ELEC321 Communication Systems

Practical Notes

Writing Practical Reports

Introduction

Even at this late stage of their studies, many students produce unsatisfactory practical reports; these notes set out some general guidelines to writing practical reports, as well as giving information which is specific to ELEC321.

What are reports for?

Practical reports can have a variety of aims and audiences. For example, you may be required to test a particular piece of commercial equipment and report on its suitability to replace current equipment. You may be required to measure the properties of a given circuit and check whether they meet the manufacturer's specifications.
Even these two examples illustrate two important things to guide your report: what is its purpose and who will read it?

Aim of ELEC321 reports

For ELEC321 the answers are clear: Your lecturer will read the report, which should demonstrate that you understand the relevant theory, can apply it to real circuits, have made measurements as accurately as your equipment permits and have checked that, within these limits, the behaviour of the circuit agrees with the theory. If it does not, then you should find out why – which of the above aspects have you got wrong? Indeed, a convincing treatment of such discrepancies can make the difference between an excellent report and an average one.

Have you got it right?

Before you claim that the circuit behaviour does agree with the theory, make sure that you can (and do) support this claim. It is not enough to write:

Predicted value = 22.7
Measured value = 24.319 .

These values are not equal, and it is no argument to say that they are nearly equal – how close is close enough? (No more is it valid to claim that the two values are not equal, so that the theory is wrong.)

In all cases it is necessary to consider the uncertainties in your measurements and predictions, and to show that the range of uncertainty of the measured value covers the predicted one. Calculation of uncertainties need not always be a big deal; if the predicted value depends on the values of a few 5% resistors and the predicted and measured values differ by 3%, little more need be said.

Of course, agreement of this kind is not the end of the story. If more accurate measurements could be made it could well be that minor effects, which were concealed by the uncertainties, become apparent. It is then necessary to add to your model of the circuit. In designing your system we will often take some care to choose parameters which ensure that such second-order effects are negligible, while at other times we will allow them to make a significant difference. (The first alternative is most appropriate for introductory work, the second for more advanced studies.)

What to include

Because of the nature of your work in ELEC321, your report need not be highly structured. (A report intended to convince a managing director that new equipment was needed for a production line would have a very different structure.) However, too many students go to the opposite extreme; they may give tables of values without making it clear whether they are predicted or measured ones, perhaps not even revealing to what measurements they apply.

It is a waste of effort to copy slabs of prose from the practical notes to your report, but you should include enough detail to make the report intelligible without constant reference back to the notes. Identify what measurements are being made in plain English and not just by section number; explain concisely how the measurements were obtained, referring to the practical notes if appropriate but giving full details if you had to depart from the methods in the notes. Explain how you analysed these results in a similar style; don't just have stacks of raw equations. Present the results of your measurements clearly, and in such a way as to make the comparison with the predicted values easy to follow.

How to write it

We do not expect your report to be a major literary feat but it should be in good basic English. Write complete sentences, not disconnected snippets. Try to avoid spelling mistakes; a report written to the standard of a ten-year-old does not inspire confidence.

In published papers it is necessary to spell out the exact experimental procedure to strangers, so a high degree of detail and structure is appropriate. However for ELEC321 the equipment is quite basic and known to the reader, so that only unusual methods need be reported fully. The occasional heading to indicate a new section is appropriate.

Observe high standards in any graphs, with measured points and axes clearly labelled; label the graph itself to make clear what measurements it plots. Write legibly in ink, don't scrawl in pencil. Because your report is handed in as you leave the session there is no opportunity to type the report; if your writing is too terrible perhaps you could use a style nearer printing? Write on A4 paper, stapled – no need for books or folders.

References

Bruce M. Cooper:	*Writing Technical Reports*
Pam Peters:	*Strategies for Student Writers*
C. Turk & J. Kirkman:	*Effective Writing.*

Some Practical Notes

Practical Session 1

TIME AND FREQUENCY DOMAINS

1. Procedure

The emphasis in this introductory session is not on taking lots of measurements and then comparing them with the proper theory. Rather you will make relatively few measurements, but are required to discuss their interpretation from several viewpoints. The aim is to emphasize the way their behaviour in the frequency and time domains is related.

You will need to familiarise yourself with the Tektronix 2247A oscilloscope. It is not difficult to use, and in fact it makes some measurements, such as dc or peak-peak volts, and frequency or even phase, both more simply and more accurately than a more basic CRO; just follow the menu buttons. Use the normal timebase A rather than the delayed timebase B for your measurements. AUTO SETUP is a magic button.

2. Low-pass Circuit: Frequency Response

i) Set up a low-pass circuit using

$R = 22\ \text{k}\Omega \qquad C = 47\ \text{nF}$.

Calculate its 3dB frequency f_o .

ii) Take enough measurements to allow you to check this value, and to sketch the amplitude and phase response for a decade or so either side of f_o .

iii) Up to what frequency is the amplitude response constant to within 10% ?

iv) Up to what frequency is the phase response constant to within 10% of a radian? (Note how different are the results of these – apparently similar – criteria.)

3. Low-pass Circuit: Time Response

i) Apply a 50Hz square wave to the circuit and record the output waveform.

In the time domain we may solve the differential equation involved, namely

$$C\ dv_o / dt \ = \ (v_i \ - \ v_o) / R \qquad (= \ i) .$$

If the input v_i swings from –V to +V , the output v_o may be shown to be given by (check for $t = 0$ and $t = \infty$)

$$v_o \ = \ V \ - \ 2\, V \exp (-t / \tau)$$

where

$$\tau \ = \ R\, C$$

is called the time constant.

The output takes a couple of time constants to rise or fall. This is related to the time needed to charge (or discharge) C through R.

ii) Discuss the frequency response in similar terms to these. (E.g., for a given input voltage, what is the maximum rate of change of output voltage?)

iii) Discuss the observed output waveform in terms of the effect of the circuit's frequency response on the signal's Fourier components.

iv) Up to how high a frequency does the output roughly resemble the input square wave, in that it has recognizable rising and falling edges and is almost full size? (There is no unique answer here.)

Discuss how this is related to the circuit's frequency response.

v) Your oscilloscope's vertical amplifier is said to be good up to 100 MHz (i.e. frequency domain). Estimate its rise time for a square-wave input (i.e. time domain) .

vi) Explain how a single measurement on a circuit using a square wave at a frequency considerably less than the circuit's 3dB frequency f_o can give an estimate of f_o. (Don't just give a formula, but explain the apparent paradox that the input frequency can be well within the pass band.)

4. High-pass Circuit: Time Response

i) Set up a high-pass circuit using

$C = 47$ nF $\qquad R = 220$ kΩ .

Apply a square-wave input at 100 Hz and record the output waveform, using a × 10 probe to avoid loading effects.

ii) Calculate the 3dB frequency of the circuit.

iii) Calculate the effect of the circuit on the fundamental, and on the first few harmonics, of the input (i.e. frequency domain) .
Explain how these effects are linked to the shape of the output (i.e. time domain) .
(Warning: There is an apparent paradox here, so give a careful explanation.)

5. Band-pass Circuit: Frequency Response

i) Set up a tuned circuit consisting of the parallel combination of

$L = 10$ mH $\qquad C = 0.1$ μF .

Apply a sinusoidal signal v_i to it through a 10kΩ resistor and observe the output v_o across the tuned circuit. Theory says that the maximum output is at a frequency ω_o given by

$$L\,\omega_o = 1 / C\,\omega_o$$

and that the output drops by 3 dB at frequencies given approximately by

$$\omega_o \times (1 \pm 1 / 2\,Q)$$

where

$$Q = R / L\,\omega_o \quad .$$

ii) Check these predictions by measurement. (In order to take account of losses in the tuned circuit, the value R used in these calculations needs to be reduced to be effectively about 5700 Ω .)

6. Band-pass Circuit: Time Response

i) Apply a square wave at 50 Hz to the circuit and record the output.
The parameter Q (quality factor) has two alternative definitions.

a) It is the ratio of the centre frequency f_o to the 3dB bandwidth (frequency domain) .

b) It is the ratio of the total stored energy to the average energy lost per unit angle of the oscillation (time domain) .

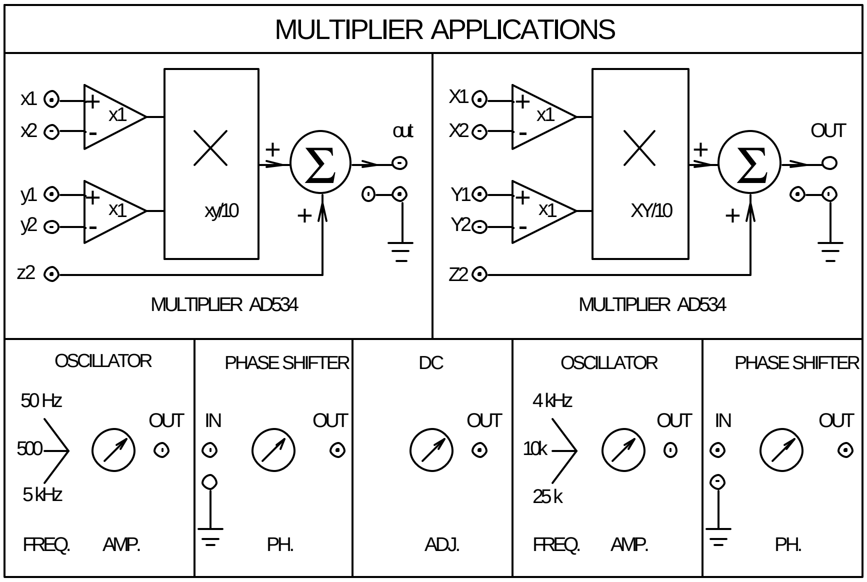
ii) Discuss how the output of the circuit is related to its properties in the frequency domain and in the time domain.

Practical Sessions 2 – 4

FOURIER SERIES AND MODULATION

1. Procedure

You will perform some of your practical work using the Multiplier Applications chassis illustrated below.



The multiplier (Analog Devices type AD534) is usable to 1 MHz, so has near-ideal performance for the frequencies in the audio range at which you will use it.
The same principles as illustrated with this multiplier apply at the normal radio frequencies, but more specialised circuitry is normally used.

The multiplier has inputs

X_1, X_2, Y_1, Y_2, Z_2 (or x_1, x_2, y_1, y_2, z_2)

which may take values in the range –10 V to +10 V.

The output is given by

$$V_0 = Z_2 + (X_1 - X_2) \times (Y_1 - Y_2) / 10$$

and must also have values in the range –10 V to +10 V to avoid distortion.

On the Multiplier Applications chassis all multiplier inputs have pull-down resistors of 22 kΩ. Except for critical measurements (e.g. in section 3) unused inputs may therefore be regarded as grounded.

The chassis has two sinusoidal oscillators and phase shifters. Each oscillator can be switched to one of three frequencies; simultaneously the associated phase shifter's response is altered so that 90 degrees is at about centre scale at the oscillator frequency, with a range of about 1–130 degrees. (The other phase shifter is not so suitable.)

Power supplies are derived from ±15V regulators in the chassis, so that external supplies of 18–20 V are required. The chassis contains a dc supply, variable from –10 V to +10 V but with a current rating of only 5 mA.

2. Important Note

Some sections require the use of an external signal source. Keep this source turned down or off or disconnected until the multiplier is powered up, and turn the source down or off or disconnect it before the multiplier is powered down. (The AD534 may be damaged if an input voltage extends beyond either power supply, so without power there must be no signal.)
Note again that the signal voltage reading on the generator is the voltage that would exist with a 50Ω load so that, with the light loads of these sessions, the reading must be doubled.

3. Initial Tests

Ideally

$$V_o = Z_2 + (X_1 - X_2) \times (Y_1 - Y_2) \,/10 \ .$$

However each term of this expression is in practice subject to errors such as

i) zero errors ('offsets');
ii) scale-factor errors (wrong gain); and
iii) non-linearity errors ('distortion').

Zero errors are normally temperature dependent, while the other errors are moderately frequency dependent. By working with large signals at moderate frequencies, such errors may often be ignored.

Check this by making (at least) the following measurements.
(Sketch a block diagram of the circuit in each case, and throughout this session.)

i) Make

$Z_2 = X_2 = Y_2 = 0 , \qquad Y_1 = +10{\cdot}0 ,$

X_1 = 20Vpp sine wave at 500 Hz.

The output should be identical to X_1 . Check this by comparing the two, e.g. by subtracting them on the CRO. (Vary Y_1 a little if necessary.)
Now vary Y_1 from +10 through to –10 volts, checking that the output varies in size.
Note the phase change around $Y_1 = 0$, and see how small you can make the output by adjusting Y_1 .
(Note particularly the frequency of this minimum output, and comment.)

ii) Perform a similar set of measurements except with X_1 and Y_1 signals swapped.

(The X and Y inputs affect the output using quite different mechanisms.)

iii) Repeat i) and ii), not so thoroughly, but with the sine wave at 25 kHz.
Look particularly for phase shifts and 'feedthrough' − output that can't be reduced with a dc offset.

Comment on what these measurements reveal about the imperfections of the multiplier. Refer back to these measurements if one of your later circuits does not work as well as it might.

4. Product of Two Sine Waves at Different Frequencies

If

$$X_1 = A \cos (w_1 t + f_1)$$
$$Y_1 = B \cos (w_2 t + f_2)$$

then their product is

$$X_1 \; Y_1 = A\,B \cos (w_1 t + f_1) \cos (w_2 t + f_2)$$

which, using a standard trigonometrical identity, becomes

$$X_1 Y_1 = (A B / 2) \{ \cos [(w_1 - w_2)t + (f_1 - f_2)] + \cos [(w_1 + w_2)t + (f_1 + f_2)] \}$$

so that, if w_1 and w_2 are not equal, the result is the sum of two sinusoidal oscillations, at two new frequencies $(w_1 - w_2)$ and $(w_1 + w_2)$ and with equal amplitudes $(A B / 2)$. This surprising result is very important in the study of communication systems, with a wide variety of interpretations / applications.

i) Make

$X_2 = Y_2 = Z_2 = 0$,

$X_1 = 10 \cos (w_1 t + f_1)$ $\qquad (f_1 = 5$ kHz)

$Y_1 = 10 \cos (w_2 t + f_2)$ $\qquad (f_2 = 4$ kHz)

and observe the multiplier output

$OUT = X_1 Y_1 / 10$.

Check that it appears to be a (slow) 1kHz sine wave with a (fast) 9kHz sine wave superimposed. (Check the precise f_1 and f_2 values using the CRO. Investigate various trigger sources for the most appropriate display; triggering off the very highest peaks of the output may give the best display, but explain why.)

ii) Note that, if either input amplitude is varied, the amplitude of the output waveform varies but its two components remain of equal size.

iii) Make

$x_2 = y_2 = 0$

$x_1 = 5 \cos (w_1 t + f_1)$ $\qquad y_1 = +10{\cdot}0$ $\qquad (f_1 = 500$ Hz)

$z_2 = 5\ \cos(w_2 t + f_2)$ $\qquad\qquad$ $(f_2 = 4\ \text{kHz})$

and observe the output

$out = x_1 + z_2$.

This waveform is the sum of two sine waves whose frequencies are in much the same ratio as those in the previous result, so that if the timebase is half as fast the two displays should be similar.
Check this.

iv) Note that, if either input amplitude is varied, the behaviour is different from that observed if two inputs are multiplied. Record and comment on the differences.

5. Product of Two Sine Waves at the One Frequency

If

$X_1 = A\cos(w\,t + f_1)$ $\qquad\qquad$ $Y_1 = B\ \cos(w\,t + f_2)$

then their product is

$X_1\,Y_1 = A\,B\ \cos(w\,t + f_1)\ \cos(w\,t + f_2)$

which, using the same trigonometrical identity as before, becomes

$X_1\,Y_1 = (A\,B\ /\ 2)\cos(f_1 - f_2)\ +\ (A\,B\ /\ 2)\cos(2w\,t + f_1 + f_2)$

so that the result is the sum of a dc component dependent on the phase difference (the first term) and a sinusoidal component at twice the input frequency.
If this signal is passed through a low-pass filter we may reject the sinusoidal component and retain the dc component.

i) Make

$X_2 = Y_2 = Z_2 = 0$

$X_1 = 10\ \cos(w\,t + f_1)$ $\qquad\qquad$ $(f = 10\ \text{kHz})$

$Y_1 = 10\ \cos(w\,t + f_2)$

and observe the multiplier output

$OUT = X_1\,Y_1\ /\ 10$.

Note that the observed output consists of a dc component, whose value varies with the phase shift between X_1 and Y_1 , and a sinusoidal component of constant amplitude.

ii) Patch up a simple R-C low-pass filter consisting of

$R = 220\ \text{k}\Omega$ $\qquad\qquad$ $C = 0{\cdot}047\ \mu\text{F}$.

(Calculate its 3dB frequency.)
Check its function on the CRO, using a ×10 probe to avoid loading the filter unduly.
Measure the dc output of the multiplier as in part i using this filter, and check and plot the dependence of this voltage on the phase difference $f_1 - f_2$.

6. Product of Two Sine Waves at Nearby Frequencies

Remember that, if sine waves at frequencies w_1 and w_2 are multiplied, the output is the sum of two sine waves, at frequencies $(w_1 - w_2)$ and $(w_1 + w_2)$. If $w_1 \approx w_2$ then it is relatively easy, with a low-pass filter, to pass only the difference frequency.

i) Make

$X_2 = Y_2 = Z_2 = 0$

$X_1 = 10 \cos(w_1 t + f_1)$ $(f_1 = 4 \text{ kHz})$

$Y_1 = 10 \cos(w_2 t + f_2)$

where Y_1 is derived from a signal generator to allow w_2 to be varied. (Read again the Important Note of section 2.)
Pass the output through the low-pass filter used in section 5 and observe the output on the CRO.
Vary f_2 about 4 kHz, carefully adjusting it to obtain the largest possible output amplitude. (Make this output only a few Hz, by precise selection of the frequency of the signal from the generator. Note that, with an input frequency of the order of 1 Hz, the CRO amplifier must be dc coupled for accurate results.)
Verify that the output amplitude is 5 volts as expected.
Check that this amplitude varies as expected with the amplitude of X_1.
Verify that $w_1 \approx w_2$ when the output is large. (Using the frequency-measurement function of the CRO (Counter-Timer), this can be done very precisely.)

7. Electronic Fourier Analysis

If a signal which contains a variety of Fourier components is multiplied by a sinusoidal signal at frequency w, the output of the multiplier contains Fourier components at the various sum and difference frequencies.
If the input frequency w is approximately equal to one component, at frequency w_n, of the original signal, the output component at frequency $(w - w_n)$ is readily separated from all the others, hence its amplitude determined. (Phase is not so simple.)
By varying w over an appropriate range it is possible to measure the amplitude of each of the Fourier components of the signal.

i) Set up the signal of section 4(iii) using one multiplier.
Use the other multiplier to multiply this by a 20Vpp sinusoidal input of variable frequency. (Remember the Important Note of section 2.)
Observe the output of this second multiplier via the low-pass filter of section 5(ii).
Verify the amplitude and frequency of the two Fourier components using the above method. (Note again that the signal generator frequency can be very precisely set to

ensure an output frequency of about 1 Hz. The peak-to-peak voltage of this signal can be accurately measured by setting the CRO cursors to its peaks and troughs.)

ii) Repeat (i), except using the signal generated in section 4(i).

iii) Now similarly check the Fourier components of a square wave.

To avoid the need for two external generators, use a 20Vpp sine wave at a fixed 25 kHz and make the input a 20Vpp square wave of variable frequency.
Verify that you only get an appreciable output from the low-pass filter if the square-wave frequency is at an odd submultiple of the sine-wave frequency.
Check that the amplitude series is as expected, going to at least the eleventh harmonic.

8. Suppressed-Carrier Modulation (DSBSC)

We continue to multiply sine waves, but interpret the results differently.
To reduce the length of equations, and because they don't really matter, we omit f_1, f_2 .

Make

$x_2 = y_2 = z_2 = 0$

$x_1 = 10 \cos w_c t$ $\quad (f_c = 4 \text{ kHz})$

$y_1 = 10 \cos w_m t$ $\quad (f_m = 50 \text{ Hz})$.

The output is

$$
\begin{aligned}
out &= 10 \cos w_c t \times \cos w_m t \\
&= 5 \cos(w_c - w_m)t + 5 \cos(w_c + w_m)t \quad .
\end{aligned}
$$

The first line of the expression for out shows how the gain for the frequency w_c (the carrier) is modulated at frequency w_m (the modulation).
The second shows that the output contains two Fourier components – the sidebands.

Vary the input frequencies to see the effect on the carrier and the envelope.
Note particularly the zero crossings of the envelope, for comparison with the next section.
Check for carrier reversal around these zero crossings.
Vary the input amplitudes to see the effect on the envelope.
Explain your observations.

9. Amplitude Modulation (AM)

Make

$x_2 = y_2 = 0$

$x_1 = z_2 = 5 \cos w_c t$ $\quad (f_c = 4 \text{ kHz})$

$y_1 = 10 \cos w_m t$ $\quad (f_m = 50 \text{ Hz})$.

The output is

$$\begin{aligned} OUT &= 5\ \cos w_c t\ + 5\ \cos w_c t\ \cos w_m t \\ &= 5\ \cos w_c t\ (1 + \cos w_m t)\ . \end{aligned}$$

The latter relation shows that the amplitude of the carrier is again modulated, but never passes through zero; note the difference of the zero crossings from section 8.
A component at the carrier frequency is now present, as well as the two sidebands of section 8.

i) Vary the amplitude of the y_1 input to observe degrees of modulation of less than 100%.
Sketch and comment.

ii) An AM waveform may alternatively be generated by making

$$X_1 = 10\ \cos w_c t \qquad X_2 = 0$$
$$Y_1 = A_m\ \cos w_m t \qquad Y_2 = -A_c$$
$$Z_2 = 0\ \ .$$

(What now determines the amplitudes of the carrier and sidebands?)

Check this method, in particular making

$A_c < A_m$ (but $A_c + A_m \leq 10$ – why ?)

to observe the effect of overmodulation.

(But note that overmodulation gives a different effect in a normal AM transmitter.)

10. Phase Modulation (PM)

Make

$x_2 = y_2\ = 0$

$x_1 = 5\ \cos w_c t$ ($f_c\ = 4$ kHz)

$y_1 = 5\ \cos w_m t$ ($f_m = 50$ Hz)

$z_2 = 5\ \sin w_c t$

(i.e. we use quadrature carriers) .

The output is

$$OUT\ =\ 5\ \sin w_c t\ +\ 2{\cdot}5\ \cos w_c t\ \cos w_m t\ \ .$$

The second term can be regarded as being in phase with cos $w_c t$, but with a waxing and waning amplitude (at frequency w_m) .

This phasor is at 90° to the carrier 5 sin $w_c t$, so it advances and retards its phase at rate w_m ; since the amount of phase shift is independent of w_m we have phase modulation rather than frequency modulation.
With only the two sidebands (from the second term) the amount of phase modulation can only be small before the amplitude modulation becomes unacceptable.

Vary the amplitude of y_1 to observe various degrees of modulation; trigger the CRO off z_2 to make the modulation easily visible.

Calculate values of the modulation index (= peak phase deviation in radians), and compare with expected values.

11. Phase Multiplication

This uses frequency doubling.
Make

$X_1 = Y_1$ = the phase-modulated signal output of section 10 .

Note the increase in the amount of phase modulation; the increase in amplitude modulation needs to be countered.

Calculate the index of modulation; is it doubled?

12. Single-Sideband Modulation (SSBSC)

Make a suppressed-carrier modulated wave by multiplying 10 cos $w_c t$ by 10 cos $w_m t$.

$$\begin{aligned} v_1 &= 10 \cos w_c t \ \cos w_m t \\ &= 5 \cos (w_c - w_m)t + 5 \cos (w_c + w_m)t . \end{aligned}$$

Make a second modulated wave using quadrature inputs.

$$\begin{aligned} v_2 &= 10 \ \sin w_c t \ \sin w_m t \\ &= 5 \cos (w_c - w_m)t - 5 \cos (w_c + w_m)t . \end{aligned}$$

If these two modulated waves v_1 and v_2 are added using a Z_2 input the result is

$$V_o = v_1 + v_2 = 10 \cos (w_c - w_m)t$$

so that both the carrier and one of the sidebands are missing.

This method is not popular in practice – e.g. it needs a quadrature generator usable over the whole audio band.

[More simply

$$\begin{aligned} V_o &= 10 \ (\cos w_c t \ \cos w_m t + \sin w_c t \ \sin w_m t) \\ &= 10 \ \cos (w_c - w_m)t \ .] \end{aligned}$$

Note and explain the effect of varying the amplitudes (both equal) and frequency of the modulation inputs.

13. Square-law Detection

Make

$X_2 = Y_2 = Z_2 = 0 \qquad X_1 = Y_1 =$ AM signal ,

using the method of section 9(ii) with $A_c = 5$, $A_m = 5\, m \le 5$.

The input is of the form

$$v_i = 5 \cos w_c t \ (1 + m \cos w_m t)$$

and the output is

$$V_o = v_i^2 \ / \ 10$$

$$= 2{\cdot}5 \ \cos^2 w_c t \ (1 + \text{m} \ \cos \ w_m t)^2$$

$$= 1{\cdot}25 \ (1 + \cos 2w_c t) \ (1 + 2m \ \cos w_m t + ½ \ m^2 + ½ \ m^2 \ \cos 2w_m t) \ .$$

This includes terms at dc, which are easily removed, and terms at high frequency (e.g. $2w_c$, $2w_c \pm w_m$ etc.) which may also be removed.

The only terms in the region of the modulation frequency w_m are

$$v_{of} = 2{\cdot}5 \ m \cos w_m t \ + \ 0{\cdot}625 \ m^2 \cos 2w_m t \ .$$

This method does produce an output at the modulation frequency w_m , i.e. it acts as a detector. However it also produces second-order distortion of the detected output which becomes very significant for full modulation, with $m = 1$.

Check these conclusions in practice.

Keep the AM signal circuit for the next section.

14. Synchronous Demodulation

Make

$X_2 = Y_2 = Z_2 = 0$

$X_1 =$ AM waveform as in section 13

$Y_1 = \ 10 \ \cos w_c t$ (i.e. in phase with the carrier) .

The output is

$$V_o = 5 \ \cos^2 w_c t \ (1 + \ m \cos w_m t)$$

$$= 2{\cdot}5 \ (1 + \cos 2w_c t) \ (1 + m \cos w_m t) \ .$$

This has only one term near w_m , namely (after filtering out dc and high-frequency components)

$$V_{of} = 2{\cdot}5\ m\ \cos w_m t$$

so that we have distortion-free detection capability.

The trouble is that Y_1, the carrier signal produced at the receiver, must be phase-locked to X_1, whose carrier signal is produced at the transmitter.

For example, if

$$Y_1 = 10\ \sin w_c t \qquad \text{(at 90° to the carrier)}$$

then

$$\begin{aligned} V_o &= 5\ \cos w_c t\ \sin w_c t\ (1 + m \cos w_m t) \\ &= 2{\cdot}5\ (1 + m\ \cos w_m t)\ \sin 2w_c t \end{aligned}$$

and there is no component near w_m ; we have just moved the received signal to have carrier frequency $2w_c$ rather than w_c .

Check these conclusions in practice.

It is a matter of simple algebra to find similar conclusions for a suppressed-carrier (DSBSC) signal; however square-law detection is useless.

Keep the AM signal circuit for the next section.

15. Quadrature Amplitude Modulation

This section requires the use of two chassis; get a spare one or cooperate with another group of students.

Suppose that we generate an AM signal (again using the method of section 9(ii) with $A_c = 3$, $A_m = 3\ m \leq 3$)

$$v_o = 3\ \cos w_c t\ (1 + m_1\ \cos w_{m1} t)$$

and add it to a second AM signal, with the same carrier frequency but in quadrature with it, namely

$$v = 3\ \sin w_c t\ (1 + m_2 \cos w_{m2} t)\ ;$$

i.e. generate

$$V_o = v_o + v\ .$$

From the combined signal V_o we may recover the first modulating signal alone using synchronous demodulation with an in-phase carrier

$$10 \cos w_c t$$

while the second modulating signal alone may be also recovered from V_o using synchronous demodulation with the quadrature carrier

$$10 \sin w_c t \ .$$

Demonstrate this technique; you may like to use one or two non-sinusoidal modulating signals.

16. Single-Sideband Detection

If a single sideband is transmitted (SSBSC), the signal is of the form

$$V_i = 10 \ m \ \cos (w_c - w_m)t \ .$$

Square-law detection is clearly hopeless, but synchronous demodulation can succeed if a phase-locked carrier is available. SSBSC signals sometimes contain a small carrier content, allowing this method to be implemented fairly easily.

The demonstration in practice is most simply done using a sinusoidal input (as above) directly, rather than generating it as in section 10.

If a single sideband and a carrier (even a reduced one) are transmitted, a square-law detector could be used.
Verify this in theory and in practice if time permits.
This looks like a simple SSB detection method, but there is a catch.
Can you find it ?

Practical Session 5

INTRODUCTION TO TIMS
(Telecommunications Instructional Modelling System)

1. Procedure

The aim is to become familiar with most of the (dozen or so) different modules that make up TIMS.
A systematic test procedure is suggested, including some sections similar to those in sessions 2-4. You should use this to test the basic function of each module (and if possible some of its limitations) in turn, recording your procedure and results in your report.
Particularly note, for later reference, any features of a module that are not obvious from the front panel or from the 'TIMS Quick Reference' sheet.
You will use in the laboratory, but may not borrow for home use, a copy of the 'TIMS User Manual'.
First read pages 1-3 of the manual.
Switch on TIMS (power switch at rear left of unit).

2. Variable DC (p. 23)

Just check its operation on the CRO.

3. Voltage Controlled Oscillator (pp. 24-25)

Control it with the Variable DC and check its two frequency ranges using the CRO.

4. Frequency Counter (p. 8)

Use it to again check the two frequency ranges of the VCO.

5. Tuneable Lowpass Filter (p. 20)

Check its operation for several TUNEings and on both frequency ranges, using the output of the VCO as a test signal (LO range).

6. Master Signals (pp. 10-11)

i) Check each of these briefly. Using the 2kHz message to trigger the CRO, check that the other four signals are locked in phase to it.

ii) Convert the 8.3kHz sample clock into a sine wave using the tuneable LPF, and check the frequency ratios of the master signals using Lissajous figures on the CRO.

7. Audio Oscillator (p. 5)

i) Briefly check its operation.

ii) Apply the two analogue outputs to the X-Y inputs of the CRO, and check the display over the frequency range.

8. Buffer Amplifiers (p. 6)

Briefly check their operation, using a signal from the VCO.
Leave them set for a gain of 0.5.

9. Adder (p. 4)

Add (setting $g = G = 1$) a 2Vpp sine wave at 1 kHz derived from the audio oscillator and a 2Vpp sine wave at 9 kHz derived from the VCO.
Compare this waveform with that obtained in section 4 of sessions 2-4.

10. Phase Shifter (pp. 13-14)

Briefly check its operation at both low and high frequencies.

11. Multiplier (p. 12)

i) Use it to make a DSBSC signal, peaking at ±2 V, with
$f_c = 100$ kHz $f_m = 2$ kHz .
Are all envelope peaks equal? (balance)
If the inputs are, in turn, connected to ground rather than the signal, is the output zero? (feedthrough)
Have the X and Y inputs equal feedthrough at 100 kHz?
In sections ii and iii, take particular care with triggering of the CRO.

ii) Reduce the modulation input to make the output about 1 Vpp.
Add a 2Vpp in-phase carrier to check that you get AM.
(You may need to slightly phase shift one version of the carrier to get best results in this and the next part.)

iii) Add a quadrature carrier instead, and check that you get PM.
Revert to AM ready for the next section.

12. Utilities Module (p. 22)

Set up the AM waveform as in section 11(ii). Connect this signal to the DIODE + LPF and discuss the output. (It may be helpful to view input and output together, triggering the CRO off the message.)
Keep this circuit ready for the next section.

13. 60kHz Lowpass Filter (p. 26)

i) Check this filter using the VCO output as a test signal.

ii) Apply the AM waveform (as in section 11(ii)) to this filter and check that you get zero output. (No component near f_m.)

14. Quadrature Phase Splitter (p. 15)

Apply a signal from the audio oscillator to both inputs and apply the two outputs to the X-Y inputs of the CRO. Check the display over the full frequency range of specification.

15. The Rest (pp. 7, 9, 16, 17, 19, 21, 22)

Unless you have time to spare you may delay investigation of the other (mainly digital) facilities of TIMS until a later session.

Practical Sessions 6-8

1. Coherent Demodulation

Reading

Schwartz 4.5, 4.6, 4.8; Lecture Notes 6.4-6.5, 7.4-7.7, 7.11, 8.3-8.4, 8.6-8.8.

Notes

A signal modulated on to a carrier may be recovered by multiplying by the carrier, as this translates the signal back to the baseband.

But we may not be given a carrier signal, and may have to use a local one.
Must it be the exact right frequency? If so, is its phase important?

You should generate an AM signal and check the above. You may like to try using a phase-locked loop (see below) to extract an appropriate carrier from the AM signal.

Then generate a SSBSC signal and check how well the method works.

Generate the modulated carriers using the Master signals, with f_c = 100 kHz, f_m = 2·08 kHz.

Report fully, including theory, and comment on the significance of these results for real-world reception.

Amplitude Modulation - Method 1 (Ideas p. 14)

The equation to be modelled is:

$$y(t) \;=\; V \times (1 + m \cos w_m t) \cos w_c t$$

A suitable model is:

$$AM \;=\; V_2 k_2 \left[1 + \left(\frac{V_1 k_1}{V_2 k_2} \cos w_m t \right) \right] \cos w_c t$$

where

$$m \;=\; \frac{V_1 k_1}{V_2 k_2}$$

$$V \;=\; V_2 k_2 \quad .$$

This model allows separate control of the amplitude of the sidebands and the carrier, as measured at the output of the Multiplier. This is done by controlling the ratio of the amplitudes of the message and the DC at the output of the Adder. When their amplitude values are equal at this point, then the depth of modulation of the AM signal is 100% (or $m = 1$).

Note that

- the depth of modulation m is **not** a function of the amplitude of the high-frequency signal into the multiplier.
- the amplitude of the AM signal itself is directly proportional to the amplitude of the high-frequency signal into the multiplier.
- there is no need in this arrangement to make any phase adjustments.
- by lowering the DC value to zero, the AM signal is reduced to a double-sideband suppressed-carrier signal (DSBSC).

Coherent Demodulation (Ideas p. 21)

This is the standard demodulator for DSBSC signals (see Fig. 2 below). But it will also demodulate other linearly-modulated carriers. It **will** demodulate SSB, for example, but only provided that there is not another sideband on the other side of the carrier. Should this other sideband exist (e.g., as the other sideband of an ISB signal, or perhaps an adjacent unwanted signal), then a true SSB demodulator is required.

Procedure

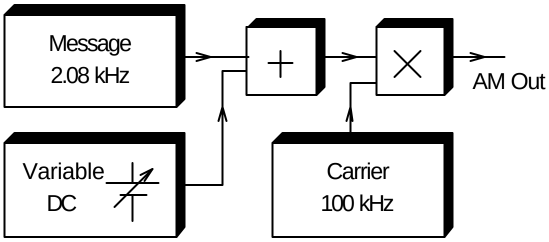


Fig. 1. AM Generation

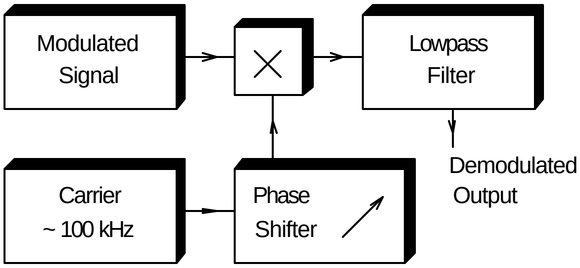


Fig. 2. AM Demodulator

Set up an AM signal as in Fig. 1. Demodulate as in Fig. 2, first using the same carrier, then trying phase-shifted versions. (You may use the signal generator for f_m if you wish.) Now try using an output from the VCO as a local carrier; first adjust the VCO to close to 100 kHz. (This should be hopeless.)

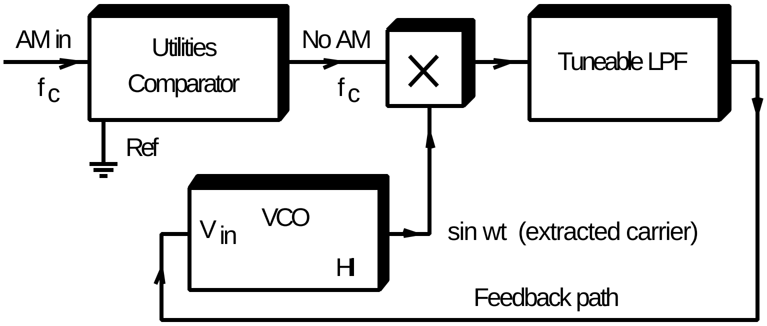


Fig. 3. Phase-Locked Loop

You may now like to try extracting a good carrier from the AM signal using a phase-locked loop, as in Fig. 3.

This set-up will recover a carrier from the complete AM signal, but needs careful adjustment. First reduce the modulation to zero. Set both Tuneable LPF controls mid-way. Turn the VCO gain right down, and adjust f_0 until very close to f_c .
Then turn up the VCO gain just a little until the two signals lock in frequency.
Adjust the f_0 control to get the two roughly in quadrature (0 V out of the LPF).
Perhaps adjust the VCO gain to the centre of the usable range.

Now turn up the modulation and check that it still locks.

Check that this extracted carrier, with perhaps some phase shift, may be successfully used instead of the original carrier in the demodulator of Fig. 2.

Single-Sideband Generation – Phasing Method (Ideas p. 18)

(See Fig. 4 below.)

The output may be changed from one sideband to the other depending upon
- which local carrier leads the other.
- whether the adder is additive or subtractive.

Amplitude balancing controls are available in the Adder.

The output is based on a carrier of frequency w_c .

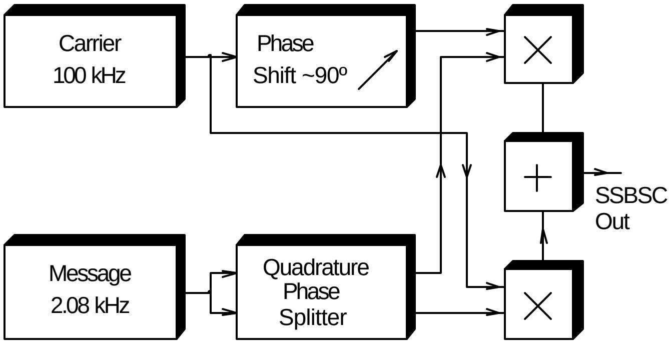


Fig. 4. SSBSC Modulator

Generate a SSBSC signal as in Fig. 4, then demodulate as in Fig. 2.
(See also note below.)

As with AM, try first the correct carrier, then a phase-shifted one, then one at not quite the right frequency.

Notes on SSB Generation

The Phase Shifter in Fig. 4 could perhaps be saved by just using quadrature signals.
On the other hand, it can be adjusted around 90° for best performance.

Each half of the Quadrature Phase Splitter produces a phase shift which is a function of frequency, but one half produces, at all audio frequencies, close to 90° more phase shift than the other.

Practical Sessions 6-8

2. Quadrature Amplitude Modulation

Reading

Lecture Notes 7.5-7.7, 7.8, 7.11.

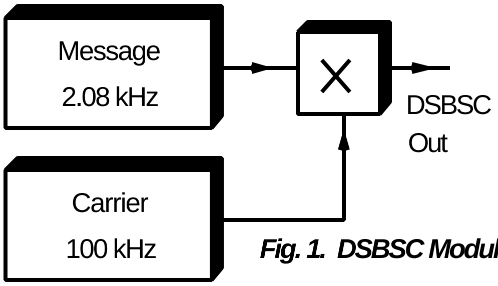


Fig. 1. DSBSC Modulator

If a carrier signal is multiplied by a modulation signal, then the result is a DSBSC (double-sideband, suppressed-carrier) signal. If a quadrature carrier is used, the same band of frequencies results. Surprisingly, however, two independent messages may be sent in the same bandwidth and recovered separately using synchronous detection.

Double-Sideband Suppressed-Carrier – Generation (Ideas p. 12)

$$DSBSC \;=\; V \cos w_m t \;\cos w_c t$$

This process can be modelled with a multiplier.

Quadrature Amplitude Modulation (Ideas p. 29)

A QAM signal is made by adding two double-sideband (DSB) signals, derived from the same (suppressed) carrier, in phase quadrature. The DSB signals will in general carry different messages.

If the bandwidth of each of the messages is B Hz, then the bandwidth of each of the DSB signals will be ($2 \times B$) Hz. Each DSB signal occupies the same place in the frequency spectrum, so the bandwidth of the QAM signal is also ($2 \times B$) Hz.

Phase-Division Multiplexing And Demultiplexing (Ideas p. 35)

The phase-division multiplexer is a block which appears in many guises, depending upon the nature of the signals involved. In this example the two are different (independent) voice channels, each converted to a DSB signal based on the same (suppressed) carrier. The carriers are ideally in quadrature (at 90° phase difference).

Each DSB signal has the same bandwidth, and occupies the same band in the frequency spectrum at the same time. The demodulator can separate the two channels by virtue of their phase differences.

Note especially that the carriers (at the transmitter) do **not** need to be in exact quadrature. Their phases only need to be different for complete channel separation to be achieved. The quadrature condition ensures best signal-to-noise performance; but even a 45° error from this is not exactly disastrous. What is essential, however, is that, once the transmitter phase has been set, the receiver phase is adjusted, not to maximise the wanted channel, but to **null** the unwanted channel.

How carrier-frequency synchronisation is achieved, and phase tracking maintained, is of course of great importance in practice. This experiment serves to illustrate the principles of the multiplexing process.

Question

If 90° is not essential, then why not use a smaller angle, and fit in an extra channel (or two)?

Double-Sideband Suppressed-Carrier – Demodulation (Ideas p. 13)

$$DSBSC \;=\; V \times x(t) \times \cos w_c t$$

Here the message is $x(t)$. A simple form of message is

$$x(t) \;=\; A \cos w_m t \;\; .$$

A demodulator will recover the message $x(t)$ from the DSBSC signal. Demodulation may be performed by multiplying the DSBSC signal by a carrier of the same frequency and phase, and filtering the message from the result, thus:

$$\left[x(t)\cos w_c t\right]\times\left[V\cos(w_c t+f)\right] \;=\; \tfrac{1}{2}V\, x(t)\, \cos f \;+\; \tfrac{1}{2}V\, x(t)\, \cos(2w_c t+f)$$

Note that the phase f is set to maximise the amplitude of the recovered message. The term at twice the carrier frequency is removed with a lowpass filter.

Quadrature Amplitude Demodulation (Ideas p. 30)

If the DSB signals are derived from different messages, these messages can be independently recovered by synchronous demodulation with appropriate phases.

Note that if, at the transmitter, there is a phase error from quadrature between the two DSB signals, then this demodulator will be able to separate the two messages if its Phase Shifter is re-adjusted to match that of the transmitter; complete suppression **can** be achieved.

The price of an error from phase quadrature is a reduced signal-to-noise ratio (because of a reduced signal).

Procedure

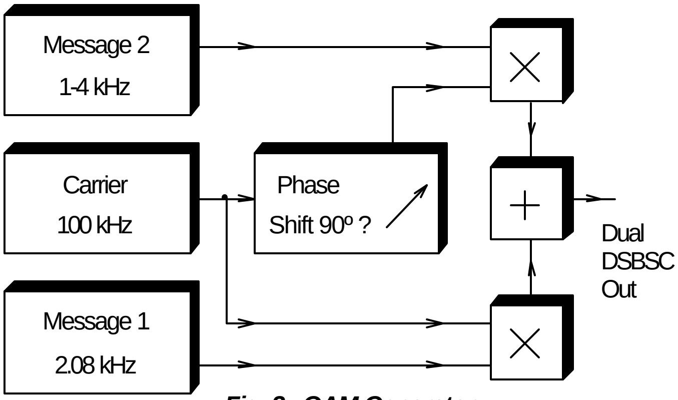


Fig. 2. QAM Generator

Set up the combined message as in Fig. 2.
Use the 2kHz Master signal for one message and the Audio Oscillator for the other.
Use the 100kHz Master signal for the carrier. At first, just use the cos and sin outputs instead of the Phase Shifter; as noted above, an exact 90° is not needed.

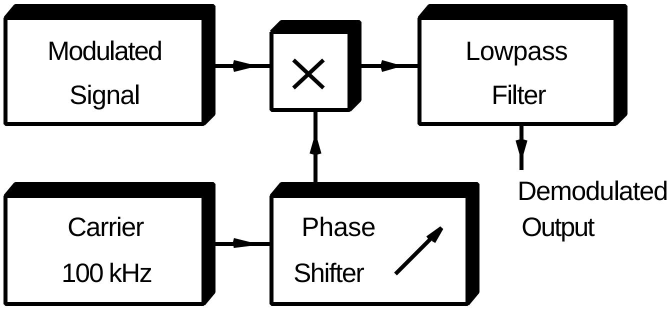


Fig. 3. Demodulator

Demodulate as in Fig. 3.

Check that, by careful phase adjustment, one signal may be received with little sign of the other. Is the adjustment correct when the carrier used is exactly in phase with the original carrier?

Check that the modulator need not use exactly 90° phase shift between the carriers.

Practical Sessions 6-8

3. Wideband FM

Reading

Schwartz 4.9-4.11; Lecture Notes 10.1, 10.5, 10.9.

Wideband FM Generation (Ideas p. 27)

The VCO can be used to generate a wideband FM signal. The TTL output from the VCO will be rich in harmonics of the carrier frequency, and each of these will be frequency modulated. It is convenient to extract the fundamental with a lowpass filter (e.g., the 100kHz lowpass Channel Filter if the VCO is set to a frequency of about 100 kHz) to obtain a 100kHz FM signal.

FM Demodulation (Ideas p. 28)

There are many methods for demodulating FM signals. For example:

- phase-locked loop (see below)
- FM-to-AM conversion (differentiation) followed by envelope detection
- pulse-counting detection (see below)
- slope detection using a single tuned circuit plus envelope detection
- ratio detector
- Foster-Seeley discriminator.

Procedure

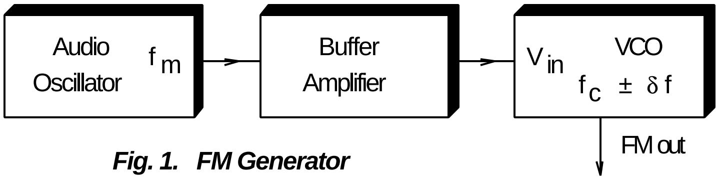


Fig. 1. FM Generator

Using a VCO, an FM generator is simply set up as in Fig. 1.
With V_{in} = 0, set the VCO frequency to 90 kHz.

Set the Audio Oscillator to give a few volts at a frequency of 1 kHz.
Increase the VCO gain to give a frequency deviation of ± 10 kHz
(i.e. f_c = 90 kHz, f_m = 1 kHz, b = 10).

Check on the CRO that this is in fact an FM generator, i.e. that the frequency deviation depends only on the amplitude of V_{in} , not on its frequency.

(Without a fixed carrier, it is not easy to check that the phase deviation does differ.)

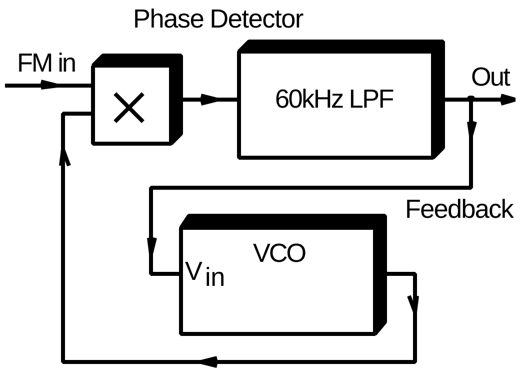


Fig. 2. FM Receiver 1 (PLL)

This FM signal may be demodulated using a phase-locked loop as shown in Fig. 2. Before wiring it as shown, set up the VCO parameters to be much the same as those of the generator. (The gain needs to be fairly low, else the feedback loop will oscillate.)

Set the circuit up, with the gain of the 60kHz LPF set to give a few volts of output. Check that the output amplitude and frequency vary with that of the original modulation signal.
Vary the parameters of the feedback loop of Fig. 2, and record and comment on the results of those variations; record more than just the output waveform if the circuit misbehaves.

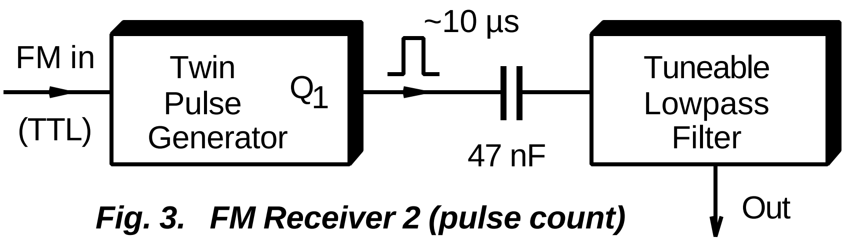


Fig. 3. FM Receiver 2 (pulse count)

Now alter the FM generator to have parameters: f_c = 80 kHz, f_m = 1 kHz, b = 10.
Demodulate the signal using a pulse-counting method; i.e. produce a standard pulse for each period of the FM input and filter this pulse train. The basic set up is as in Fig. 3; note that, because the pulse train has a large dc component, the ac output will be rather small unless that dc is removed. You may do this by adding a dc offset or by capacitive coupling as shown in Fig. 3.

Check that the pulse width is appreciably less than a period of ($f_c + b\ f_m$), otherwise reduce f_c somewhat – but check that (f_c - $b\ f_m$) is not too small.

Record the various waveforms and compare your results with an appropriate theory.

Practical Sessions 6-8

4. Sampling and Reconstruction

Reading

Schwartz 3.2, 3.3; Lecture Notes 12.1-12.2, 12.6-12.8.

Sampling

The sampling theorem shows that a band-limited signal may be completely reconstituted from a sequence of suitably spaced and shaped samples.

The circuit of Fig. 1 (below, without the lowpass filter) allows a suitable set of samples to be taken of a single sine wave. The sampling theorem requires that, for undistorted reconstruction,

$$f_s > 2f_m \ .$$

For the case

$$f_s = 2f_m$$

the reconstruction filter would need to have a 'brick wall' response. For a practical filter, with a finite transition-band slope, it would need to be correspondingly wider.

Reconstruction

The circuit of Fig. 1 (below) recovers the original sine wave from the set of samples above using the lowpass filter.

The above could be used as a simple qualitative demonstration of sampling and reconstruction. It can be made quantitative by examining more closely the relationships between the system parameters, including f_s , f_m , pulse width, filter cutoff f_c , and input and output amplitudes.

Sampling and reconstruction as illustrated above is an example of pulse-amplitude modulation (PAM).

Bandlimited Channel; Pulse Shaping

The above is a simple introduction to sampling and message recovery. The experiment can be extended to cover pulse shaping, channel band-limiting, and so on.

Procedure

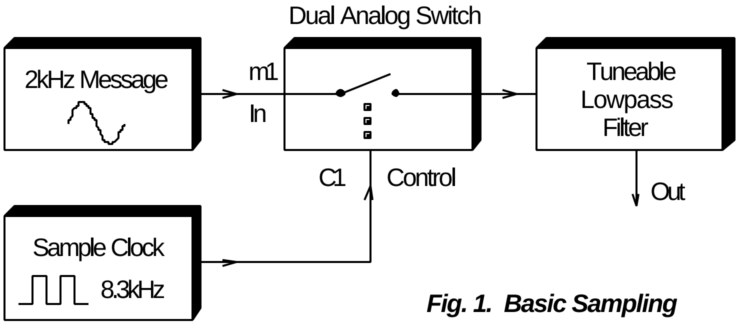


Fig. 1. Basic Sampling

First set up the Tuneable LPF to cut off at about 4 kHz (near top of NORMal range). (Gain = 2?)

Set up the basic sampling circuit of Fig. 1. Record the sampled waveform. Check that the output shape and size is as predicted by theory. Comment. (Note− the LPF inverts.)

Now replace the 2kHz message in Fig. 1 by a variable-frequency one from the Audio Oscillator. Check that the output is still the same shape as the input for frequencies of a few kHz or less. Then try the full range of audio frequencies, record some representative results, and comment.

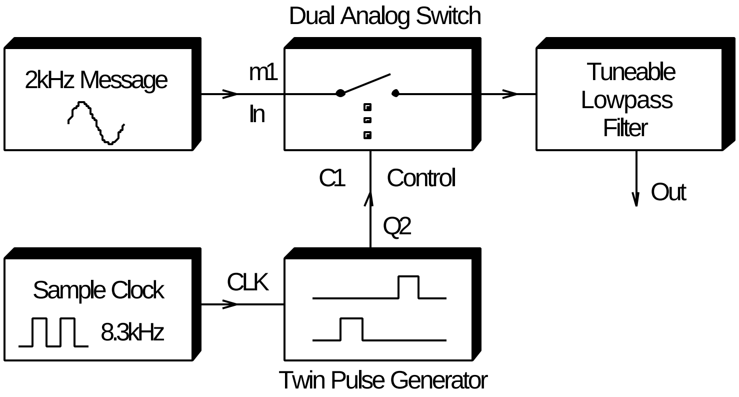


Fig. 2. Variable Sampling

Set up as in Fig. 2, which allows the width and phase of the sampling pulse to be varied. Record the effect of these variations and check, for a few cases, that the output shape and size are as predicted by theory.

Time-Division Multiplexing – TDM (Ideas p. 32)

If samples of two or more bandlimited messages are interlaced in time, the resulting signal is known as a time-division-multiplexed signal. Using commutation techniques, the samples from a particular channel can be isolated at the receiver, and then reconstructed (as illustrated previously for the single-message case).

The model of Fig. 3 (below) is a two-channel time-division multiplexer. By reducing the sample widths, more channels can be accommodated.

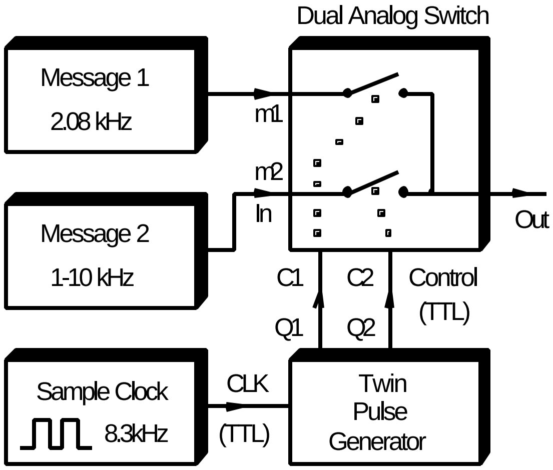


Fig. 3. Time Division Multiplexing

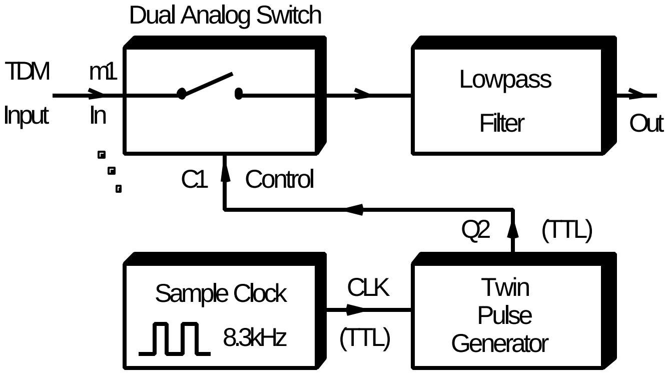


Fig. 4. Time Division Demultiplexing

Set up a time-division multiplexing system as in Figs 3 and 4, with each message sampled for a little less than half the sampling period. Try a variety of pulse widths and delays in the recovery circuit, and record and explain what you see.

Practical Sessions 6-8

5. Quadrature Phase-Shift Keying

Reading

Schwartz 4.3; Lecture Notes 23.2-23.3, 23.6-23.7.

Pseudorandom Sequence Generator

Using a common external clock signal, the Sequence Generator outputs two independent pseudorandom sequences X and Y. A SYNC output is provided which is coincident with the start of the sequences. The sequences may be stopped and restarted at any time via front-panel controls. Sequences X and Y are available as either standard TTL or analog-level outputs.

Use

An external clock signal must be provided to operate the Sequence Generator. This may be sinusoidal or TTL; separate input sockets are used.

The sequences may be stopped at any time by either depressing the Reset button or applying a TTL HI signal to the Reset input. To restart the sequences from the beginning, release the Reset button or apply a TTL LO to the Reset input.

The length of the sequences may be selected by a PCB-mounted DIP switch.
Four independent sequence pairs are available, from lengths of 2^5 to 2^{11} .
The sequences are selected as follows.

DIP Switch Code	***n***	**Sequence Length 2^n**
msb 0 0	5	32
0 1	8	256
1 0	8	256
1 1	11	2048

Theory

Where bandwidth is limited but noise is not severe, e.g. in the telephone system, it is common to transmit n bits of data at a time by sending one of 2^n distinguishable signals / symbols.

Here we send 2 bits at once, using the signals $+2\sin(w\,t + p/4)$, $-2\sin(w\,t + p/4)$, $+2\cos(wt + p/4)$, $-2\cos(wt + p/4)$. These may be derived from data b_1 , b_0
by generating

$$X\ \cos wt + Y\ \sin wt$$

where

$$X = +2 \text{ if } b_1 = 1\ ,$$

$$X = -2 \text{ if } b_1 = 0\ ;$$

similarly for Y and b_0 .

The set of signals is often pictured as a 'constellation' – cf. Schwartz Fig. 4.13 or Lecture Notes Fig. 24.8(b).
Since their amplitude is constant, demodulation clearly needs a synchronous carrier; $\cos wt$ for X but $\sin wt$ for Y.

The modulator may be rather like Schwartz Fig. 4.14 or Lecture Notes Fig. 24.9(a),
the demodulator like Schwartz Fig. 4.15 or Lecture Notes Fig. 24.9(b).

Procedure

You will use the Sequence Generator to produce a 2-bit stream of data, so first study its characteristics (above). Use the bipolar (yellow) outputs to get signals swinging between -2 and +2 volts. In observing the data stream, trigger the CRO off the SYNC of the Sequence Generator, and use ×10 MAGnification of the timebase. A sequence length of 32 is appropriate.

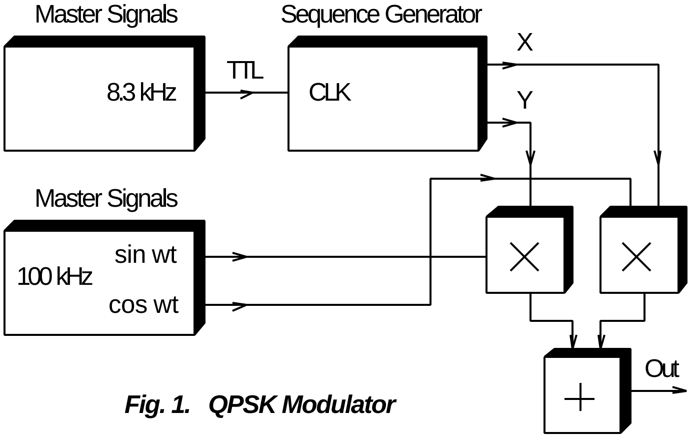


Fig. 1. QPSK Modulator

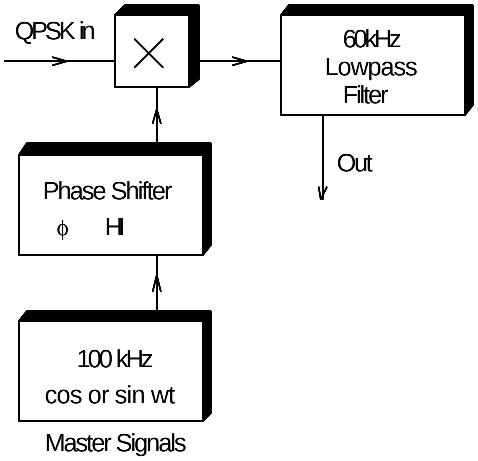


Fig. 2. QPSK Demodulator

The set-up is as shown in Figs 1 and 2. In the receiver you may like to use the Phase Shifter for fine adjustment and for selecting X or Y as the final output.
(We save on use of Multipliers by getting only one output at once.)

You may square up the output pulse using a Comparator (Utilities), with 0 V an appropriate REFerence level.

Watch out for an output which may be an inverted version of the original input.

You may like to replace the 60kHz LPF by a Tuneable LPF, which can scarcely be made wide enough to get reasonably-shaped output pulses.

Practical Sessions 6-8

6. Amplitude-Shift Keying; Quadrature ASK

Reading

Schwartz 4.2; Lecture Notes 22.1-22.2, 22.5-22.5.

100kHz Channel Filters

Three switch-selectable, 100kHz channels are provided, comprising two different filters and one straight-through connection.

Use

Only one channel may be selected and used at a time.

Note that each of the three channels may be AC or DC coupled by front-panel toggle switches.

Channel Characteristics

Before using any of these three channels in experiments, each channel should be characterised by actual measurement of amplitude and phase responses.
As a minimum, the cut-off and stop-band frequencies should be measured, using the VCO and true-rms meter modules or an oscilloscope.

Basic Specifications

Input coupling

AC or DC, Channels 1 to 3

Channel responses

Channel 1 straight-through
Channel 2 bandpass filter (as per specs, but lowpass in fact)
Channel 3 lowpass filter (as per specs, but bandpass in fact)

Stop-band attenuation

Approx. 40 dB

Procedure

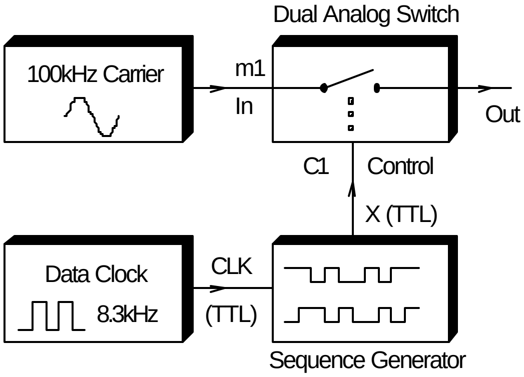


Fig. 1. ASK Modulator

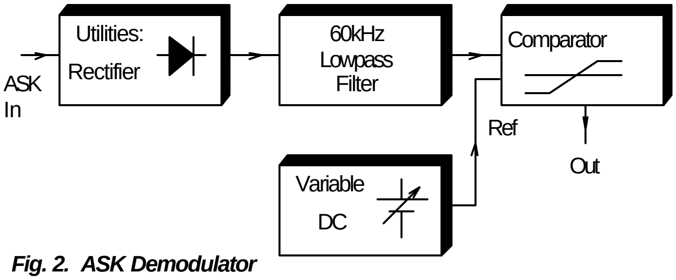


Fig. 2. ASK Demodulator

Start with the simple ASK system of Figs 1, 2. Initially use a 100kHz Master signal for the carrier and the 8·3kHz Master signal for the data clock.

Use a sequence length of 32, trigger the CRO off the SYNC of the Sequence Generator, and put the CRO timebase on ×10 MAGnification, to ease the task of checking the various waveforms.

Set the REFerence level of the Comparator about half-way up the input pulse.
Record appropriate waveforms and check the data output against the *X* data input.

Now try a simpler detector, using the Diode + LPF of the Utilities module rather than the Rectifier + 60kHz LPF.
Note the effect of varying the REFerence input to the Comparator.

Next try band-limiting the transmission using a bandpass 100kHz Channel Filter. Examine particularly the first few and last few cycles of a transmitted pulse to estimate the filter properties. (Or measure them using the signal generator.)

Using the signal generator rather than the 8kHz Master signal as the data clock (analogue; dial 2Vpp), check how high the data rate may be, both with and without the 100kHz Channel Filter.
What limits the data rate? (We'll look more fully into such situations later in ELEC321.)

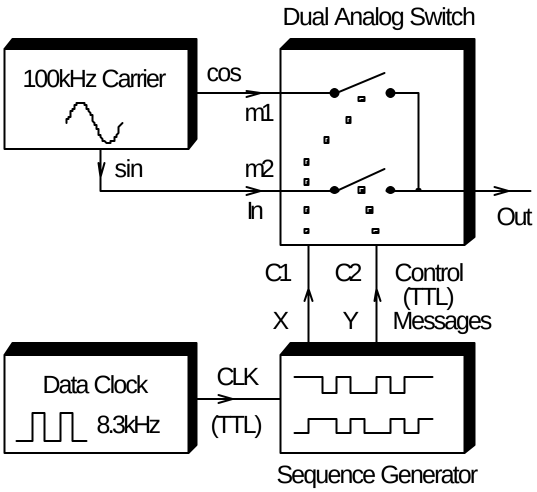


Fig. 3. QASK Modulator

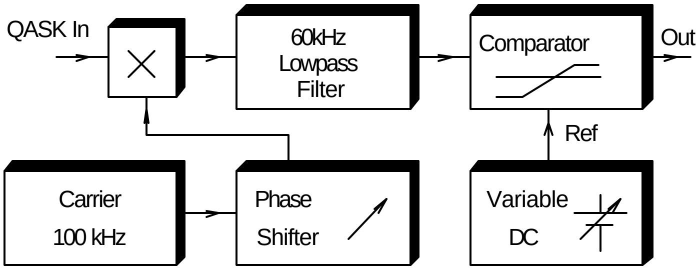


Fig. 4. QASK Demodulator

Set up the QASK modulator of Fig. 3. Similarly set up half the demodulator as shown in Fig. 4, using cos*wt* to get one data stream and sin*wt* to get the other data stream in turn. Note and comment particularly on the need to choose a suitable REFerence level for the Comparator.

Practical Sessions 9-10

INTRODUCTION TO COMMSIM

These sessions will be spent using the CommSim (Communication Systems Simulation) package, which permits easy simulation of communication functions and systems.

Session 9 will be a tutorial one, to get you familiar with CommSim. You will work on your own at a PC doing some simple exercises with full instructions. You will then do a slightly harder exercise with less detailed instructions; it will simulate an earlier practical exercise and involves most of the skills needed for later exercises.

There will be no report required for Session 9 (but Session 10 may carry more weight than a normal one). It is thought that three hours will be adequate time to complete Session 9; you may even be able to get started on the work for Session 10, which will be allocated at Session 9.

Please arrive on time for Session 9, which will start with a talk by Steven on various system details. (If you miss it, you may have trouble even logging on.)

Session 10 will have you u sing CommSim to perform a variety of communication-systems functions and get hard copy of the resulting waveforms and spectra.
Each student will perform, working on their own, three exercises chosen from a pool of nine. The selection of exercises has been done to (try to) give you each work of comparable difficulty and diversity. Which set you get will be decided by a blind draw at Session 9; you will therefore have time to revise the theory before starting.

While you are welcome to perform exercises of your own devising (and such attempts will be highly regarded), our experience is that it is not easy to choose all parameters appropriately at the first attempt, and the whole lot often has to be repeated.
Accordingly, unless you are particularly confident, we suggest that you basically follow the course suggested in the handouts available at Session 10 (which must be returned at the end of that session). This has the advantage that, if you run into trouble, you can compare your results with our records. Improvements or additions to the suggested exercises will be highly regarded. (See Footnote.)

Make sure that every printout you make includes your name and the date somewhere, perhaps in the header as detailed for Session 9. Make sure also that every waveform is fully labelled. Make each screen as useful as possible before printing it. Keep your own record of all parameters which do not appear on a printout, or manually incorporate them on the screen, for example by adding a descriptive title at the end of the header and notes on details of the plots at the end of the footer − but such comments are not permanently attached to the file− or using a label. Since CommSim files are stored in only a few kilobytes each, feel free to save each version of any file separately; this has the advantage that an identifying file name will be attached to each printout. For each file name use the format 321g03p09s1a; g03 if that is in your user name, p09 for Practical Session 9, s1a for Section 1, Figure 1a; this should identify each printout uniquely.

A report will be required for Session 10, being due about one week after its completion. Make sure that your report is self-contained, with an introduction to the relevant theory and with frequent reference, in detail and with associated calculations, to how the results support the theory.

Practical Session 9

COMMSIM TUTORIAL

CommSim is a package that simulates communication systems. The purpose of this tutorial is to introduce you to CommSim, getting you to near the point where you could devise and test your own implementation of a complex communication system.
CommSim is installed for each PC in the ELEC321 laboratory.

Log on at a convenient computer; you will be given a user name and password at the start of the session. Launch CommSim (Start | Programs | Commsim2001 | Commsim2001 | left-click). Use the full screen.

There may be a pane on the left, labelled Diagram 1, which is not much use; left-click-drag its right boundary left to conceal it if necessary.

At the top is a menu bar, which lets you do almost anything. Check out the dropdown windows.
File: The usual.
Edit: Note Repaint Screen, Preferences. In the latter, tick Snap to Grid.
Simulate: Under Simulation Properties, for the moment choose a Frequency of 4096000 Hz to End at 0.001 seconds, both for Defaults and Range.
Blocks: Do a quick survey of what is available.
View: Tick: Block Labels, Connector Labels, Status Bar, Tool Bar, and Presentation Mode not Display Mode for the moment. Under Colors, choose a dark green for Wires. (We want to see them on-screen and from a black-white printer.)
Comm: Do a quick survey of the comms blocks that are available.
Help: Take a quick look. (It is not always very helpful, to my mind!)

Below the Menu bar are icons/shortcuts for popular functions. They are grouped into separate toolbars. Look them over:
Main: The usual things to deal with files and editing the display. Hold the mouse over the icons in turn, when a tooltip will give you a shortform identification of the function and the status bar at the bottom of the screen will give you more detail.
Sim Control: Starts with a green arrowhead. CommSim runs a simulation on sampled data, and these icons control the simulation.
Producer blocks: That is, blocks that produce signals, starting with [1] (a constant).
Consumer blocks: That is, blocks that accept signals, starting with [0] (display) and plot.
Annotation blocks: Starting with label.
Arithmetic blocks: Starting with abs.
Boolean blocks: Starting with [>].
Do a quick survey of all the above icons.

Now start using CommSim.

Left-click-release the sinusoid icon ([~]). Locate the mouse at a convenient point and left-click-release again to add a sine-wave generator to the diagram. Do the same for a plot icon.

Right-click-release the sine-wave generator to set up its properties. Give it an amplitude of 1 at 1000 Hz and label it appropriately.

Move the cursor over the output lead of the sine-wave generator until it becomes an upright arrow. Left-click-drag a wire from there to the dark blue input arrowhead (2nd down) of the plot block.

Move the cursor to an edge of the plot block (double arrow) and left-click-drag the edges in turn to make the block a more appropriate size and shape.

Click the green-arrowhead icon to run a simulation.

Edit the properties of the sine-wave generator so that it starts with a phase of 90° at t=0. (Check by running a simulation. Unfortunately, phase itself cannot be specified, so if the frequency is changed the phase will not be 90°.)

Move the cursor to the input of the plot block and drag the wire from the blue to the light-green input. Run the simulation again and note the difference. Drag the wire to an empty spot on the diagram and release the mouse button to remove the wire. Now reconnect the wire from the generator to the blue plot input. Run the simulation again.

Right-click-release the plot block to change its properties. Under Axis, change the Y scale limits to +2 and -2, and the X scale limits to 0.5 and 0 msec. (use Time Scaling). Label the Y axis and the plot block; Title will label the whole block, while Subtitle may be used to identify individual waveforms. Left-click OK and check the effect; note that no simulation run was needed. Run the simulation again and note that the axes lose your settings. In the plot block properties, under Options, choose Fixed Bounds; change the axes as above and run a simulation, noting that the Y axis does not change this time but the X axis does, to display all samples.

Move a block to some other spot by moving the cursor over the block (arrowed cross) and left-click-dragging the block elsewhere. Note how the wire moves with the block.

Use shift-left-click to select the sine-wave block, and type <Delete> to remove it (and the attached wire). Recover it (but not the wire) with Edit | Undo. Note that you need to click elsewhere to deselect.

Add a 4kHz sine-wave generator (1 V, 90°) and a summing junction ([Σ]) to the diagram. Make connections to add the two sine waves and display the result as an orange trace (2nd plot input up). Display the 1kHz sine wave in blue (2nd plot input down). You may need to shuffle the blocks around somewhat to get a good clear schematic with no wires or labels or blocks obscured. Try to make as many wires as possible have no corners. Run the simulation. (The two trace colours suggested are easily distinguished on the screen and on a black-white print, where one is near black and the other dotted dark grey.)

Note: When you edit a diagram, the screen may be not fully updated. You can fix this using Edit | Repaint Screen, but this is used often enough that you may like to add a special icon for this function.
Go through View | Toolbar | User | OK, then Edit | Toolbar and for Button 0 choose Function | Edit -> Repaint Screen | OK.

Try selecting a set of blocks by left-clicking to their top left and dragging a dotted box over the selected part of the diagram. Move the selected section a little to check.

At this stage, you will probably have to add the printer to your configuration. Go through Start | Settings | Printers click, Add Printer double-click, NEXT click, Network Printer tick, NEXT click, Find Now click, E6A219-laser double-click, FINISH click.

Add your name and the date and time to the diagram, so that your printout can be identified; go to File | Page Setup, then after $F put a few spaces, then $D (date & time), then a few spaces, then your name in full. Choose Landscape and Fit Diagrams to Page. Check that all is well with File | Print Preview, then save and print your diagram. You should choose a resolution of 600 rather than 300 dots per inch.

Incidentally, if at some time you need more than one plot block, a good idea is to get one just as you want it, then Edit | Copy and Paste (or Ctrl+C and Ctrl+V) it to get more of just the same size and properties. (But use a new block for your first spectrum, perhaps.)

Now do a simple exercise without detailed instructions.

The aim is to demonstrate that, if cosine waves at frequencies f_1 and f_2 are multiplied together, the resultant is the sum of cosine waves at frequencies (f_1+f_2) and (f_1-f_2) :

$$2\cos w_1 t \cos w_2 t = \cos(w_1 + w_2)t + \cos(w_1 - w_2)t \ .$$

f_1 = 4 kHz and f_2 = 5 kHz are appropriate values. Include blocks to perform the multiplication and the sum mentioned, and compare the results. Plot plenty of waveforms, using no more than 2 per plot (dark blue and orange) and making the truth of the proposition very clear to see. Make sure that your name and the date/time are included on the printout page.

Now add a different layer of proof by determining the spectrum of the product signal. Change the Simulation Properties | Range to run for 8 msec. at 4·096 MHz. Choose a spectrum analyser block using Comm | Operators | Spectrum (Real) and add it to the diagram.

Right-click-release it to set up values of Trigger Mode | Triggered, Spectral Output | Magnitude / Phase, FFT Size | 32k, Output Freq Units | kHz, FFT Window Type | Rectangular, Power Spectrum Units | dBm/Hz, Load | 1 ohm, Number of FFT Averages 1, ignore Unwrap Phase and Remove Linear Phase.

Set up a trigger to start the analysis using Comm | Signal Sources | Impulse and connect its output to the Trg input of the analysis block. Add a plot block to display the spectrum; right-click the plot block and select External Trigger, which will now be a (red?) circle above the normal inputs.
Connect Trg, Mag() and freq() from the analysis block to the trigger input, the blue input and the pink (magenta; 4th arrowhead down) input respectively of the plot block.
Note that two of the wires are thick, to denote that they are vectors, not simple signals.
Change the plot properties, selecting XY Plot using X-axis value 4. Insert labels, noting that the X-axis is now trace 4. Make sure that the axis values are unbounded.

Run the simulation. The spectrum display will not have very appropriate axes, so you will need to change them; with appropriate choices you can zero in on the exact values of the peaks of the response, for comparison with theory. Note however that there may be a bug in the software that sometimes ruins the plot if the vertical axis is changed appreciably, so you may be unable to print with your preferred axes.
(Sometimes the screen display is OK but the print not, sometimes the reverse.) Even better, perhaps, is to right-click the spectral plot, then Save Data to File. This file will be a simple list of x-y values; it is very long, but it is easy to delete large slabs of useless data and produce a compact table of the vital data. For better printing from Notepad, if that is what .dat files open to, use Edit | Set Font. Another ploy is to read off x and y values using cross-wires: right-click the plot block and click Read Coordinates, when you can record quite precise values (particularly if you zoom in on a relevant section of waveform by altering the axis limits).

You may like to save your diagram at this stage, as what follows may make it too crowded, and you may like to remove some plots.

Another proof of the proposition is to put the product signal through a filter to select one or other of the predicted Fourier components, and check that the output is as predicted.

Add a filter to the diagram using Comm | Filters | FIR (Finite Impulse Response).
Choose Number of Taps | 8192 or 8191, Cutoff Freq 1 | 1500 or 7000 Hz for Filter Type | Lowpass or Highpass respectively, Window Type | Rectangular, OK.
Connect the product signal to the input, run a simulation and plot the output.
Check the result, first selecting one component and then the other (or use two different filters for simultaneous plots).

Note that, if you finish early after thoroughly checking all of the above, you may start on the work for Practical Session 10. You should probably start with the topic whose theory you understand best, and prepare for the other exercises in the intervening week by studying the theory behind them.

Practical Session 10

EXERCISES USING COMMSIM

See the earlier notes about this session, particularly the final warning about comparing the precise results obtained with the theory; this is what your report will be judged on.

The exercises to be performed have been split into three groups, and each student will be allocated one from each group.

Group A Waveforms

1. Sampling and reconstruction
2. Quantisation noise and companding
3. Eye patterns

Group B Analogue modulation

4. AM generation and detection
5. SSBSC generation and detection
6. FM generation and detection

Group C Digital modulation

7. OOK
8. FSK
9. 16-QAM

A brief description of the content of each exercise is given below, and detailed suggestions will be provided in the laboratory.

1. Sampling and reconstruction

- Set up a sampling waveform
- Set up an input message
- Sample the message
- Filter the sampled signal to recover the message
- Find the spectrum of the sampled message
- Compare with the spectrum of the sampling signal (the pulses)
- Repeat all this with a signal near half the sampling frequency
- Repeat with a signal chosen to demonstrate aliasing

2. Quantisation noise and companding

- Set up a signal
- Compress the original signal using μ=255
- Expand the compressed signal and plot its error
- Quantise the original signal to 8-bit accuracy
- Plot its error
- Determine the rms error
- Quantise the compressed signal
- Expand this signal and plot its error
- Determine the rms error
- Repeat for several other signal sizes

3. Eye patterns

- Set up a square wave
- Put it through a lowpass filter
- Avoiding the filter transient, plot the eye pattern
- Add noise to the original signal and filter the result
- Plot the eye pattern of this filtered noisy signal
- Repeat for several degrees of noise and filtering
- Repeat all this for a pseudo-random bit stream

4. AM generation and detection

- Generate AM by multiplying a carrier by a suitable signal
- Generate AM by adding a DSBSC signal to a carrier
- Plot the spectrum of the AM signal
- Recover the modulation signal from the AM using synchronous detection
- Attempt synchronous detection using a quadrature carrier
- Attempt synchronous detection using an off-frequency carrier
- Recover the modulation signal from the AM using rectification and filtering

5. SSBSC generation and detection

- Generate a SSBSC signal using in-phase and quadrature carrier and modulation
- Check the spectrum
- Make a small change to generate the other sideband
- Check the spectrum
- Recover the modulation signal from the SSBSC using synchronous detection
- Attempt synchronous detection using a quadrature carrier
- Attempt synchronous detection using an off-frequency carrier

6. FM generation and detection

- Generate AM by adding a DSBSC signal to a carrier
- Make a small change to produce narrowband FM
- Check the amplitude modulation and phase deviation of this signal
- Set up a wideband FM signal
- Check the frequency deviation
- Check the spectrum
- Demodulate the FM signal by differentiation, rectification and filtering
- Demodulate the FM signal using zero-crossing pulses

7. OOK

- Set up a carrier signal
- Set up a square-wave data signal
- Generate an OOK signal
- Check its spectrum
- Recover the data signal using synchronous detection (including filtering and squaring)
- Recover the data signal using rectification, filtering and squaring
- Repeat for a pseudo-random bit stream

8. FSK

- Set up two carrier signals
- Set up a square-wave data signal
- Generate a FSK signal
- Check the spectrum of the FSK signal
- Demodulate the FSK signal using single-sided detection (filtering, rectification, filtering and squaring)
- Demodulate the FSK signal using double-sided synchronous detection (including filtering and squaring)
- Repeat for a pseudo-random bit stream

9. 16-QAM

- Set up four different pseudo-random bit streams
- Set up in-phase and quadrature carriers
- Generate a 16-QAM signal to transmit the four data signals
- Using synchronous demodulation (including filtering and squaring), recover each of the four data signals from the 16-QAM signal

Added Notes

You are welcome to vary the parameter values from those suggested in the detailed notes. However we would suggest that you consider the following guidelines.

Sampling Frequency: Make this a reasonably large multiple of the carrier frequency, particularly if you are interested in details of carrier phase. If you intend to calculate spectra, ensure that the waveform repeats itself after 4096, 8192, 16384 or 32768 sample points, as the Fast Fourier Transform will always assume that the waveform is periodic outside the analysis interval. You can't do this for a random bit stream, but should analyse at least 8192 points to make the spectral lines as near continuous as possible.

Duration of Simulation: For clear waveforms at a single frequency, regard 1024 sample points as a minimum. While a small number gives faster processing, choose a value nearer 8192, particularly if you have a wide range of frequencies (nice to have plenty of points per carrier period) and intend to use filters (which waste hundreds of points in a turn-on transient).

Signal Frequencies: Choose 1 kHz for the modulation frequency or bit rate.
For a random bit stream the number of points between bit changes is
$n \times$ repetition period $\times$ sampling frequency.
Make it an integer which is a multiple of the number of points in a carrier cycle.
Probably choose 16 kHz for the carrier frequency to ensure that the spectrum does not fold around zero frequency and that carrier cycles and modulation cycles may be viewed on the one timebase. These values also allow easy comparison with the results of others
(See also Sampling Frequency) and make it fairly easy to filter modulation frequencies from carrier frequencies.

Signal Delay: For easiest comparison with earlier practical sessions and the theory, ensure an initial phase of 90°. You can't do this with a random bit stream, and 0 is probably
more appropriate.

Duty Cycle: It is best to make each pulse last for an integer number of sample points.

Filters: If you have to filter one frequency from another, choose a cutoff frequency at the geometric mean so that the filter will pass one well and stop the other well.
To get a sharp filter response you'll need lots of calculations, so the calculation time will be long. We suggest that you make the Number of Taps about equal to the number of sample points in one period of the lowest frequency of interest; a filter to remove dc from a demodulated output is the worst case.
If in doubt, click on the Block Properties | View Response option to see if the filter is satisfactory.

Spectral Analysis: For the FFT Size choose 4k, 8k, 16k or 32k making sure, as noted earlier, that the waveform is periodic after this number of points. Note that, if the simulation runs for time T, the resolution of the spectrum will be 1 / T (e.g. if you End at 0.008 sec., the resolution will be ±62·5 Hz). The display will do a dot-to-dot picture, so that a spectral line may appear widened; look at the raw data as outlined below under Exact Data to check whether any widening of lines is real. (Remember that the spectrum is always given as a Fourier transform, never a Fourier series.)

Exact Data: If you want to get exact values from a plot of a waveform or spectrum, save the data from the plot in a file (Block Properties | Save Data to File). The file will contain a long list of x and y values, but it is a simple task to delete the many entries of little interest and turn the vital data into a table which is easily incorporated in your report.

1. Sampling and reconstructio n

Use Simulation Properties | Frequency 4096000 Hz, End at 0.001 sec.

- **Set up a sampling waveform**

Comm | Signal Sources | Rectangular Pulses
Pulse Frequency (Hz) 32000, High Level 1, Low Level 0, Duty Cycle, Duty Cycle (%) 25

- **Set up an input message**

Sinusoid ([~])
Frequency (Hz) 8000, Amplitude 1, Label Signal 8 kHz

- **Sample the message**

Multiply ([x]) the two waveforms.
Add and connect a Plot block to record the original (blue) and sampled (orange) signals.

- **Filter the sampled signal to recover the message**

Add a FIR filter to the screen. (Comment on the parameters suggested below.)
Use Number of Taps 2048, Cutoff Freq 1 15000 Hz, Filter Type Lowpass, Window Type Rectangular
Connect the sampled signal to the input and plot the output.
Perhaps start the time axis a little late to avoid the filter transient.
Perhaps also plot a constant input equal to the predicted recovered amplitude.

- **Find the spectrum of the sampled message**

Increase the simulation time to 0.008 sec.
Add a spectrum analyser to the screen. (Comm | Operators | Spectrum (Real))
Triggered, 32 k, Rectangular, kHz, dBm / Hz, 1 ohm
Add a (new) plot block to the screen to display the spectrum.
External Trigger, X-Y Plot, X-Axis 4
Label the plot and the axes.
Add an Impulse at t=0 (Comm | Signal Sources | Impulse) to trigger the spectral analysis.
Connect three outputs of the analysis block to the plot block (trigger, blue and pink).
Run a simulation and change the scales of the various plots for an appropriate display. Check up to 320 kHz and explain what you see (or don't see).
(These values have been carefully chosen to avoid certain spurious effects. To see a sample of what we avoided, try an input at 8002 Hz, or a sampling frequency of 32002 Hz or 32250 Hz, instead.)

- **Compare with the spectrum of the sampling signal (the pulses)**

Back with an 8kHz signal and 32kHz sampling, set up a second spectrum analyser
and plot to display this on the same basis as the sampled signal.

- **Repeat all this with a signal near half the sampling frequency**

Change the input frequency to 14 kHz and record the altered plots.

- **Repeat with a signal chosen to demonstrate aliasing**

Change the input frequency to 28 kHz and record the altered plots.

2. Quantisation noise and companding

Use Simulation Properties | Frequency 4096000 Hz, End at 0.001 sec.

- **Set up a signal**

Sinusoid ([~])
Frequency (Hz) 1000, Amplitude 0.1, Label Signal 0.1 V

- **Compress the original signal using µ=255**

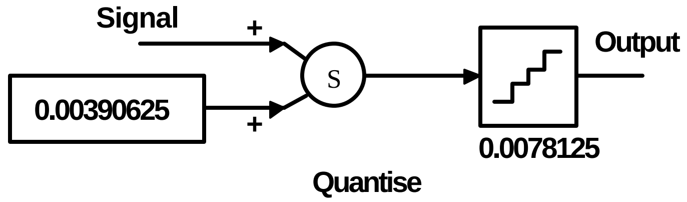
Comm | Operators | Compander
gets you a ready-to-use µ=255 compresser / expander.
Set Compander Properties | Compress, Max Value 1, µ-Law, µ Value 255.
Connect the signal to it. Display the input and output, probably on one plot.
(You may need to vary the axis scales to display various features of the signals.)

- **Expand the compressed signal and plot its error**

Get a µ=255 expander as above, except using Expand rather than Compress.
Connect the signal and plot the output.

- **Quantise the original signal to 8-bit accuracy**

One possible method is as follows. (Or try Comm | Operators | A/D Converter??)



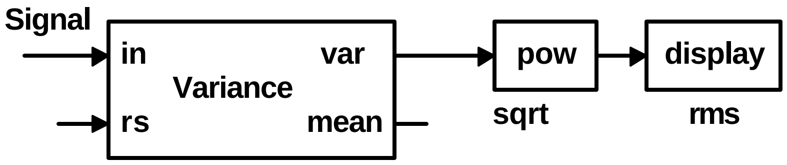
The constant ([1]) is a half-bit offset to ensure that the analogue value is rounded, not just truncated. Get the truncating quantiser with Blocks | Nonlinear | quantize and specify the Resolution shown (for 8-bit precision for ±1V signals).

- **Plot its error**

Invert ([-X]) the original signal and add this to the companded signal.

- **Determine the rms error**

This is the square root of the variance, as shown below (specify 0.5 for the power).
Use Comm | Estimators | Variance , Blocks | Arithmetic | pow and [0] .



- **Quantise the compressed signal**

- **Expand this signal and plot its error**

- **Determine the rms error**

- **Repeat for several other signal sizes**

Say 0·02, 0·05, 0·2, 0·5 and 0·95 volts. (+1 V is out of range.)

3. Eye patterns

Use Simulation Properties | Frequency 2000000000 Hz (2 GHz), End at 5 µsec.

- **Set up a square wave**

Comm | Signal Sources | Rectangular Pulses
Pulse Frequency 5000000 Hz (5 MHz), Duty Cycle 50%
High Level 1, Low Level -1

- **Put it through a lowpass filter**

Add a FIR filter to the screen. (Comment on the parameters suggested below.)
Use Number of Taps 2000, Cutoff Freq 1 30000000 Hz, Filter Type Lowpass, Window Type Rectangular
Put the square wave through the filter and plot the input and the output separately.

- **Avoiding the filter transient, plot the eye pattern**

Plot the filtered signal in another Plot block, set as follows:
X Upper Bound 0.1 MicroSeconds, Retrace Enabled, Start Time 1, End Time 5, Interval 0.1

- **Add noise to the original signal and filter the result**

Comm | Signal Sources | Noise
Set at 300000000000 Deg. Kelvin (3×10^{11}) from 50 ohms.
Click and drag a summing junction ([Σ]). Add the noise and the signal. Put this through the filter.

- **Plot the eye pattern of this filtered noisy signal**

- **Repeat for several degrees of noise and filtering**

- **Repeat all this for a pseudo-random bit stream**

Instead of the square wave, use Comm | Signal Sources | PN Sequence. Set Shift Register Size 10, Bilevel (-1,+1), Timing | Internal, Bit Rate (bps) 10000000 (10 Mbps).
Change Plot Axis Interval to 0.2 MicroSeconds.

4. AM generation and detection

Use Simulation Properties | Frequency 4096000 Hz, End at 0.002 sec.

- **Generate AM by multiplying a carrier by a suitable signal**

Set up a carrier at 16 kHz with amplitude 1; make it a cosine wave.
Set up a message signal at 1 kHz with amplitude 1; make it a cosine wave.
Add a dc constant voltage of 2 V to the message signal (constant [1], summing junction [Σ]).
Multiply this sum by the carrier ([x]); plot all waveforms.
Modify to get 100% modulation with the same carrier component and plot these waveforms.

- **Generate AM by adding a DSBSC signal to a carrier**

Use a multiplier to get the DSBSC signal, and a summing junction to add a carrier.
Use cosine waves where possible. Choose values to get 40% modulation with a peak voltage of 2·8.

- **Plot the spectrum of the AM signal**

Increase the simulation time to 0.008 sec.
Add a spectrum analyser to the screen. (Comm | Operators | Spectrum (Real))
Triggered, 32 k, Rectangular, kHz, dBm / Hz, 1 ohm
Add a (new) plot block to the screen to display the spectrum.
External Trigger, X-Y Plot, X-Axis 4
Label the plot and the axes.
Add an Impulse at t=0 (Comm | Signal Sources | Impulse) to trigger the spectral analysis.
Connect three outputs of the analysis block to the plot block (trigger, blue and pink).
Run a simulation and change the axes of the various plots for an appropriate display. Check up to 40 kHz and explain what you see (or don't see).

- **Recover the modulation signal from the AM using synchronous detection**

Multiply the AM signal by an in-phase carrier and put the output through a low-pass filter. (Comment on the parameters suggested below.)
Use Number of Taps 2048, Cutoff Freq 4000 Hz, Filter Type Lowpass, Window Type Rectangular.
Plot all waveforms, perhaps starting at 1 msec. to avoid the filter transient.
You may like to establish a fine grid, or add a negative dc voltage, or use a high-pass filter (but this is not as easy as it looks!), to measure the amplitude of the demodulated signal.
Or you may save the data as a file, and get the peak output values from the file.
Or you may use cross-wires to read off x and / or y values.

You may now like to save this version of the file, then remove a few plots, perhaps only leaving the AM signal.

- **Attempt synchronous detection using a quadrature carrier**

As above, except multiplying the AM signal by a quadrature carrier.

- **Attempt synchronous detection using an off-frequency carrier**

As above, except using a carrier at 16250 Hz.

- **Recover the modulation signal from the AM using rectification** (Blocks | Arithmetic | abs) **and filtering** (as above).

5. SSBSC generation and detection

Use Simulation Properties | Frequency 4096000 Hz, End at 0.002 sec.

- **Generate a SSBSC signal using in-phase and quadrature carrier and modulation**

Use 16 kHz for the carrier and 1 kHz for the modulation.
You'll have to watch the various phases.
Multiplication and addition are standard icons ([x], [Σ]).

- **Check the spectrum**

Increase the simulation time to 0.008 sec.
Add a spectrum analyser to the screen. (Comm | Operators | Spectrum (Real))
Triggered, 32 k, Rectangular, kHz, dBm / Hz, 1 ohm
Add a (new) plot block to the screen to display the spectrum.
External Trigger, X-Y Plot, X-Axis 4
Label the plot and the axes.
Add an Impulse at t=0 (Comm | Signal Sources | Impulse) to trigger the spectral analysis.
Connect three outputs of the analysis block to the plot block (trigger, blue and pink).
Run a simulation and change the axes of the various plots for an appropriate display. Check up to 40 kHz and explain what you see (or don't see).

- **Make a small change to generate the other sideband**

What is it?

- **Check the spectrum**

- **Recover the modulation signal from the SSBSC using synchronous detection**

Multiply the SSBSC signal by a carrier and put the output through a low-pass filter. (Comment on the parameters suggested below.)
Use Number of Taps 2048, Cutoff Freq 4000 Hz, Filter Type Lowpass, Window Type Rectangular.
Plot all waveforms, perhaps starting at 1 msec. to avoid the filter transient.
You may like to establish a fine grid, or add a negative dc voltage, or use a high-pass filter (but this is not as easy as it looks!), to measure the amplitude of the demodulated signal. Or you may save the data as a file, and get the peak output values from the file.

- **Attempt synchronous detection using a quadrature carrier**

As above, except multiplying the SSBSC signal by a quadrature carrier.

- **Attempt synchronous detection using an off-frequency carrier**

As above, except using a carrier at 16250 Hz.

6. FM generation and detection

Use Simulation Properties | Frequency 4096000 Hz, End at 0.002 sec.

- **Generate AM by adding a DSBSC signal to a carrier**

Set up a carrier at 16 kHz with amplitude 2; make it a cosine wave.
Set up a message signal at 1 kHz; make it a cosine wave.
Use a multiplier ([x]) to get the DSBSC signal, and a summing junction ([Σ]) to add a carrier. Use cosine waves where possible. Choose values to get 40% modulation with a peak voltage of 2·8 V.

- **Make a small change to produce narrowband FM**

What is it?

- **Check the amplitude modulation and phase deviation of this signal**

For example, compare it directly with the original carrier.

- **Set up a wideband FM signal**

Get an FM modulator (Comm | Modulators - Real | FM(Re)).
Suitable parameters are Translation Frequency (Hz) 16000 (the carrier frequency), Amplitude (V) 1, Initial Phase (deg) 90, FM Deviation (Hz / V) 1000.
Provide a sine-wave source at 1 kHz to modulate this carrier, and set it to produce a frequency deviation of ±2 kHz.

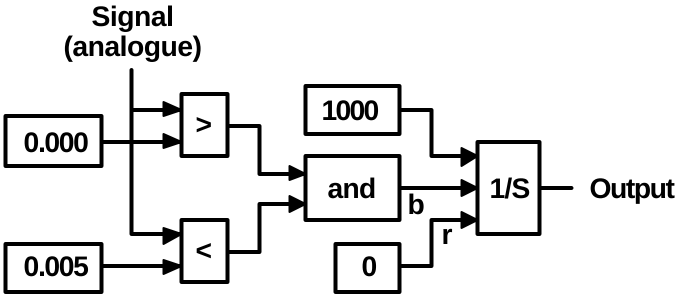
- **Check the frequency deviation**

To do this, you need to compare lots of half-periods of the signal.

The circuit below will produce a suitable timebase for this purpose. It generates a narrow pulse soon after the input (the FM signal) crosses through zero, and this triggers a sweep of 1 V / msec. until the next zero crossing of the input.
The numbered boxes are constants. [>], [<] and 'and' ([∩]) are standard icons.
The 1 / S block is an integrator (Blocks | Integration | resetIntegrator); leave its various parameters equal to 0.

Just for this section, use Simulation Properties | Frequency 40960000 Hz (40·96 MHz), End at 0.002 sec.

Plot the signal (y) against this timebase (x) and calculate the extreme frequencies from this plot.



- **Check the spectrum**

Use Simulation Properties | Frequency 4096000 Hz, End at 0.008 sec.

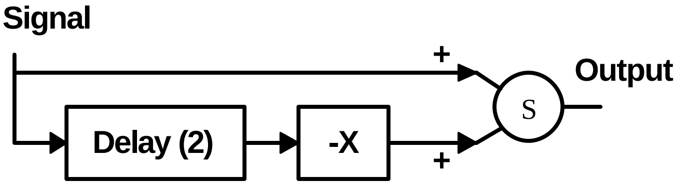
Add a spectrum analyser to the screen. (Comm | Operators | Spectrum (Real))
Triggered, 32 k, Rectangular, kHz, dBm / Hz, 1 ohm
Add a (new) plot block to the screen to display the spectrum.
External Trigger, X-Y Plot, X-Axis 4
Label the plot and the axes.
Add an Impulse at t=0 (Comm | Signal Sources | Impulse) to trigger the spectral analysis.
Connect three outputs of the analysis block to the plot block (trigger, blue and pink).
Run a simulation and change the scales of the various plots for an appropriate display. Check up to 50 kHz and explain what you see. You may like to store the data values in a file so that the precise values are available.

You may use the following values of the Bessel function in predicting the spectrum.

$J_0(2) = 0{\cdot}223891$	$J_1(2) = 0{\cdot}576725$	$J_2(2) = 0{\cdot}352834$	$J_3(2) = 0{\cdot}128943$
$J_4(2) = 0{\cdot}033996$	$J_5(2) = 0{\cdot}007040$	$J_6(2) = 0{\cdot}001202$	$J_7(2) = 0{\cdot}000175$

- **Demodulate the FM signal by differentiation, rectification and filtering**

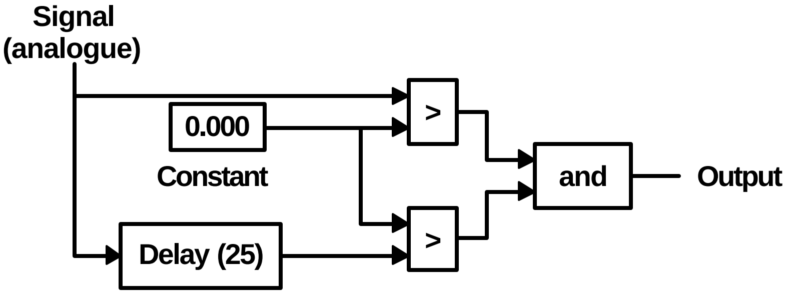
The circuit below will give a good approximation to differentiation for this purpose.
Get the delay from Comm | Operators | Delay (Real) and set the delay value to 2 SIM Steps. [-X] and [Σ] are standard icons.



The rectification is done by Blocks | Arithmetic | abs.
Use a FIR filter. (Comment on the parameters suggested below.)
Use Number of Taps 4096, Cutoff Freq 1500 Hz, Filter Type Lowpass, Window Type Rectangular.

- **Demodulate the FM signal using zero-crossing pulses**

The circuit below will produce these pulses. Use Comm | Operators | Delay (Real) .



Filter its output to recover the modulation signal.

7. OOK

Use Simulation Properties | Frequency 1024000 Hz, End at 0.032 sec.

- **Set up a carrier signal**

(Say) a cosine wave of 1 V at 16 kHz.

- **Set up a square-wave data signal**

(Say) a rectangular wave of 50% duty cycle at 500 Hz swinging between 0 and 1 V.

- **Generate an OOK signal**

Multiply the two signals.

- **Check its spectrum**

See earlier sections. Try 32k for FFT size.

- **Recover the data signal using synchronous detection (including filtering and squaring)**

Extract the modulation signal using a low-pass filter with 512 taps at (say) 8 kHz;
discuss your choice of cutoff frequency.
Squaring may use a [>] icon, with appropriate comparison value (a constant [1] block).

- **Recover the data signal using rectification, filtering and squaring**

Rectification may use Blocks | Arithmetic | abs.

- **Repeat for a pseudo-random bit stream**

Comm | Signal sources | PN sequence gets you the bit stream. Use a shift-register size of 10 bits. Set the bit interval at 1 msec. and the levels at 0 and 1 V.

8. FSK

Use Simulation Properties | Frequency 1024000 Hz, End at 0.032 sec.

- **Set up two carrier signals**

(Say) 1 V at 16 and 32 kHz.

- **Set up a square-wave data signal**

(Say) a rectangular wave of 50% duty cycle at 500 Hz swinging between 0 and 1 V.
To make it easier to replace it later with a pseudo-random data source, send the output through a wire positioner (top right; solid arrow).

- **Generate a FSK signal**

First get a complement to the data signal (levels 0,1); various methods suggest themselves.
Multiply one carrier by the data signal and the other carrier by the complement of the data signal.
To be more realistic, you may liketo filter the data signals (say, low-pass at 8 kHz) before modulation to avoid excessive bandwidth.

- **Check the spectrum of the FSK signal**

See earlier sections. Try 32k for FFT size.

- **Demodulate the FSK signal using single-sided detection (filtering, rectification, filtering and squaring)**

Use a low-pass or high pass filter at (say) 22 kHz with 1024 taps to get an OOK signal.
Rectification may use Blocks | Arithmetic | abs.
Extract the modulation signal with a low-pass filter at (say) 8 kHz with 512 taps; discuss your choice of cutoff frequency.
Squaring may use a [>] icon, with appropriate comparison value (a constant [1] block).

- **Demodulate the FSK signal using double-sided synchronous detection (including filtering and squaring)**

Use one carrier at 16 kHz and one at 32 kHz to get two outputs, theoretically complementary. Combine these two outputs to get a better signal-to-noise ratio.

- **Repeat for a pseudo-random bit stream**

Comm | Signal sources | PN sequence gets you the bit stream. Use a shift-register size of 10 bits. Set the bit interval at 1 msec. and the levels at 0 and 1 V.

9. 16-QAM

Use Simulation Properties | Frequency 1024000 Hz, End at 0.032 sec.

- **Set up four different pseudo-random bit streams**

Comm | Signal sources | PN sequence gets you a bit stream. Use a shift-register size of 10 bits. Set each bit interval at 1 msec. and use levels of ±1 V. (This is not necessarily just what you want.) Give each stream a different Initial State.

- **Set up in-phase and quadrature carriers**

(Say) 1 V at 16 kHz.

- **Generate a 16-QAM signal to transmit the four data signals**

Use multiplication ([x]) (at times by a constant and at others by a carrier) and addition ([Σ]).
To be more realistic, you may like to filter the data signals (say, low-pass at 8 kHz) before modulation to avoid excessive bandwidth.

- **Using synchronous demodulation (including filtering and squaring), recover each of the four data signals from the 16-QAM signal**

Use a filter at (say) 8 kHz with 512 taps to remove the carrier-frequency components; discuss your choice of cutoff frequency.
Squaring may use a [>] icon, with appropriate comparison values (constants [1]).
The signals with amplitude 2 are the simplest (just look for a voltage of greater than, say, +0·5), so recover them; the ones with amplitude 1 may then be derived, but you may not have time to sort them out.

00:	½(-3-1) = -2
01:	½(-3+1)= -1
10:	½(+3-1)= +1
11:	½(+3+1)=+2

Practical Sessions 11–13

ADVANCED SYSTEMS WITH TIMS

You will cover one topic in each of these three weeks. If the number of students is not too large, you will all do Session 11 in Week 11 and so on. However, if numbers rise unexpectedly, we will not have enough equipment, and the Sessions will be done in a different order for different groups.

There will not be printed practical notes along the normal lines, rather you will mostly work from 'Ideas for Experiments' compiled by the TIMS manufacturer. It is considered that this will normally be a good three hours' work, and you may well have to be selective.

Handouts provided (for the session only) in the laboratory include these instructions, specifications for all modules used, and often reading from sources other than Schwartz. Some extra notes for each session are provided below.

Your report need not contain copies of large slabs of the printed instructions,
provided that you make it quite clear what you did by direct references to those notes.

As usual, make sure that your report makes all possible qualitative and quantitative comparisons of experimental results with theory.

Practical Session 11
BIT ERROR RATES

The effect of limited bandwidth and added noise on the transmission of data.
Eye patterns as visual indicators; choosing a decision level.
Bit error rate as a function of SNR.

Reading: Lecture Notes 19, 26, F2; Schwartz 181–182, 408–410, 422–432 .
Extra reading may be found in the laboratory handout.

TIMS AMSI User Manual: 6-11, 22-24.

TIMS AMSI Ideas for Experiments: 5-13 (reference only).

After examining waveforms and eye patterns for a variety of channel characteristics, you study how the Decision Maker works. This allows you to make measurements of the bit error rate as a function of SNR and compare this with the theory. These measurements must take account of the delays, inversions and level shifts needed to ensure that the Decision Maker works correctly, and that the final output is fairly compared with the original data.

Notes on how to make these measurements are given below, and need to be followed to the letter if the results are to be valid.

Bit Error Rate as function of SNR etc.

Set up these units, from left to right;

1. Audio Oscillator
   Set Δf to give 2 kHz.

2. Sequence Generator
   Set switches for minimum sequence length (both up).

3. Tuneable LPF
   Initially set Tune clockwise, Gain top centre, Wide.

4. Baseband Channel Filters
   Initially set to 1.

5. Noise Generator
   Initially set to 0 dB.

6. Adder
   Initially set G and g to top centre.

7. Wideband True RMS Meter
   Initially set to AC 10 V.

8. Decision Maker
   Set to NRZ-L, INT.

9. Utilities

10. Error Counting Utilities
    Check the PCB switches – they should be in the default positions as follows:

J1		NORM		
Active Level	Trig	HI	]	SW1
	Gate	LO	]	
Count Mult		×1		SW2

11. Twin Pulse Generator
    Initially set Width to min., Delay to max..

12. Twin Pulse Generator
    As for 11.

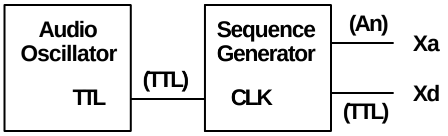
Note that the Procedure that follows tells you how to get sensible results.

It does not tell you everything you should do and observe and record.

Think of useful and informative things to do and record at each stage so that your report will clearly illustrate how the TIMS units worked, how results agree with theory, how you understand the circuit behaviour, and so on.

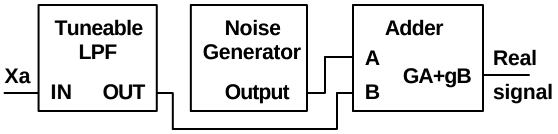
Procedure

1. Set up a random bit sequence clocked at 2 kHz.



   Trigger the CRO (ch. 3, 5V with ×10 probe) off the SYNC output of the Sequence Generator for most of the time (except for eye patterns).
   Check that some pulses of the output are only 05 msec. wide, and note that Xa and Xd are the inverse of each other (as well as having different levels).

2. Set up a sequence which simulates the result of using a real channel, i.e. limited frequency response and added noise.

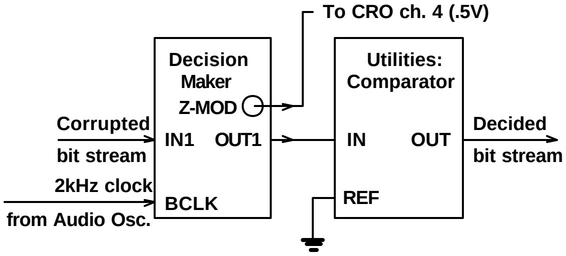


   Set g with G=0 so that the output swings ±2 V. Now set G with +20 dB of noise so that the noise swings about ±2 V about the signal.

3. Refine these settings using the True RMS Meter.
   Disconnect the noise from the Adder.
   Adjust g to give an indicated Adder output of 2·00 VRMS.
   Disconnect the bit stream from the Adder and connect the noise (+20 dB).
   Adjust G to give an indicated Adder output of 1·41 VRMS.
   Retain these G and g settings.
   With both Adder inputs connected, measure the RMS output voltage; does it agree with a value predicted from the separate signal voltages?
   (Some students get close to the correct answer by adding the two voltages and dividing the sum by the square root of 2, but this is nonsense.
   For example, what if one of the signals were zero?)
   Turn the noise down to 0 dB.

4. Now try to square up the bit stream using the Decision Maker.
   Note that we used the analogue bit stream for two reasons:
   - It is an inverted version of the digital bit stream, but this compensates for inversion in the Adder.
   - When set to NRZ-L, the Decision Maker has a threshold at 0 V, central on the waveform.

   However, the bit stream out of the Decision Maker is bipolar, so it needs to be set to TTL levels for later use.

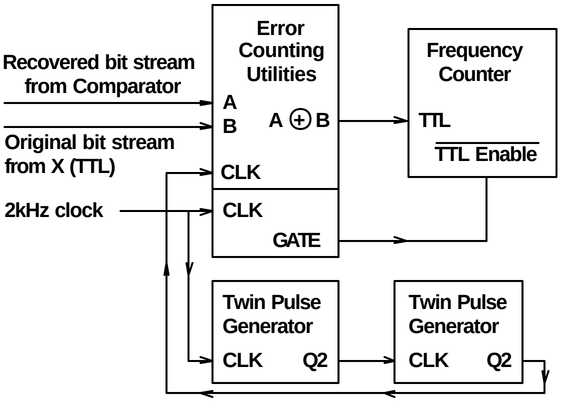


   Observe IN1, OUT1, Z-MOD on the CRO, while still triggering off the SYNC of the Sequence generator (ch. 3).
   The short pulses of Z-MOD indicate the times at which the output level is changed, depending on whether the input is then positive or negative.
   The decision should not be made while the input is changing, so adjust the Decision Point until Z-MOD pulses avoid such transitions but coincide with the centre of the shortest input pulses.
   Check that the output is now a good copy of the input; note however that the output has to be a delayed version of the input.
   You may like to try other filters; set the Tuneable LPF to its widest setting and put this signal also through the Baseband Channel Filters.
   (N.B.: 2 & 4 are swapped.)

5. Check the operation of the Decision Maker using eye patterns; simply trigger the CRO off the 2kHz clock instead of SYNC for the purpose.
   Reduce the channel bandwidth and note the deterioration in shape of the bit stream; note also that its delay also varies, so that the Decision Point needs to be varied for best results.
   Add more noise to the bit stream and note that the safe vertical eye height is reduced – in fact with 22 dB of noise the eye closes with the recommended settings.

6. Observe the behaviour of the Decision Maker on a longer timebase, observing the original (X - TTL) bit stream and the recovered bit stream while again triggering off the SYNC of the Sequence Generator. Check that some Decision Point settings produce errors in the output.

7. Use the Error Counting Utilities to count such errors.



Start with a bit stream near perfect, using the widest LPF setting and 0 dB of noise. Initially disconnect Q2 from the CLK input of the Error Counting Utilities. Observe A, B, A⊕B and note that, although one bit stream is a good copy of the other if the Decision Point is reasonable, the exclusive-OR indicates lots of errors because of their relative delay.

Observe A⊕B and the Q2 output of the second Twin Pulse Generator. With both delays at a maximum and a reasonable (if not optimum) setting of the Decision Point, the Q2 pulses should only occur when A⊕B is LOW.

This means that the Q2 pulses occur when the original and recovered waveforms are equal; however, if the Decision Maker was in error a Q2 pulse would occur when A⊕B=1.

Connect the Q2 pulse to the CLK input of the Error Counting Utilities. You should now only get pulses from A⊕B when an error really occurs.

Provided that you can keep the adjustments so that there is a time when the original and recovered waveforms are equal, and can adjust the Q2 timing to get sampling pulses then, you are in a position to make real bit-error-rate measurements.

8. Measure the bit error rate as a function of SNR.

Put the LPF to its widest setting; the Decision Point is then not too critical. Ensure, as above, that you are ready to measure real errors due to noise.

Monitor A⊕B and increase the noise (2dB steps) until you start to see only an occasional error on the CRO. Start with the noise 2 dB lower.

Check that the signal alone is 2·00 VRMS.

Set the Pulse Counter section of the Error Counting Utilities to $\times 10^5$; one measurement will therefore take about 50 seconds.

The measurement routine is:

- Check and record the signal in VRMS, by disconnecting the noise from the Adder.
- Similarly measure the noise signal in VRMS.
- Reset the Frequency Counter (set on COUNTS) (red button).
- First check that the clock is at 2 kHz (using the CRO). Press the red TRIG button of the Pulse Counter of the Error Counting Utilities to start the 10^5-pulse gate (about 50 sec.) and start counting.
  (Notes: Ignore the first pulse; it is spurious. An 'Active' light indicates that the count is proceeding. One simple check is to disconnect the A input to the ex-OR; this should give 50000 errors.)

Now measure the bit error rate as a function of SNR for successive 2dB increases in noise level.

Compare with theory.

Practical Session 12
LINE CODES

Study of various line codes for favourable properties such as: easy extraction of clock, minimal spectral width, no dc component, immunity to inversion,
error detection capability.

Reading: Lecture Notes 20; Couch 144–163; Roden 208–213; Schwartz 192, 355 .
Extra reading may be found in the laboratory handout.

TIMS AMSI User Manual: 25-31.

TIMS AMSI Ideas for Experiments: 14-16.

Record the waveforms for the various line codes for the full 32 bits of data in the sequence.
This will be easier if you use the ×10 timebase scale and decalibrate it so that the transitions of the waveforms occur on main graticules of the CRO screen; some will occur with a half-division delay.

A good scheme is to use the back of ordinary graph paper for the waveforms;
put your sheet on something white so that the grid lines are easily visible.

Make sure that you check each waveform against the stated method of generating it.
It would be best to have a record in your report of what these methods are.

When you try the inverted codes, explain your results; don't just state what they are.

Try AC coupling using a series 47nF capacitor, not the nonsense method
given in the Ideas. Explain these results also.

Practical Session 13
DELTA MODULATION

Variation of step size and clock rate – effect on overload noise and quantisation noise. Study of various methods of modulation and demodulation.

Reading: Lecture Notes 17; Schwartz 145–159 .

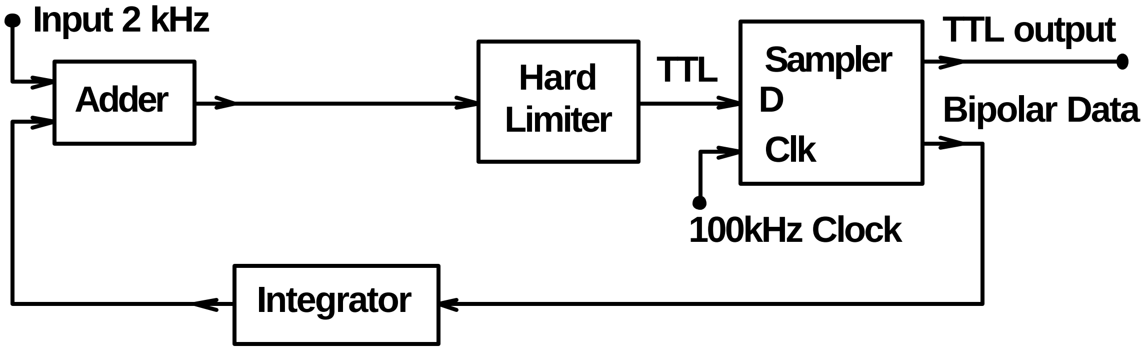
TIMS AMSI User Manual: 12-21.

TIMS AMSI Ideas for Experiments: 19-29.

It is unlikely that you will get through all of the tests suggested in the Ideas, but do as much as you can. A reasonable scheme is to go through sections 6.1, 7.1, 7.2, 6.2, 7.2, 6.3, 7.1 in that order, testing demodulation for each modulator.

The modulators and demodulators given as Ideas do not correspond exactly to those in your printed notes, so a few brief notes follow.

6.1 Simple Delta Modulator



This samples at well above the Nyquist rate.

Note that the circuit forms a feedback loop, comparing the integral of the bipolar data output with the input (in the adder, which produces an error signal). The loop attempts to make the two inputs to the adder equal (and opposite).

If the input is zero, the output will alternate rapidly, with equal times for a HIGH and a LOW value, so that the output of the integrator is zero, equal to the input.

If the input now suddenly changes to a positive value, the bipolar data output will jump to a constant state which opposes the change at the adder, and will stay there until the integrator output reaches the new input value; the output will then revert to a 50% duty cycle and the integrator will hold the new input voltage.

Note that any dc input will give the same output – a 50% duty cycle.

If the input is a ramp of voltage, the output will take on a duty cycle corresponding to that average voltage which, applied to the integrator, tracks the ramp; the faster the ramp, the higher the average voltage.

What is sent is therefore the rate of change of the input; after integration in the Delta Modulator it tracks the input voltage. The demodulation process will need to perform integration on the transmitted data stream, but the result will have an arbitrary dc level.

Note again that what is compared with the inp ut (in the adder) is the integral of the bipolar output, which should therefore be the differential of the input.

Overload noise can occur if the output is a constant 2 V (or -2 V); the resulting slope of the ramp from the integrator is the maximum slope of the input before overload occurs.

So much for the principle of this delta modulator. But what about the step size; how is it determined?

A basic integrator produces an output v_{out} from an input v_{in} according to

$$v_{out} \quad = \quad -\frac{1}{t} \int v_{in}\, dt$$

where $t = RC$ is the time constant of the R and C used in the circuit.
If the integrator has had an input of ±2 V (from the bipolar signal) for time T_S, the output will change (between sample instants) by

$$\frac{\pm 2T_S}{t}$$

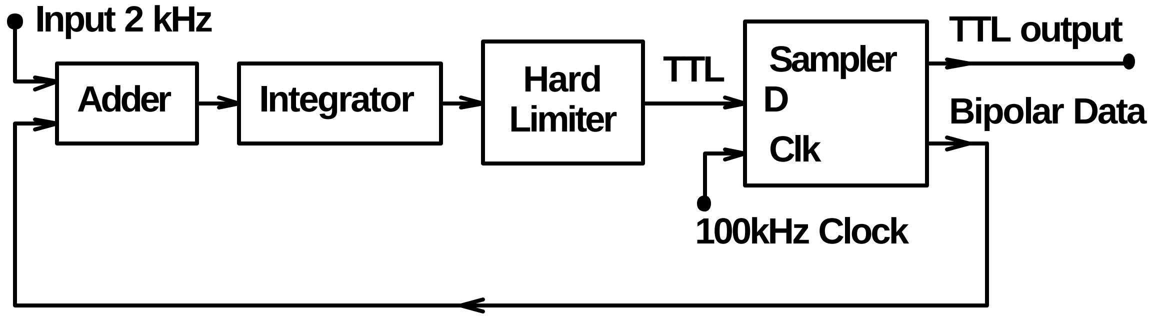
and this is clearly the step size.

The TIMS integrator has three possible values of R , giving three step sizes to choose from.

It also has three possible clock frequencies; note that increasing the time of integration also increases the step size.

In the first laboratory circuit there is an added complication − the adder has adjustable gain from each input to the output. In order to avoid overload when first it is tested, you should probably set appreciably higher gain from the integrator input than from the signal input. (As a rough guide, set the integrator gain control fully clockwise, and the message gain control about half-way.) The output should not spend appreciable time at a steady high or low voltage, but should rapidly alternate most of the time; the integrator output should not have long straight sections but should be quite jagged. This adjustment may be refined later if necessary.

6.2 Delta-Sigma Modulator



Here the integrator is moved from its previous position; the error is integrated rather than the output. (However, the error is an analogue voltage, and is not digitised.)

Note that what is compared with the input (in the adder) is the bipolar output, so that we could expect the output to represent the input directly, not its rate of change.

For a given dc input voltage, the output should alternate rapidly between HIGH and LOW, with the duty cycle adjusting so that the average voltage is equal to the input voltage. The adder output swings rapidly between two equal and opposite voltages, so that the integrator output ramps above and below the dc input voltage; the average error is zero and the output alternates rapidly between HIGH and LOW.

The demodulator need only average the bipolar data, and the frequency response reaches right down to dc.

Note that both an RC circuit and an integrator can be regarded as low-pass filters for the demodulator. However, an integrator has a $1/f$ frequency response, which an RC circuit only has beyond the cutoff frequency.