

Name: _____

Class: _____

20 - Undersampling in software defined radio

Experiment 20 - Undersampling in SDR (software defined radio)

Preliminary discussion

Software defined radio

A striking feature of the relatively short history of electronic communications is the significant improvement in performance with each innovation (usually in terms of bandwidth requirements and/or noise immunity). This has often meant that, as better communications systems have been introduced, they have quickly replaced existing technologies. For a recent example of this, consider the switch from analog to digital cell phones.

However, where the existing technology has been too well established to be abandoned, the new system has run in parallel with the old. For a long-standing example of this, consider the commercial AM and FM radio systems.

Despite the benefits of new communications techniques, the disadvantages can't be ignored. Hardware is either rendered useless or it must be duplicated. These problems have lead to the development of the latest communications concept called *software defined radio (SDR)*. SDR is a single tuner that can receive and decode any of the existing communications formats (AM, FM, DSBSC, ASK, FSK, DSSS, etc). Moreover, it's also capable of decoding any communications format that will be developed in the foreseeable future.

As its name implies, the astounding flexibility of SDR is achieved using software. Instead of implementing a hardware receiver that is necessarily band and modulation-scheme specific, SDR is a wideband receiver that converts radio signals to digital then decodes them using the software appropriate to the modulation scheme of the transmission signal. For a different modulation scheme, simply change the program. Better still, for a new modulation scheme, simply install the new program that's capable of decoding it.

Undersampling

An SDR receiver capable of receiving (and decoding) the majority of electronic communications would need to operate at frequencies up to and beyond 2.4GHz (a typical cell phone frequency). Recalling the *Nyquist Sample Rate*, you might be tempted to imagine the SDR receiver's Analog-to-Digital Converter (ADC) needing to sample cell phone signals at over 4.8GHz! However, the Nyquist requirement to sample at two or more times the highest frequency of the input signal is for avoiding aliasing of baseband signals.

Bandwidth limited signals (like radio signals in communications) don't have frequency components near DC. That being the case, the type of aliasing that the Nyquist Sample Rate attempts to avoid isn't a problem. In fact, Shannon's Information Theorem states that all of the information in a bandwidth limited signal can be captured with a sampling rate as low as twice the signal's bandwidth.

In other words, a 2.4GHz carrier signal with a 30kHz bandwidth can be sampled at a frequency as low as 60kHz and still capture all of the signal's information. That said, there are

certain sampling frequencies that will still cause aliasing and there is a mathematical process for identifying them.

Sampling of bandwidth limited signals at less than the Nyquist Sample Rate is known as *undersampling*, *band-pass sampling* and *super-Nyquist sampling*. Importantly, as well as allowing for communications signals up to very high frequencies to be sampled, undersampling has another significant advantage that makes it ideal for SDR. When the undersampling frequency is twice the signal's bandwidth, one of the sampled signal's aliases occurs at the same frequency as the original message used to modulate it. In other words, undersampling demodulates the sampled signal. All that need be done to recover the original message is to pass it through a low-pass filter to filter out the higher frequency aliases.

The experiment

In this experiment you'll use the Emona Telecoms-Trainer 101 to set up a bandwidth limited signal. You'll then use undersampling to demodulate the bandwidth limited signal and recover the message. Finally, you'll explore the effects on the recovered message of mismatches between the modulated carrier's bandwidth and the frequency used for undersampling.

It should take you about 40 minutes to complete this experiment.

Equipment

- Emona Telecoms-Trainer 101 (plus power-pack)
- Dual channel 20MHz oscilloscope
- two Emona Telecoms-Trainer 101 oscilloscope leads
- assorted Emona Telecoms-Trainer 101 patch leads

Part A - Setting up a bandwidth limited signal

To experiment with undersampling you need a bandwidth limited signal. Any of the modulation schemes can be used for this purpose, but for simplicity of wiring, we'll use a DSBSC signal. The first part of the experiment gets you to set one up.

Procedure

1. Gather a set of the equipment listed on the previous page.
2. Set up the scope per the instructions in Experiment 1. Ensure that:
 - the *Trigger Source* control is set to the *CH1* (or *INT*) position.
 - the *Mode* control is set to the *CH1* position.

3. Connect the set-up shown in Figure 1 below.

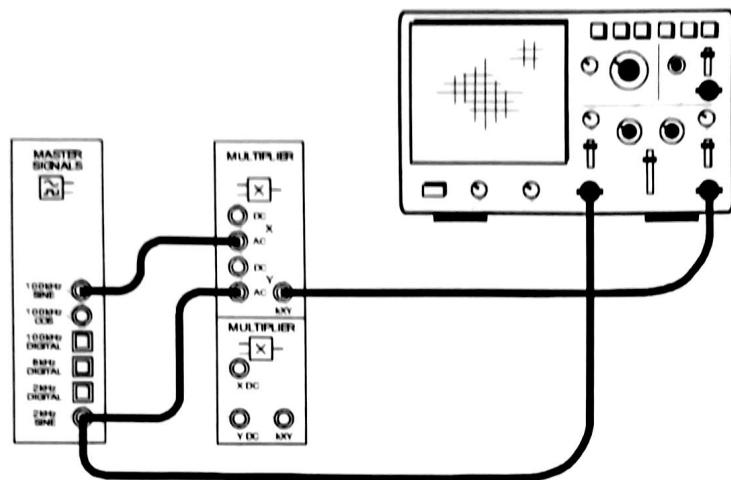


Figure 1

This set-up can be represented by the block diagram in Figure 2 below. It generates a 100kHz carrier that is DSBSC modulated by a 2kHz sinewave message.

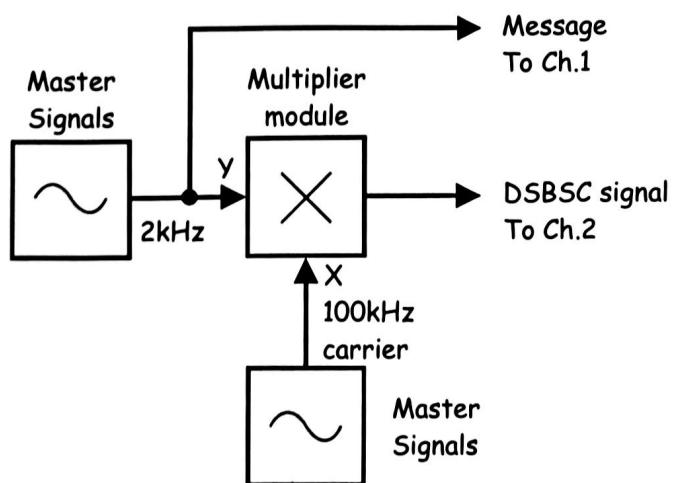


Figure 2

4. Adjust the scope's *Timebase* control to view two or so cycles of the *Master Signals* module's 2kHz *SINE* output.
5. Set the scope's *Mode* control to the *DUAL* position to view the *DSBSC* signal out of the *Multiplier* module as well as the message signal.
6. Set the scope's *Channel 1 Vertical Attenuation* control to the *1V/div* position and the *Channel 2 Vertical Attenuation* control to the *2V/div* position.

Note: The *Multiplier* module's output should be *DSBSC* signal with alternating halves of its envelope forming the same shape as the message.

Question 1

For the given inputs to the *Multiplier* module, what are the frequencies of the two sinewaves that make up the *DSBSC* signal?

Question 2

What's the bandwidth of the *DSBSC* signal?



Ask the instructor to check
your work before continuing.

Part B - Direct down-conversion using undersampling

If you have successfully completed the experiment on sampling and reconstruction (Experiment 11) you have seen that the mathematical model that defines the sampled signal is:

$$\text{Sampled signal} = \text{the sampling signal} \times \text{the message}$$

As the sampling signal is a digital signal, the expression can be rewritten as:

$$\text{Sampled signal} = (\text{DC} + \text{fundamental} + \text{harmonics}) \times \text{message}$$

When the message signal is modulated carrier like the DSBSC signal that you have set up, the expression can be rewritten as:

$$\text{Sampled signal} = (\text{DC} + \text{fundamental} + \text{harmonics}) \times (\text{LSB} + \text{USB})$$

Solving the expression (which necessarily involves trigonometry that is not shown here) gives:

- Duplicates of the LSB and USB (due to their multiplication with sampling signal's DC component)
- Aliases of the LSB and USB at frequencies equal to the sum and difference of their frequencies and the sampling signal's fundamental frequency
- Numerous other aliases of the LSB and USB at frequencies equal to the sum and difference of their frequencies and the sampling signal's harmonic frequencies

Recall that the math also proves that, where a low-pass filter is being used to reproduce the original signal by plucking its equivalent out of the sampled signal, the sampling rate must be at least twice the highest frequency in the original signal. If the sampling rate is less than this, aliasing occurs.

At first glance then, this suggests that if the DSBSC signal that you have generated is to be sampled, the sampling rate must be at least 204kHz because of the upper sideband is a 204kHz sinewave.

However, as the DSBSC signal is bandwidth limited (that is, its spectral composition doesn't extend down to DC), it's possible to sample at rates lower than 204kHz without necessarily causing aliasing. For proof, Table 1 on the next page shows some of the aliases produced by sampling the DSBSC signal at 150kHz.

Table 1

Components due to DC	Components due to fs	Components due to 2fs	Components due to 3fs
98k & 102k	Diff: 48k & 52k Sum: 248k & 252k	Diff: 198k & 202k Sum: 398k & 402k	Diff: 348k & 352k Sum: 548k & 552k

Notice that none of the aliases overlap the 98kHz and 102kHz components in the sampled signal's spectral composition. The aliases are either below or above them. So, in this instance, aliasing wouldn't occur if a band-pass filter (with sufficiently steep skirts) is used to pluck the duplicate of the original DSBSC signal out of the sampled signal. That said, aliasing is still possible by choosing a sampling rate that produces aliases at frequencies that fall inside the band-pass filter's pass-band.

Obviously, as the sampling rate decreases, so too do all of the components in the sampled signal's spectrum. It makes sense then that, if the right undersampling frequency is used, it must be possible to produce aliases centre on DC. This is crucial because it means that, when a modulated carrier is undersampled, one of its sidebands can be directly down-converted back to a baseband signal without needing to use an intermediate frequency first. All that is needed is a low-pass filter to reject the other aliases.

A more sophisticated way of understanding direct down-conversion using undersampling involves thinking of the sampling action as product detection. This is entirely appropriate to do because the math is almost identical - if you're not sure about that, compare the notes here with the notes in the preliminary discussion on product detection in Experiment 7. The difference is however, instead of multiplying the modulated carrier with a single local sinusoidal carrier, sampling involves multiplying it with dozens of sinewaves (the sampling signal's fundamental and harmonics). Importantly, as long as one of the harmonics is the same frequency as the modulated carrier, the explanation for a product detector applies equally to undersampling as a form of demodulation.

To ensure that one of the sampling signal's harmonics is the same frequency as the modulated carrier, the sampling rate must be a whole integer sub-multiple of the modulated signal's carrier frequency. That said, to avoid aliasing, the sampling rate must be at least twice the bandwidth limited signal's bandwidth.

The next part of this experiment lets you demodulate your DSBSC signal to recover the 2kHz message using undersampling instead of using a product detector.

7. Return the scope's Channel B Scale control to the 0.5V/div position.
8. Modify the set-up as shown in Figure 3 on the next page.

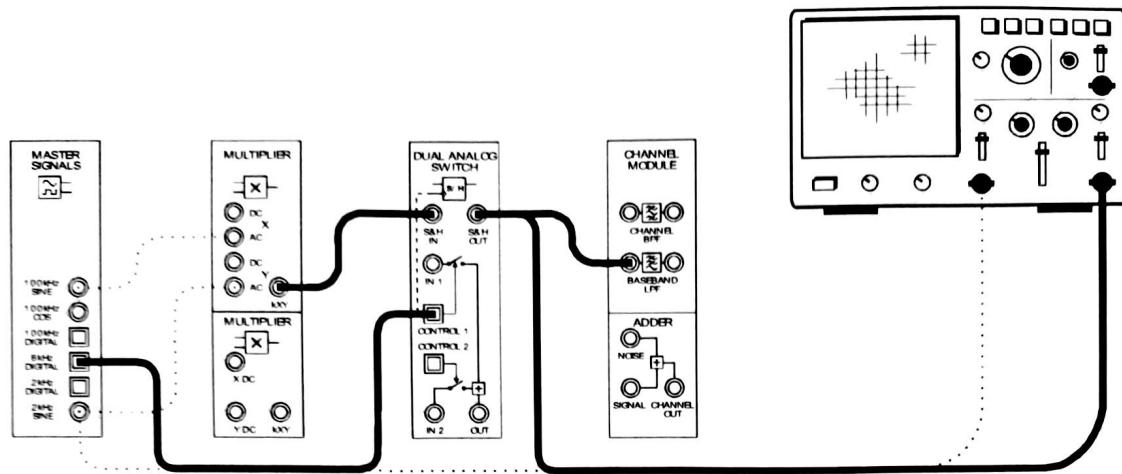


Figure 3

This set-up can be represented by the block diagram in Figure 4 below. The Multiplier module is used to generate a modulated carrier (DSBSC). The Sample-and-Hold circuit together with the Baseband LPF is used demodulate it using undersampling.

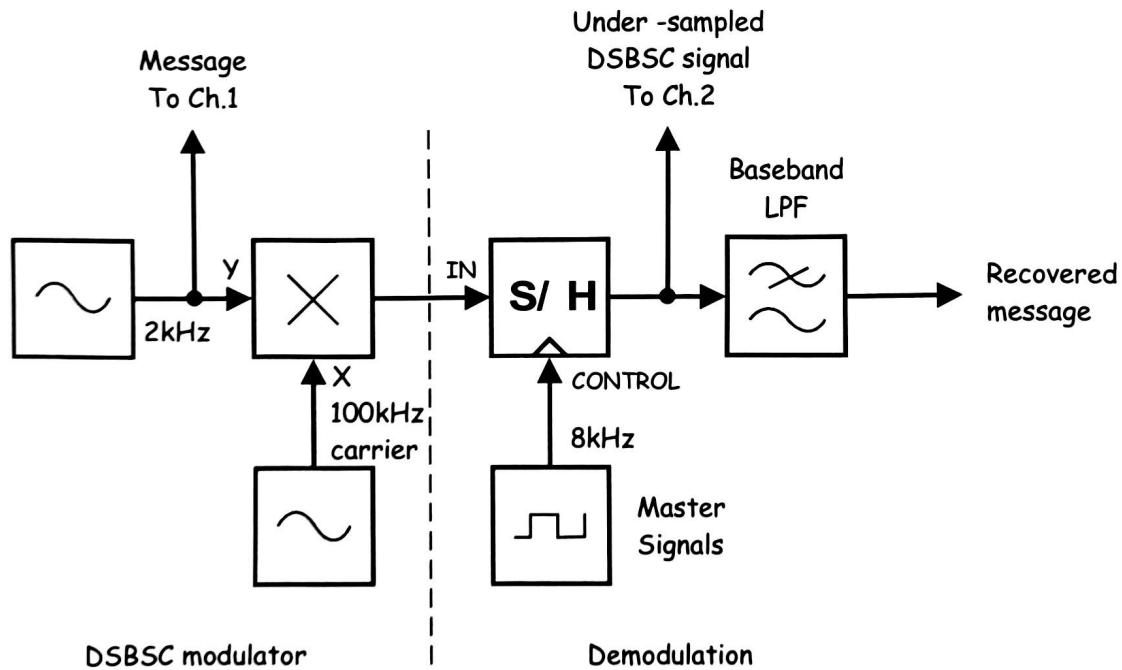


Figure 4

9. Compare the undersampled DSBSC signal with the original message.

Note: If you look closely, the undersampled DSBSC signal looks a little like an inverted version of the original message.

10. Modify the scope's Channel B connection to the set-up as shown in Figure 5 below.

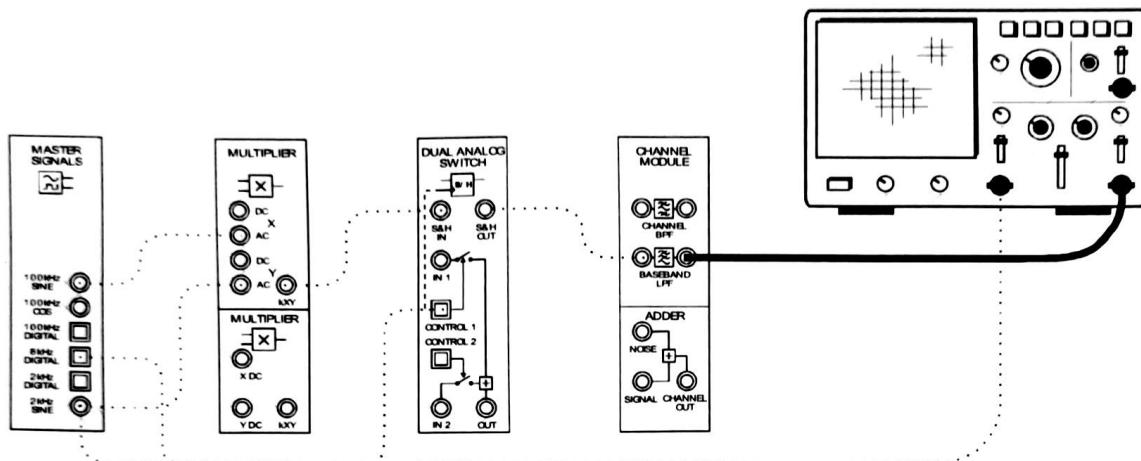


Figure 5

Question 3

What's the significance of the signal on the Baseband LPF's output?

Question 4

Given the sampling frequency is 8.333kHz (the signal's specified value of 8kHz is rounded down for simplicity), which harmonic in the sampling signal is demodulating the DSBSC signal?



Ask the instructor to check
your work before continuing.

Part C - Synchronisation

Recall that transmitter and receiver carrier synchronisation is essential to successful demodulation using product detection. If the local carrier of a product detector has even the slightest frequency or phase error (relative to the modulated carrier), the demodulated signal is affected.

Phase errors can reduce the magnitude of the recovered message and even result its complete cancellation. The effect of frequency errors depends on size. If the error is small (say 0.1Hz) the message is periodically inaudible but otherwise intelligible. If the frequency error is larger (say 5Hz) the message is reasonably intelligible but fidelity is poor. When frequency errors are large, intelligibility is seriously affected. (For a brief explanation of why these effects occur, refer to Part E in Experiment 7.)

As direct down-conversion using undersampling is a form of product detection, the sampling signal must be synchronised to the modulated carrier if these effects are to be avoided. The next part of the experiment let's you see these effects for yourself.

11. Locate the VCO module and set its *Range* control to the *LO* position.
12. Set the VCO module's *Frequency Adjust* control to about the middle of its travel.
13. Set scope's *Mode* control so that only Channel 1 is displayed.
14. Connect the scope's Channel 1 input to the VCO module's *Digital*/output.
15. Adjust the VCO module's *Digital*/output to 8.333kHz.

Note: You do this by adjusting the signal's period to 120 μ s (recall that $P = \frac{1}{f}$).

16. Return the scope's Channel 1 input to the Master Signals module's 2kHz *SINE* output.
17. Readjust the scope's controls to view two or three cycles of both the original and recovered messages.

18. Disconnect the plug to the Master Signal module's 8kHz DIGITAL output.

19. Modify the set-up as shown in Figure 6 below.

Note: Leave the scope's connections as they are.

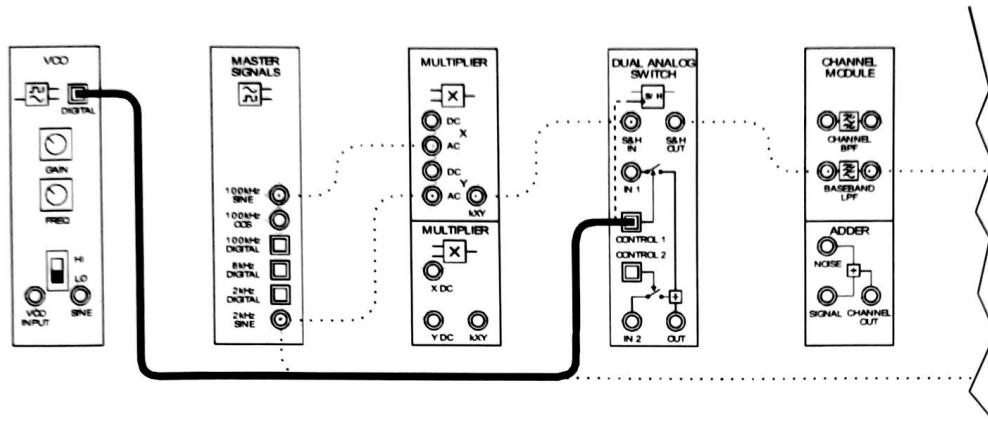


Figure 6

This modification substitutes the Master Signals module's 8kHz DIGITAL output (which is actually 8.333kHz) for an 8.333kHz digital signal from the VCO module. However, it is highly unlikely that the VCO module's output will be exactly the right frequency at exactly the right phase for successful demodulation.

20. Observe the effect of this change on the recovered message.

21. Try to correct the frequency error by turning the VCO module's Frequency Adjust control very slightly to the left and right.

Question 5

Why doesn't this solve the problem and allow the demodulator to recover the message?



Ask the instructor to check
your work before finishing.