Chapter 3

Quadrature Amplitude Modulation & Coherent AM Detection

Chapter 2 explored large-carrier AM signalling. Transmit energy was diverted to the carrier tone, resulting in a nonnegative envelope and allowing simple noncoherent demodulation of the message signal using an envelope detector. Further, the double-sideband signal created redundancy in the AM spectrum. That is, the upper and lower sidebands were conjugate symmetric, and thus the same message information was embedded in either sideband.

In this chapter, quadrature amplitude modulation and demodulation are introduced as a means of removing unwanted redundancy in the AM spectrum. Complex baseband is presented as a convenient mathematical notation for representing QAM signals. A feedback loop is presented for adaptively recovering the carrier phase and frequency for coherent demodulation of double-sideband suppressed-carrier AM signals.

3.1 Background

3.1.1 Quadrature amplitude modulation

In quadrature amplitude modulation (QAM) two real-valued messages are modulated simultaneously, resulting in a bandpass spectrum that is not necessarily symmetric about the carrier frequency, f_c. The modulation exploits the orthogonality of the sine and cosine functions

$$\int_{-1/(2f_c)}^{1/(2f_c)} \sin(2\pi f_c t)\cos(2\pi f_c t)dt = 0. \tag{3.1}$$

The quadrature modulator is shown in Figure 3.1. The modulator has two "branches," one for an *in-phase* signal, $m_\mathrm{I}(t)$, that modulates the cosine carrier and a second for a *quadrature* signal, $m_\mathrm{Q}(t)$, that modulates the sine carrier. The sine wave is 90 degrees ($2\pi/4$ radians) phase delayed from the cosine, giving rise to the name "quadrature." The two modulated waveforms, sometimes called "I" and "Q" for short, are combined to produce the modulated waveform, $s(t)$, given by

$$s(t) = m_\mathrm{I}(t)\cos(2\pi f_c t) - m_\mathrm{Q}(t)\sin(2\pi f_c t). \tag{3.2}$$

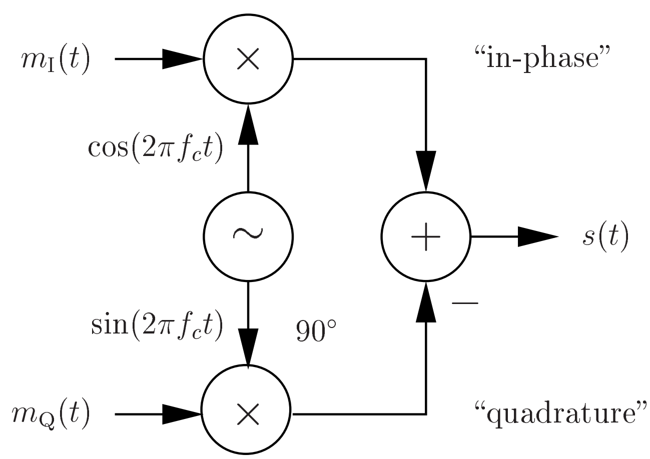


Figure 3.1: Quadrature amplitude modulator.

Ideal QAM demodulation is accomplished as shown in Figure 3.2. For baseband signals $\{m_\mathrm{I}(t),\, m_\mathrm{Q}(t)\}$ with one-sided bandwidth W Hz, the lowpass filter in the demodulator has passband edge frequency $B_p \geq W$ Hz and stopband edge frequency $B_s \leq (2f_c - W)$ Hz.

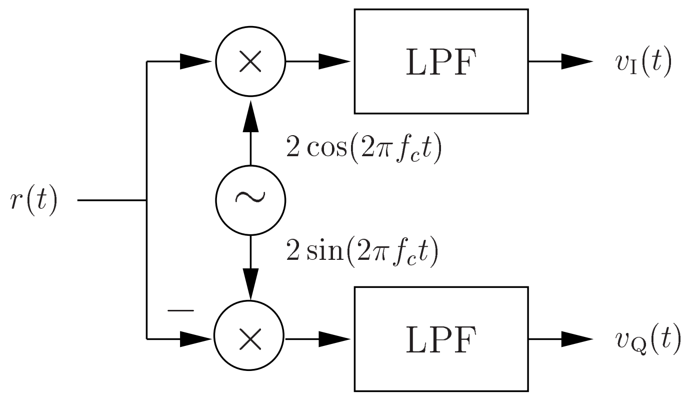


Figure 3.2: Quadrature amplitude demodulator.

Simple trigonometric identities can be used to verify that, for $r(t) = s(t)$ (i.e, a noiseless and distortionless channel), the message signals are perfectly recovered.

$$
\begin{aligned}
v_{\mathrm{I}}(t) &= \mathrm{LPF}\{r(t)\cdot 2\cos(2\pi f_c t)\} \\
&= \mathrm{LPF}\Big\{m_{\mathrm{I}}(t)\underbrace{2\cos^2(2\pi f_c t)}_{1+\cos(4\pi f_c t)} - m_{\mathrm{Q}}(t)\underbrace{2\sin(2\pi f_c t)\cos(2\pi f_c t)}_{\sin(4\pi f_c t)}\Big\} \\
&= m_I(t) \qquad (3.3) \\
v_{\mathrm{Q}}(t) &= \mathrm{LPF}\{-r(t)\cdot 2\sin(2\pi f_c t)\} \\
&= \mathrm{LPF}\Big\{-m_{\mathrm{I}}(t)\underbrace{2\cos(2\pi f_c t)\sin(2\pi f_c t)}_{\sin(4\pi f_c t)} + m_{\mathrm{Q}}(t)\underbrace{2\sin^2(2\pi f_c t)}_{1-\cos(4\pi f_c t)}\Big\} \\
&= m_Q(t). \qquad (3.4)
\end{aligned}
$$

Lowpass filtering removes the *double frequency* terms $\cos(4\pi f_c t)$ and $\sin(4\pi f_c t)$. For the coherent receiver described by Equations 3.3 and 3.4, we require that the receiver oscillator is perfectly synchronized to the transmitter oscillator: that is, the frequency f_c and phase of the sine and cosine terms at the demodulator exactly match the frequency and phase at the modulator. When the oscillators are not synchronized, one gets coupling at the receiver between the I & Q components, as well as attenuation of each.

An advantage of digital upconversion and digital downconversion versus analog mixers is the ability to closely match the in-phase and quadrature branches in amplitude, frequency, and 90 degree phase offset.

3.1.2 Complex baseband representation

The use of a complex baseband signal representation provides simplification in writing, analyzing, and programming quadrature amplitude modulation waveforms. In the complex baseband form, the in-phase and quadrature message signals are conveniently packaged together as the real and imaginery components of a single, complex-valued signal,

$$\begin{aligned} \tilde{m}(t) &= m_{\mathrm{I}}(t) + j m_{\mathrm{Q}}(t), \\ \tilde{v}(t) &= v_{\mathrm{I}}(t) + j v_{\mathrm{Q}}(t). \end{aligned} \tag{3.5}$$

Employing the complex baseband notation, the modulation and demodulation are very simply described, as seen in Figure 3.3.

Euler's identity is invoked to verify the complex baseband model shown in Figure 3.3. For the modulator, observe

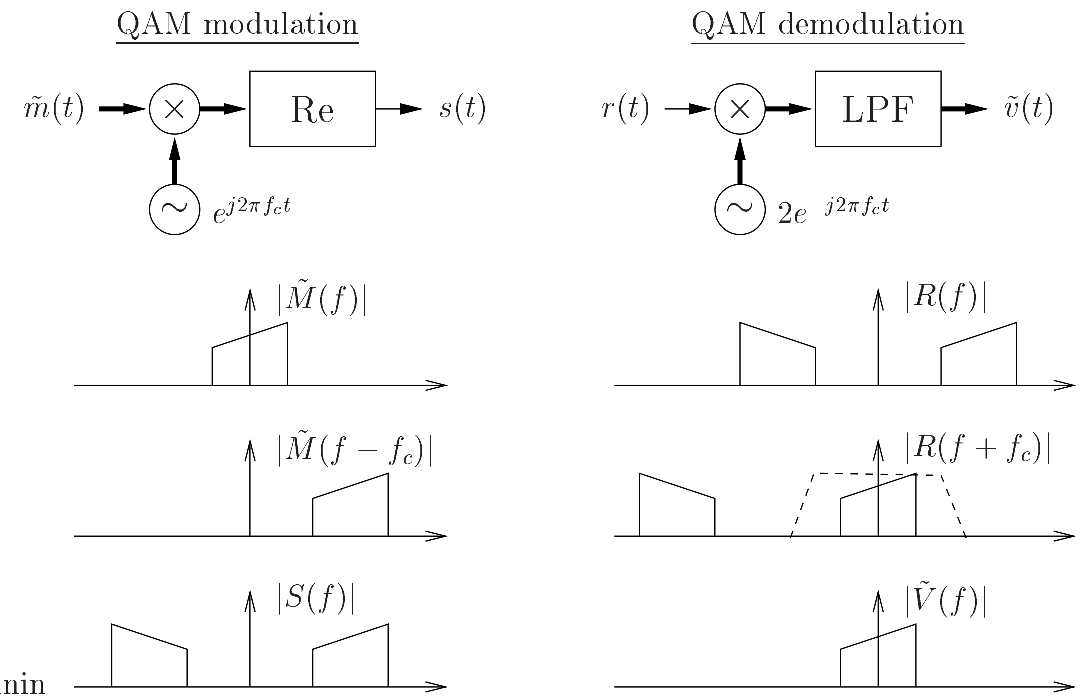
$$\mathrm{Re}\{\tilde{m}(t) e^{j2\pi f_c t}\} \tag{3.6}$$

$$\begin{aligned} &= \mathrm{Re}\Big\{\Big(m_{\mathrm{I}}(t) + j m_{\mathrm{Q}}(t)\Big)\Big(\cos(2\pi f_c t) + j\sin(2\pi f_c t)\Big)\Big\} \\ &= m_{\mathrm{I}}(t)\cos(2\pi f_c t) - m_{\mathrm{Q}}(t)\sin(2\pi f_c t) \;=\; s(t). \end{aligned} \tag{3.7}$$

The demodulation structure is verified in a similar manner:

$$\begin{aligned} \tilde{v}(t) &= \mathrm{LPF}\{s(t)\cdot 2e^{-j2\pi f_c t}\} \\ &= \mathrm{LPF}\{\Big(m_{\mathrm{I}}(t)\cos(2\pi f_c t) - m_{\mathrm{Q}}(t)\sin(2\pi f_c t)\Big)\cdot 2e^{-j2\pi f_c t}\} \\ &= \mathrm{LPF}\Big\{m_{\mathrm{I}}(t)\Big(e^{j2\pi f_c t} + e^{-j2\pi f_c t}\Big)e^{-j2\pi f_c t} \\ &\qquad - m_{\mathrm{Q}}(t)\Big(je^{-j2\pi f_c t} - je^{j2\pi f_c t}\Big)e^{-j2\pi f_c t}\Big\} \\ &= \mathrm{LPF}\Big\{m_{\mathrm{I}}(t)\Big(1 + e^{-j4\pi f_c t}\Big) - m_{\mathrm{Q}}(t)\Big(je^{-j4\pi f_c t} - j\Big)\Big\} \\ &= m_{\mathrm{I}}(t) + j m_{\mathrm{Q}}(t). \end{aligned} \tag{3.8}$$

A frequency domain interpretation of each step in the modulation and demodulation is given by the frequency spectra in Figure 3.3. Observe the asymmetry of the passband signals about the carrier, f_c, and the corresponding asymmetry of the baseband signal. Thus, the upper and lower sidebands



inin

Figure 3.3: Quadrature amplitude modulation and demodulation are simplified using a complex baseband notation.

are not redundant, in contrast to AM. The Re operation at the modulator can be interpreted using the Fourier transform relation

$$\mathrm{Re}\{u(t)\} = \frac{1}{2}\Big[u(t) + u^*(t)\Big] \overset{\mathcal{F}}{\longleftrightarrow} \frac{1}{2}\Big[U(f) + U^*(-f)\Big]. \tag{3.9}$$

The convenience of complex-baseband results in widespread use of complex-valued signal notation for passband signals encountered in communciations, radar, and medical imaging, for example.

3.1.3 DSB-SC AM demodulation: Costas loop

A double-sideband, suppressed-carrier AM (DSBSC-AM) signal is obtained as a special case of quadrature amplitude modulation: $m_{\mathrm{I}}(t) = m(t)$ and $m_{\mathrm{Q}}(t) = 0$. In the large-carrier AM signalling explored in Chapter 2, the carrier tone, $\cos(2\pi f_c t)$, held no information about the message signal, but consumed 67% of the transmit energy (in the critically modulated case). The benefit of devoting energy to the carrier was the simplicity of the noncoherent envelope detector. With DSBSC-AM, no energy is used for a carrier tone, and the price is the increased complexity of a *coherent* receiver. For coherent demodulation, the local oscillator must have frequency and phase synchronized with the transmitter. To adaptively achieve synchronization, a *Costas loop* uses a quadrature demodulator and the knowledge that $m_{\mathrm{Q}}(t) = 0$. A conceptual view of the operation of a Costas loop is shown in Figure 3.4. When the local oscillator has achieved phase synchronization with the transmitter, then the quadrature signal $v_{\mathrm{Q}}(t) = m_{\mathrm{Q}}(t)$ is zero; thus, the feedback control attempts to drive the quadrature output, $v_{\mathrm{Q}}(t)$, to zero. Note that controlling for $v_{\mathrm{Q}}(t) = m_{\mathrm{Q}}(t) = 0$ leaves a 180 degree phase ambiguity for the entire signal; this ambiguity may be resolved using a pilot symbol or differential encoding.

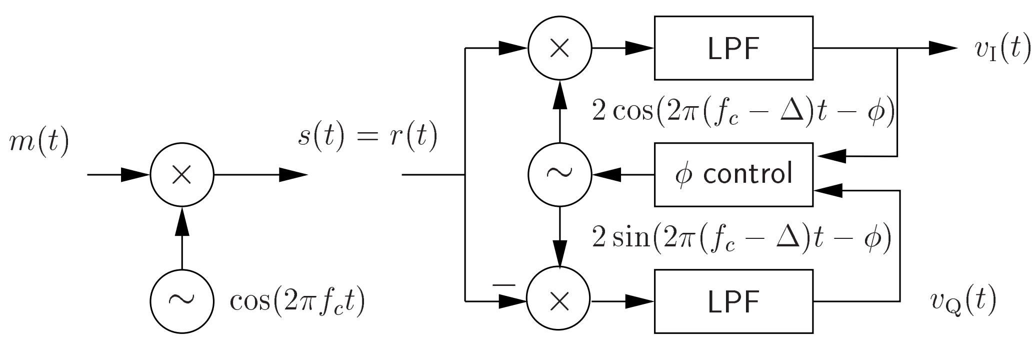


Figure 3.4: DSBSC-AM modulator (left) and Costas loop for coherent demodulation (right).

At the receiver, consider a frequency mismatch Δ Hz and a phase mismatch ϕ radians, as shown. Using the convenient complex baseband representation of Figure 3.3 we have

$$\tilde{v}(t) = \tilde{m}(t)e^{j(2\pi\Delta t+\phi)}. \tag{3.10}$$

Thus, if there is a frequency offset between transmitter and receiver or a Doppler shift of the transmit frequency, then the phase error, $(2\pi\Delta t + \phi)$, at the receiver is a linear function of time, with slope proportional to the frequency offset. Figure 3.5 gives a graph of this time-varying phase offset, which is due to the phase mismatch (intercept) and frequency mismatch (slope/2π).

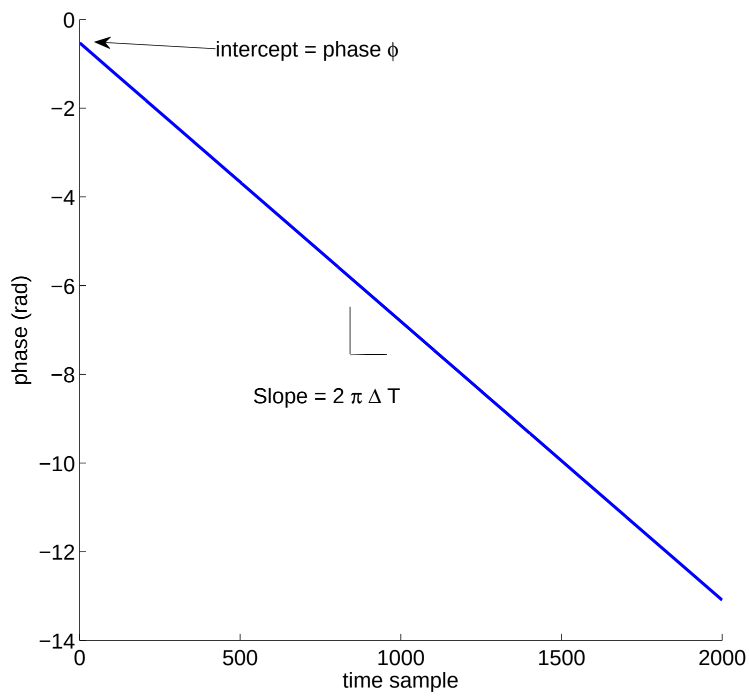


Figure 3.5: A time-varying phase error appears at the receiver output due to frequency and phase offsets in the receive oscillator.

Memoryless phase recovery for DSBSC-AM

For digital phase recovery, initially consider a memoryless nonlinear processing approach. For sampling period T_s, the complex baseband sampled-data signal produced by the digital downconverter (see Section 1.3.5) is denoted $\tilde{v}[n]$

$$\tilde{v}[n] = m[n]e^{j(2\pi\Delta nT_s+\phi)}. \tag{3.11}$$

Note that for DSBSC-AM, $m_Q(t) = 0$ implies that the ideal received signal is purely real-valued. So, the real-valued message at each sample is either

$|\tilde{v}[n]|$ or $-|\tilde{v}[n]|$. For phase recovery, we define

$$\hat{m}[n] = |\tilde{v}[n]| \operatorname{sgn}\{\theta\} \tag{3.12}$$

where $\tilde{v}[n] = |\tilde{v}[n]|e^{j\theta}$. This approach can recover not only the phase offset (up to an unknown 180 degree ambiguity) but also a frequency offset, provided $|\Delta| < \frac{1}{2T_s}$.

As is seen below and in the exercises, errors in frequency and phase recovery may be smoothed via filtering. A second-order feedback loop is described next; and, for block processing, a least-squares line fit is introduced in the exercises.

Second order feedback loop

As seen in Equation 3.11 the combination of a frequency and phase offset results in a phase error that grows linearly with time. From automatic control theory, a second-order loop can track a linear (ramp) function with zero steady-state error, whereas a first-order loop cannot. Figure 3.6 gives a second-order Costas loop well suited for digital implementation (Tretter, 2008).

The parallel solid and dashed lines represent the real and imaginary parts, respectively, of complex baseband signals. The system generates an estimate, $\hat{\theta}[n]$, of the phase error. The phase error estimate is used to create $e^{-j\hat{\theta}[n]}$, which is multiplied to the DDC output, $\tilde{v}[n]$, to produce

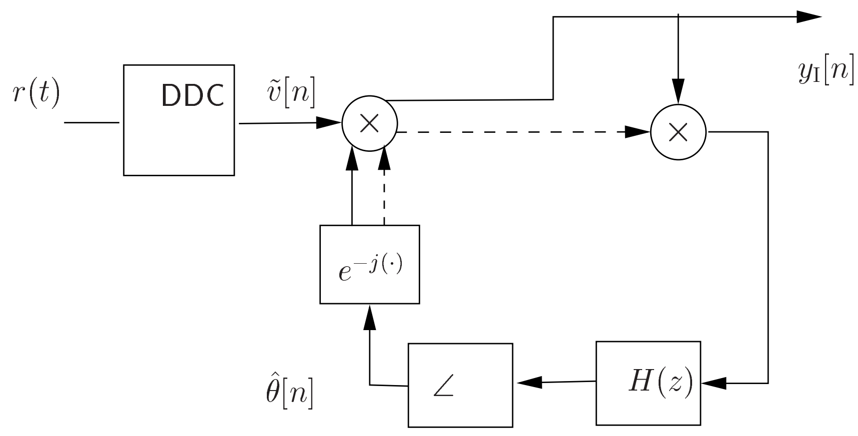


Figure 3.6: Second-order feedback loop for coherent demodulation of DSBSC-AM with phase offset.

---

the demodulated output, $\tilde{y}[n] = y_\mathrm{I}[n] + jy_\mathrm{Q}[n]$. When the estimated phase matches the phase errors present in $\tilde{v}[n]$, the loop is said to be "in lock." In this case, $y_\mathrm{I}[n]$ reproduces the DSBSC-AM message, $m_\mathrm{I}[n]$, and the imaginary part, $y_\mathrm{Q}[n]$, becomes zero. The loop filter has discrete-time transfer function

$$H(z) = \frac{(\alpha + \beta)z^{-1} - \alpha z^{-2}}{1 - 2z^{-1} + z^{-2}} \tag{3.13}$$

where α and β are small positive constant parameters for adjusting loop bandwidth. The value of β should be small relative to α. (Example values are $\alpha = 0.01$ and $\beta = 0.0002$.)

3.2 Experiment: QAM signals

In this exercise, you will explore quadrature amplitude modulation and complex baseband representations. From the directory `C:/508 templates` on the benchtop PC, copy template LabVIEW Virtual Instruments `Rx Streaming.vi` and `Tx Streaming.vi` to your workspace for use in the exercise below.

1. In the `Matlab Script Node` of `Tx Streaming.vi`, create a QAM signal using complex baseband notation. Delete any code that puts energy into the IF carrier. Scale the final signal, `Tx`, to avoid any clipping at the digital-to-analog conversion. In MATLAB, use

```
>> f=20e3;
>> mI = cos(2*pi*f*[1:buffer]/ActualIQrate);% in-phase
>> mQ = sin(2*pi*f*[1:buffer]/ActualIQrate);% quadrature
>> Tx = mI + j*mQ;% complex baseband message
>> Tx = Tx/max(abs(Tx))*0.95;%scale to avoid clipping
```

2. On the NI-PCI 5640R board, connect output AO-0 to input AI-0 using a one-foot cable with SMA connectors.

3. Upconvert your `Tx` waveform to an intermediate frequency of your choosing (say, $f_{c:Tx} = 10\,\text{MHz}$). Using the suggested steps below, downconvert the received waveform to view spectra and the in-phase time waveforms at the receiver.

   (a) Vary the downconversion frequency, $f_{c:Rx}$, starting with $f_{c:Rx} = f_{c:Tx}$ and using an offset $\Delta \leq 100\,\text{Hz}$. Observe, record, describe, and explain the results at the receiver. (Question 3.1)

   (b) At the transmitter, vary the frequency, f, of the transmit tone and the `IQRate` at the receiver to obtain aliasing of received signal, $\tilde{v}[n]$. Report your settings for f and `IQRate`. Observe, record, describe, and explain the results at the receiver. (Question 3.2) (Question 3.3)

4. Modify your `Matlab Script Node` of `Tx Streaming.vi` to implement a DSBSC-AM signal. Receive the signal and view the spectrum. Observe, record, describe, and explain the results at the receiver.

3.3 Experiment: Phase offset

In this exercise, you will observe the effect of a phase offset at the receiver. Note that the magnitude spectra above did not display the phase information when transmitting and receiving the signal $\tilde{m}(t) = e^{j2\pi ft}$.

1. Modify the `Matlab Script Node` of `Tx Streaming.vi` to create a QAM signal using complex baseband notation. Scale the final signal, `Tx`, to avoid any clipping at the digital-to-analog conversion. Your baseband signal, before scaling, should be

$$\begin{aligned} m_{\mathrm{I}}(t) &= 1 \\ m_{\mathrm{Q}}(t) &= 0. \end{aligned} \tag{3.14}$$

2. Modify the `Rx Streaming.vi` Block Diagram to create an additional display to show the quadrature component of the received baseband signal, $v_{\mathrm{Q}}[n]$.

3. Set f and $f_{c:Rx} = f_{c:Tx}$ to values of your choosing.

4. With the transmit VI running, start then stop execution of the receive VI to inspect the time waveforms and phase plot. The phase plot, in MATLAB, is `plot(phase(vI + j*vQ)*180/pi);`; repeat several times and record the phase offset at the receiver. Observe, record, describe, and explain the results at the receiver. Specifically, note the relative peak-to-peak amplitudes and DC offsets of $v_{\mathrm{I}}[n]$ and $v_{\mathrm{Q}}[n]$. Question 3.4 Question 3.5

3.4 Exercise: Memoryless phase recovery for DSBSC-AM

In this exercise, you will implement a simple memoryless procedure for coherent demodulation of DSBSC-AM signals.

1. Modify your `Rx Streaming.vi` template to implement a memoryless phase recovery algorithm of Equation 3.12 by using the following steps.

   (a) Compute the magnitude and phase of the digital downconverter's output using `Numeric >> Complex >> Polar to Complex`.

   (b) Generate $\operatorname{sgn}\theta$ using the `sign` VI.

   (c) Use `Numeric >> Multiply` to compute the product, $|\tilde{v}[n]|\operatorname{sgn}\{\theta\}$.

2. Set $f_{c:Rx} = f_{c:Tx}$. Observe the real and imaginary parts of the reconstructed signal. Do you see that the imaginary part is close to zero for a real signal? Why or why not? Question 3.6

3. Should the approach work for any DSBSC-AM signal ($m_{\mathrm{Q}}(t) = 0$)? Why or why not? Question 3.7

4. Adjust the center frequency at the receiver to set Set $f_{c:Rx} \neq f_{c:Tx}$. Try several values, in small steps. Include two or more frequency offsets satisfying $|\Delta| < 100\,\mathrm{Hz}$. Observe, record, describe, and explain the results at the receiver. Explain your observations, for example, by reference to Equations 3.11 and 3.12. Question 3.8

5. By induction, show for a phase offset ϕ, this approach reconstructs $m[n]$ up to an 180° ambiguity (that is, the output sequence is either $m[n]$ or $-m[n]$). Question 3.9

6. For what range of values of frequency offset, Δ, can this approach recover $m[n]$ (up to a 180° ambiguity)? Question 3.10

3.5 Exercise: Closer look at the phase offset

In this exercise, you will compute and display the phase of the received baseband signal to observe the effect of a phase and frequency offset at the receiver.

1. Use the modifications of `Rx Streaming.vi` Block Diagram to create an additional display showing the phase of the received signal. The phase will be computed using a signed magnitude representation.

   (a) From the real and imaginary parts of the received signal, find the magnitude and phase by using the `Re/Im to Polar` VI found at `Numeric >> Complex`.

   (b) Phase unwrapping expresses phase versus time as a continuous function, without the π or 2π jumps associated with graphing phase only on the interval $[-\pi, \pi)$. To unwrap the phase, first multiply $\bar{\theta}$ by 2, then use the `Unwrap Phase` vi to peform phase unwrapping. (The multiplication merely increases the difference between phases of consecutive samples, thereby facilitating the unwrapping.) Finally, divide the output phase by 2, to undo the scaling done prior to unwrapping. Plot the unwrapped phase. Question 3.11

   (c) Let $\theta \in [-\pi, pi)$ be the intercept of the phase plot phase from step (b) (or just the value, if the phase plot is constant). To obtain a phase $0 \leq y < \pi$ for a signed magnitude represenation, implement we need only add pi if $\theta < 0$; this can be accomplished by

$$\bar{\theta} = \theta + \text{sign}\{\theta\} \cdot \left(\frac{-\pi}{2}\right) + \frac{\pi}{2}$$

2. In the `Matlab Script Node`, create a baseband message

```
>> mI = ones(1,5*buffer);% in-phase
>> mQ = zeros(1,5*buffer);% quadrature
>> Tx = mI + j*mQ;% complex baseband message
>> Tx = Tx/max(abs(Tx))*0.95;% scale to avoid clipping
```

3. Set f and $f_{c:Rx} = f_{c:Tx}$ to values of your choosing.

4. With the transmit VI running, start then stop execution of the receive VI to inspect the time waveforms and phase plot; repeat several times and record the phase offset at the receiver. Observe, record, describe, and explain the results at the receiver; what are likely causes for the phase offset? Question 3.12

5. Repeat the previous step, but with a *frequency offset* at the digital downconverter: $f_{c:Rx} \neq f_{c:Tx}$. With the transmit VI running, again start then stop execution of the receive VI to inspect the time waveforms and phase plot; repeat several times and describe the phase offset plot seen at the receiver. Attempt several small values for frequency offset. Observe, record, describe and explain the results at the receiver. Explain the time-varying phase offset. Hint: see Equation 3.10. Question 3.13

3.6 Exercise: Frequency and phase recovery

In this exercise, you will apply a *block processing* procedure to remove a time-varying phase offset resulting from frequency and phase offsets. (The procedure is well suited for application to packet communications using digital modulation.)

1. Begin with your modifications to `Rx Streaming.vi` from Exercise 3.5. Recall that `y` (radians) denotes the unwrapped phase in that exercise.

2. Fit a line to the phase: apply `linear fit.vi` found at `Mathematics >> Fitting >> Linear fit`. For the `X` input, insert an array of 1:4000 (data block size). Make a `for` loop (`Structures >> For loop`) outside the while loop. You can see an input `N` specifying the number of iterations. Connect it to Data Block Size, `S`. You can also see an output `i`, which specifies the iteration number. Connect the iteration number to input `X` of the `linear fit.vi`. By default, indexing is enabled when you take the output `i` outside a `for loop`.

3. Use `Bundle` (`Clusters and variants >> Bundle`) to bundle together slope and intercept and plot them together on the same Time Signal plot. The time signal plot can be obtained by right clicking on front panel and using `Waveform Chart` (`Graph >> Waveform Chart`).

4. Record the values of frequency offset you obtain by changing the Rx center frequency and find a relation between the two. Hint: see Equation 3.10.

Question 3.14