p]d\alpha}. \tag{8}$$

This is identical to the expression given by Eq. (4), except for the magnitude A_r and the presence of noise (not shown in the expression). The angle of $\tilde{r}(t)$ is easily extracted using a *Complex to Polar function*.[6] Unwrap the angle before proceeding to the next step. Next, the unwrapped angle is differentiated, giving $2\pi\Delta f\left[m(t)/m_p\right]=2\pi k_f m(t)$. This result should be passed through a lowpass filter, since the differentiation step tends to enhance high-frequency noise.

To implement the differentiator, we recognize that in discrete time,

$$\frac{dx(t)}{dt}\cong\frac{x[n]-x[n-1]}{1}. \tag{9}$$

The differentiator can be implemented using an FIR Filter function from ExternalFiles folder. For the "FIR coefficients" array use $[1 \quad -1]$.

[6] For those familiar with conventional FM demodulation, this step implements the limiter.

6.3 Pre-Lab

Transmitter

1. A template for the transmitter has been provided in the file *Lab6TxTemplate.gvi*. This template contains the four interface functions along with a "message generator" that is set to produce a message signal consisting of three tones. The three tones are initially set to 1, 2, and 3 kHz, but these frequencies can be changed using front-panel controls. Your task is to add blocks as needed to produce the signal $\tilde{g}(t)$ of Eq.(4), and then to pass this signal to the Write Tx Data block. Set the carrier level A_c to a constant value of 0.9. Figure 2 shows how to implement the FM modulator described by Eq. (4).

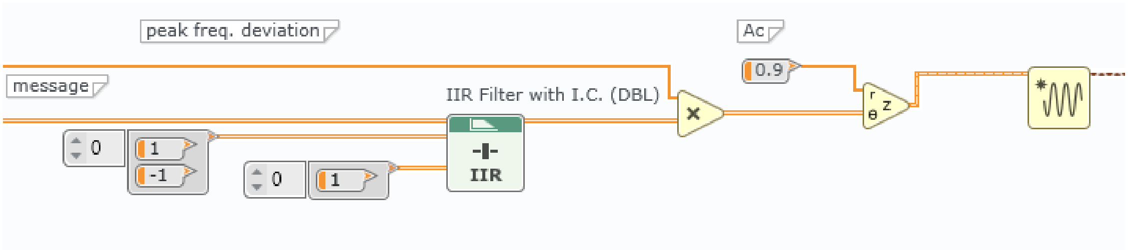

Figure 2. FM Modulator

Save your transmitter in a file whose name includes the letters "FMTx" and your initials (e.g., *FMTx_BAB.gvi*).

Receiver

2. A template for the receiver has been provided in the file *Lab6RxTemplate.gvi*. This template contains the six interface functions along with a waveform graph on which to display your demodulated output signal.

Complete the program to demodulate the complex array returned by *Fetch Rx Data* and display the result. Figure 3 shows an implementation of the FM demodulator, including extracting the angle, unwrapping, differentiation, and lowpass filtering.

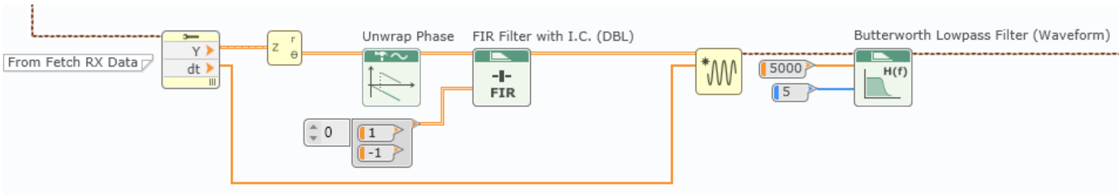

Figure 3. FM Demodulator

Save your receiver in a file whose name includes the letters "FMRx" and your initials (e.g., *FMRx_BAB.gvi*).

Questions

1. Find the frequency response of the integrator given by Eq. (7). Compare with the frequency response of an ideal integrator. Is the discrete-time integrator more like an ideal integrator when the frequency of the input is low or when it is high?

2. Find the frequency response of the differentiator given by Eq. (9). Compare with the frequency response of an ideal differentiator. Is the discrete-time differentiator more like an ideal differentiator when the frequency of the input is low or when it is high?

6.4 Lab Procedure

1. Connect a loopback cable and attenuator between the TX 1 and RX 2 connectors. Connect the USRP to your computer and plug in the power to the USRP. Run LabVIEW and open the transmitter and receiver that you created in the prelab.

   Note that all graphs in this lab are taken on a linear scale (dB on = False).

2. Ensure that the transmitter is set up to use

   Carrier Frequency: 915 MHz

   IQ Rate: Not critical; 1 MHz

   Gain: Not critical. 0 dB

   Active Antenna: TX1

   Message Length: 200,000 samples gives a good block of data to work with.

   Peak Frequency Deviation: 30 kHz seems a good value to start with.

   Start Frequency, Delta Frequency, Number of Tones: Not critical, but keep the highest frequency below 5 kHz. Three tones seems to work well, but you may wish to start with a single tone to verify operation.

3. Ensure that the receiver is set up to use

   Carrier Frequency: 915 MHz

   IQ Rate: 1 MHz

   Gain: Not critical. 0 dB

   Active Antenna: RX2

   Number of Samples: Same value as the transmitted message length.

   Run the transmitter, then run the receiver. After a few seconds, stop the receiver using the STOP button, then stop the transmitter (using the STOP button). Use the horizontal zoom feature on the graph palette to expand the "message" waveform in the transmitter and the "demodulated output" waveform in the receiver. Both waveforms should be identical, except for scaling.

4. The bandwidth of an FM signal is notoriously difficult to calculate analytically. J.R. Carson, writing in the 1920's, provided a rule of thumb for approximating the bandwidth:

$$B_{FM} \cong 2\left(\Delta f + B\right), \tag{10}$$

   where B_{FM} is the bandwidth of the FM signal, Δf is the peak frequency deviation of the signal, and B is the message bandwidth.

Add an *FFT Power Spectrum and PSD* (Analysis>>Signal Processing>>Measurement) to your transmitter. Obtain the "signal" input from the complex baseband waveform that is being sent to the *USRP Write Tx Data*. Connect the "Power Spectrum/PSD" output to a waveform graph. Change the label on the horizontal axis of the graph to "Frequency." The spectrum you obtain will be identical to the power spectrum of the actual transmitted FM signal, except that the carrier will appear at zero hertz, with the lower sideband on the negative-frequency side and the upper sideband on the positive-frequency side.

To obtain the classic textbook FM spectrum, set the message for a single tone at 1 kHz. Run the transmitter and obtain power spectra of the transmitted signal for peak frequency deviations of 1 kHz, 5 kHz, and 30 kHz. Take a screenshot of the power spectrum for each case. Be sure to scale the horizontal axis so that each spectrum is visible. Annotate your spectra to show the Carson's rule bandwidth, Eq. (10), for each case.

For a more realistic set of FM spectra, set the message for three tones at 1 kHz, 2 kHz, and 3 kHz. Run the transmitter and obtain power spectra of the transmitted signal for peak frequency deviations of 1 kHz, 5 kHz, and 30 kHz. Take another set of screenshots of the power spectrum for each case. Be sure to scale the horizontal axis so that each spectrum is visible. Annotate your spectra to show the Carson's rule bandwidth for each case.

5. One of the more curious, but also very useful, phenomena associated with FM is the so-called "capture effect." To observe this effect you will need to modify your transmitter to simultaneously generate two FM baseband signals at different carrier levels.

In your transmitter, duplicate the *Basic Multitone* to generate a second message signal. Your two *Basic Multitone* functions can share "sampling info," but each should have its own set of start frequency, stop frequency, and #tones. The constants "T" and "1" can be shared as well.

Inside the transmitter's *while* loop, create a second modulator, including a second integrator. Create two front panel controls, "Carrier 1" and "Carrier 2" to set the carrier levels A_c for each modulator. For simplicity, the two modulators can share a common peak frequency deviation. Add the two FM baseband signals produced by your modulators together and send the sum to the *Build Waveform function* that feeds the *USRP Write Tx Data*.

Set up your first message generator for three tones at 1 kHz, 2 kHz and 3 kHz. Set up the second message generator for three tones at 100 Hz, 200 Hz, and 300 Hz. These two message signals should be easy to distinguish at the receiver. Set the peak frequency deviation to 30,000 Hz.

At this point test your system by setting carrier 1 to 0.9 and carrier 2 to zero. Run the transmitter and receiver and make sure that the receiver output matches message 1. Next, set carrier 1 to zero and carrier 2 to 0.9 and make sure that the receiver output matches message 2.

At the receiver, add an *FFT Power Spectrum and PSD* to view the spectrum of the baseband output. Label the horizontal axis "Frequency" and set the range to show 0 to 5000 Hz. It will be easier to distinguish message 1 from message 2 in the frequency domain than in the time domain.

Now you are ready to observe the capture effect! Start by setting carrier 1 to 0.4 and carrier 2 to 0.6. Run the transmitter and receiver. Take a screenshot of the receiver's baseband output spectrum. Repeat with carrier 1 set to 0.5 and carrier 2 set to 0.5. Repeat a third time with carrier 1 set to 0.6 and carrier 2 set to 0.4. You should find that in the first and third cases, the receiver demodulates (captures) only the stronger signal. This is the capture effect: If two FM signals are received at the same carrier frequency, the receiver will demodulate the stronger signal, even if the stronger carrier is only slightly stronger than the weaker one.

6.5 Report

Prelab

Hand in documentation for your transmitter and receiver programs. Also include documentation for any additional functions you may have created. To obtain documentation, print out legible screenshots of the front panel and block diagram.

Submit your answers to the Questions at the end of the Prelab section.

Lab

Submit the functions you created to implement the FM transmitter and receiver. Also submit any additional functions you may have created. Be sure your files adhere to the naming convention described in the instructions above. Resubmit documentation for any functions you modified during the lab.

Submit the graphs required in Step 4. Be sure to indicate on each graph the bandwidth estimated by Carson's rule. With reference particularly to the spectra for the three-tone message, does an FM signal always have sidebands that are symmetrical with respect to the carrier?

Submit the graphs required in Step 5. Explain briefly how this sequence of graphs demonstrates the capture effect.