

Lab Manual:  
Communications Principles  
  
Using the EMONA Communications board for NI ELVIS III



Lab 16: Sampling& Reconstruction

List of Updates

|  |  |
| --- | --- |
| **Date** | **Details** |
| 7/9/2018 | Completed final document |
| 10/18/2018 | Revisions and fixes |
|  |  |

**© 2018 EMONA Instruments Pty Ltd**

All Emona TIMS/ETT-Series/DxIQ user manuals, experiment manuals and supplied software   
are (C) Copyright to Emona Instruments Pty Ltd and its related entities. All rights reserved.

LIMITED AUTHORITY TO COPY TIMS MANUALS

This License Agreement grants a limited authority only to those educational institutions who have purchased the Emona TIMS/ETT/DxIQ laboratory learning equipment, to reproduce (in whole or in part,), and/or to give away copies of any of Emona Instrument’s published TIMS/ETT/DxIQ User Manuals and Experiment Manuals for the exclusive use of their own enrolled students.

No licensing fees are payable to Emona under this limited Authority.

Emona Instruments Pty Ltd retains the copyright of any edited and/or derivative documents.

**SOFTWARE**

EMONA Instruments Pty Ltd respects the intellectual property of others, and we ask our readers to do the same. This resource is protected by copyright and other intellectual property laws.

LabVIEW and National Instruments are trademarks of National Instruments.

All other trademarks or product names are the property of their respective owners.

**ADDITIONAL DISCLAIMERS**

The reader assumes all risk of use of this resource and of all information, theories, and programs contained or described in it. This resource may contain technical inaccuracies, typographical errors, other errors and omissions, and out-of-date information. Neither the author nor the publisher assumes any responsibility or liability for any errors or omissions of any kind, to update any information, or for any infringement of any patent or other intellectual property right.

Neither the author nor the publisher makes any warranties of any kind, including without limitation any warranty as to the sufficiency of the resource or of any information, theories, or programs contained or described in it, and any warranty that use of any information, theories, or programs contained or described in the resource will not infringe any patent or other intellectual property right. THIS RESOURCE IS PROVIDED “AS IS.” ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY AND ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS, ARE DISCLAIMED.

No right or license is granted by publisher or author under any patent or other intellectual property right, expressly, or by implication or estoppel.

IN NO EVENT SHALL THE PUBLISHER OR THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, COVER, ECONOMIC, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS RESOURCE OR ANY INFORMATION, THEORIES, OR PROGRAMS CONTAINED OR DESCRIBED IN IT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, AND EVEN IF CAUSED OR CONTRIBUTED TO BY THE NEGLIGENCE OF THE PUBLISHER, THE AUTHOR, OR OTHERS. Applicable law may not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Table of Contents

[Lab 16: Sampling & Reconstruction 5](#_Toc527974817)

[Learning Objectives 5](#_Toc527974818)

[Prerequisites 5](#_Toc527974819)

[Required Tools and Technology 6](#_Toc527974820)

[Expected Deliverables 6](#_Toc527974821)

[Section 1: Sampling and reconstruction 7](#_Toc527974822)

[1.1 Theory and Background 7](#_Toc527974823)

[1.2 Implement: Sampling a simple message 10](#_Toc527974824)

[Section 2: Sampling speech 14](#_Toc527974825)

[Section 3: Observations and measurements of the sampled message in the frequency domain 16](#_Toc527974826)

[Section 4: Reconstructing a sampled message 19](#_Toc527974827)

[Section 5: Aliasing 20](#_Toc527974828)

[Answer Key 26](#_Toc527974829)

[Figure 1:Sampled signal vs Sample & Hold signal 8](#_Toc527974830)

[Figure 2: Patching for sampling with the MUX module 12](#_Toc527974831)

[Figure 3: Block diagram for sampling 12](#_Toc527974832)

[Figure 4: Patching for "Sample & Hold" sampling 13](#_Toc527974833)

[Figure 5: Block diagram for "Sample & Hold" implementation 14](#_Toc527974834)

[Figure 6: Patching for sampling speech 15](file:///C:\Users\carlo\Documents\TIMS%20modules-design\DATEx-IQ\manuals\working%20sandbox\L16-Sampling\a-Lab16-Sampling_and_Reconstruction-HL_-_Sam_Edits_A1-cm1.docx#_Toc527974835)

[Figure 7: The Scope instrument front panel with both the time display (top plots) and FFT display (lower plot) enabled. 17](#_Toc527974836)

[Figure 8: Patching for sampling reconstruction 19](#_Toc527974837)

[Figure 9: Block diagram for reconstruction 20](file:///C:\Users\carlo\Documents\TIMS%20modules-design\DATEx-IQ\manuals\working%20sandbox\L16-Sampling\a-Lab16-Sampling_and_Reconstruction-HL_-_Sam_Edits_A1-cm1.docx#_Toc527974838)

[Figure 10: Patching for aliasing 22](file:///C:\Users\carlo\Documents\TIMS%20modules-design\DATEx-IQ\manuals\working%20sandbox\L16-Sampling\a-Lab16-Sampling_and_Reconstruction-HL_-_Sam_Edits_A1-cm1.docx#_Toc527974839)

[Figure 11: Block diagram for aliasing 23](file:///C:\Users\carlo\Documents\TIMS%20modules-design\DATEx-IQ\manuals\working%20sandbox\L16-Sampling\a-Lab16-Sampling_and_Reconstruction-HL_-_Sam_Edits_A1-cm1.docx#_Toc527974840)

# Lab 16: Sampling & Reconstruction

## Learning Objectives

After completing this lab, you should be able to complete the following activities.

1. Understand natural sampling and sample-and-hold approaches and what these approaches look like when applied to time domain signals.
2. Understand the frequency-domain representation of a sampled signal.
3. Understand the role of a reconstruction filter when recovering a sampled signal.
4. Understand the cause of aliasing in a sampled system and its relationship to the Nyquist Sample Rate.

## Prerequisites

You should have completed Lab 1 and Lab 2 and be familiar with the equipment, its use and the handling precautions for the equipment.

## Required Tools and Technology

|  |  |
| --- | --- |
| Platform: NI ELVIS III Instruments used in this lab:   * Oscilloscope-Time * Oscilloscope-FFT * Function and Arbitrary Waveform Generator | * Install Instruments: [http://www.ni.com/documentation/en/ni-elvis-iii/latest/getting-started/installing-the-soft-front-panel/](http://www-preview.ni.com/documentation/en/ni-elvis-iii/1.0/getting-started/installing-the-soft-front-panel/) * Access instruments <https://measurementslive.ni.com> * View User Manual <http://www.ni.com/en-us/support/model.ni-elvis-iii.html> * View tutorials <https://www.youtube.com/playlist?list=PLvcPIuVaUMIWm8ziaSxv0gwtshBA2dh_M> |
| Hardware: Emona Communications Board Components used in this lab:   * Four BNC to 2mm banana-plug leads * Assorted 2mm banana-plug patch leads * Set of headphones or earbuds | * View User Manual <http://www.ni.com/en-us/support/model.emona-communications-board-for-ni-elvis-iii.html> |

## Expected Deliverables

In this lab, you will collect the following deliverables:

* Calculations
* Data from measurements
* Observations

Your instructor may expect you complete a lab report. Refer to your instructor for specific requirements or templates.

## Section 1: Sampling and reconstruction

## Theory and Background

So far, the experiments in this manual have concentrated on communications systems that transmit analog signals. However, digital transmission is fast replacing analog in commercial communications applications. There are several reasons for this including the ability of digital signals and systems to resist interference caused by electrical noise.

Many digital transmission systems have been devised and several are considered in later experiments. Whichever one is used, where the information to be transmitted (called the *message*) is an analog signal (like speech and music), it must be converted to digital first. This involves *sampling* which requires that the analog signal’s voltage be measured at regular intervals.

Figure 1a shows a pure sinewave for the message. Beneath the message is the digital *sampling signal* used to tell the sampling circuit when to measure the message. Beneath that is the result of “naturally” sampling the message at the rate set by the sampling signal. This type of sampling is “natural” because, during the time that the analog signal is measured, any change in its voltage is measured too. For some digital systems, a changing sample is unacceptable. Figure 1b shows an alternative system where the sample’s size is fixed at the instant that the signal measured. This is known as a *sample-and-hold* scheme (and is also referred to as *pulse amplitude modulation*).



Figure 1:Sampled signal vs Sample &Hold signal

Regardless of the sampling method used, by definition it captures only pieces of the message. So, how can the sampled signal be used to recover the whole message? This question can be answered by considering the mathematical model that defines the sampled signal:

Sampled message = the sampling signal × the message

As you can see, sampling is actually the multiplication of the message with the sampling signal. And, as the sampling signal is a digital signal which is actually made up of a DC voltage and many sinewaves (the fundamental and its harmonics) the equation can be rewritten as:

Sampled message = (DC + fundamental + harmonics) × message

When the message is a simple sinewave (such as in Figure 1) the equation’s solution (which necessarily involves some trigonometry that is not shown here) tells us that the sampled signal consists of:

1. A sinewave at the same frequency as the message
2. A pair of sinewaves that are the sum and difference of the fundamental and message frequencies
3. Many other pairs of sinewaves that are the sum and difference of the sampling signals’ harmonics and the message

This ends up being a lot of sinewaves but one of them has the same frequency as the message. So, to recover the message, all that need be done is to pass the sampled signal through a low-pass filter. As its name implies, this type of filter lets lower frequency signals through but rejects higher frequency signals.

That said, for this to work correctly, there’s a small catch which is discussed in Section 5 of the experiment.

## Implement: Sampling a simple message

The ECB has a SAMPLE & HOLD module and a MUX module that have been designed for sampling. This part of the experiment lets you use the module to sample a simple message using two techniques based on these modules.

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

**Powering up the ELVIS III + EMONA Communications Board**

|  |  |
| --- | --- |
| 1. | Ensure that the NI ELVIS III Application Board power button at the top left corner of the unit is OFF (not illuminated). |

|  |  |
| --- | --- |
| 2. | Carefully plug the Emona Communications board(ECB) into the NI ELVIS III ensuring that it is fully engaged both front and back. |

|  |  |
| --- | --- |
| 3. | Ensure that you have connected the NI ELVIS III to the PC using the USB cable and that the PC is turned on. |

|  |  |
| --- | --- |
| 4. | Turn on the Application Board *Power* button by pressing it once and confirm that it is illuminated. The LEDs on the ECB should also be illuminated. If they are not, then switch the unit off immediately and check for connection or insertion errors. |

|  |  |
| --- | --- |
| 5. | Open the Instrument Launcher software in your browser and select the required instruments. |

Scope Configuration

|  |  |
| --- | --- |
| Channel Voltage range | 1 V/div |
| Horizontal Timebase | 100*µ*s/div |
| Trigger | Analog Edge, Chan 1, Rising |
| Probe Attenuation | 1x |

|  |  |
| --- | --- |
|  |  |
| 6. | Connect the set-up shown in Figure 2. Make sure that the micro-switch found under the EX-OR GATE module is set to MUX.  **Note:** Insert the black plugs of the oscilloscope leads into a ground (*GND*) socket. | |

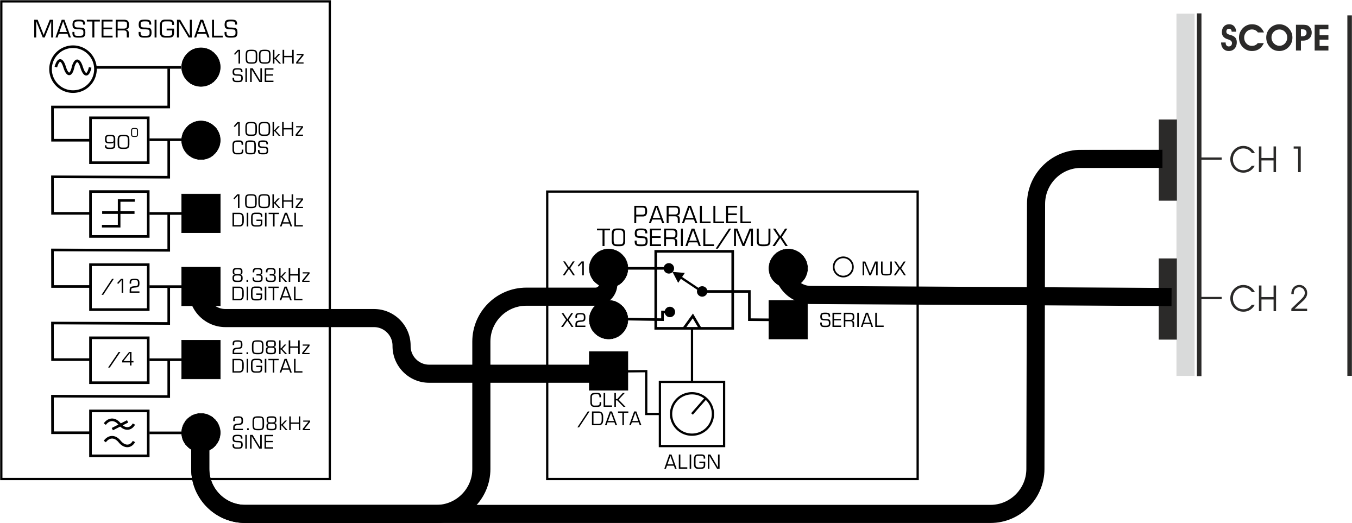


Figure 2: Patching for sampling with the MUX module

This set-up can be represented by the block diagram in Figure 3. It uses an electronically controlled switch to connect the message signal (the *2.08kHz SINE* output from the Master Signals module) to the output. The switch is opened and closed by the *8.33kHz DIGITAL* output of the Master Signals module.

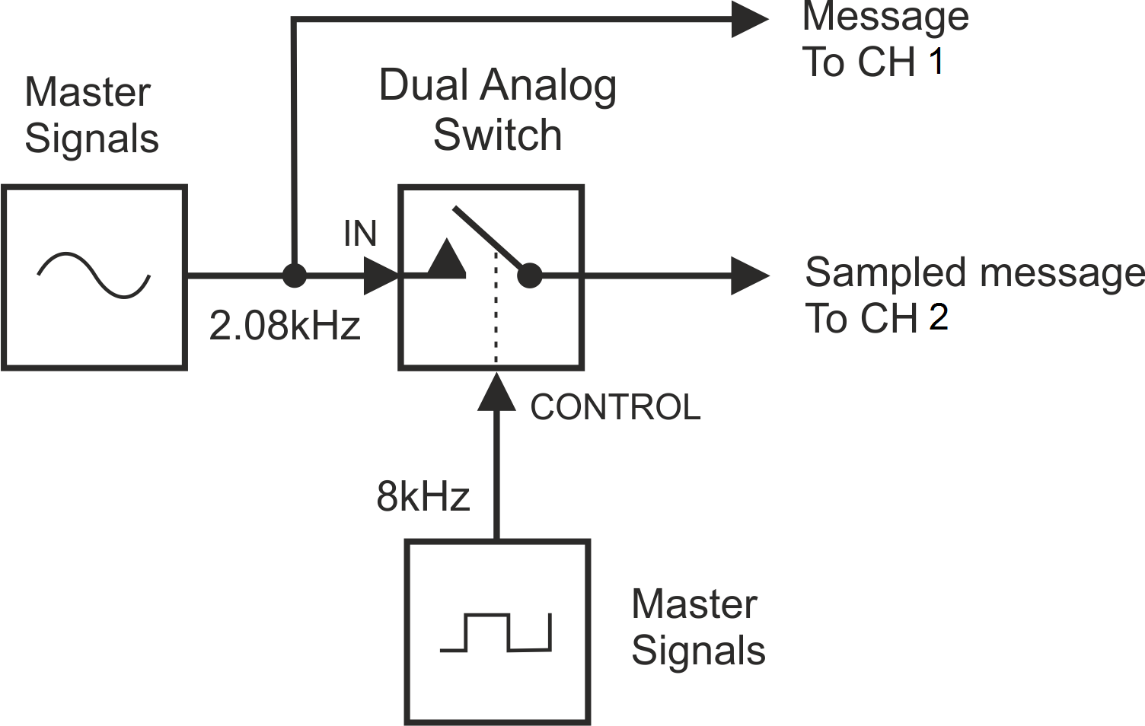


Figure 3: Block diagram for sampling

|  |  |
| --- | --- |
| 7. | Activate the scope’s Channel 1 and Channel 2 inputs to observe the message and the sampled message out of the MUX. |

|  |  |
| --- | --- |
| 8. | Use the Export Data function on the Scope to capture the two signals for your experiment report. |

1-1 What type of sampling is this an example of?

|  |
| --- |
|  |
|  |
|  |

1-2 What two features of the sampled signal confirm this?

|  |
| --- |
|  |
|  |
|  |

|  |  |
| --- | --- |
| 9. | Modify the set-up as shown in Figure 4. |

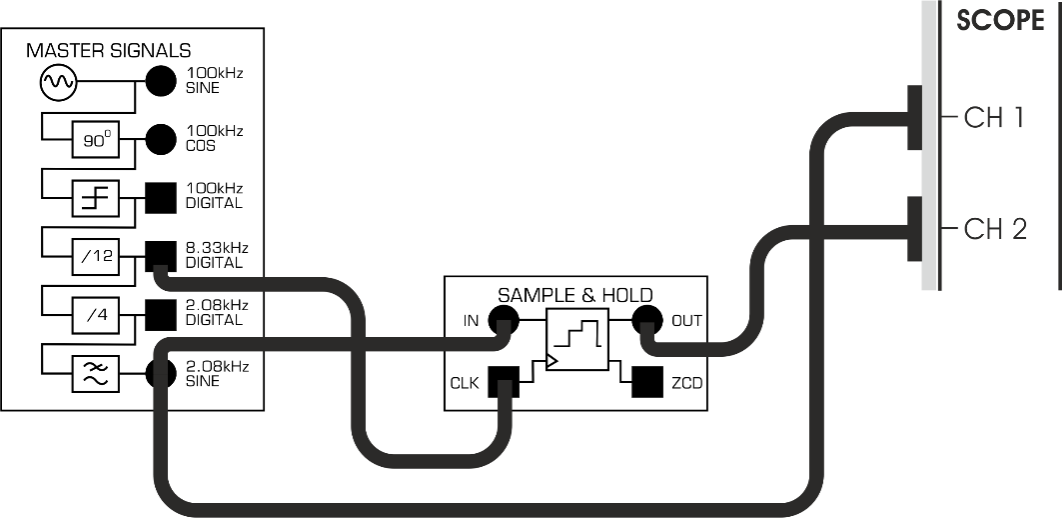


Figure 4: Patching for "Sample & Hold" sampling

This set-up can be represented by the block diagram in Figure 5. The electronically controlled switch in the original set-up has been substituted for a sample-and-hold circuit. However, the message and sampling signals remain the same (that is, a 2.08kHz sinewave and an 8.33kHz pulse train).

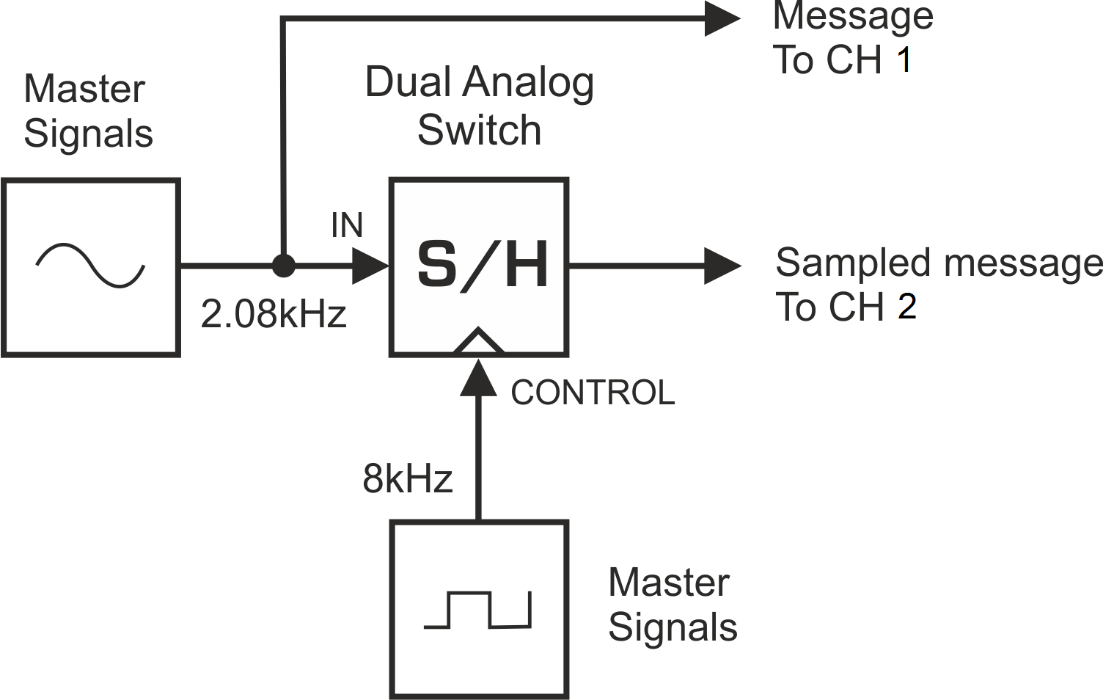


Figure 5: Block diagram for "Sample & Hold" implementation

|  |  |
| --- | --- |
| 10. | Use the Export Data function on the Scope to capture these signals and annotate in your report. |

1-3 What two features of the sampled signal confirm that the set-up models the sample-and-hold scheme?

|  |
| --- |
|  |
|  |
|  |

## Section 2: Sampling speech

The last experiment has sampled a 2.08kHz sinewave. However, the message in commercial digital communications systems is much more likely to be speech and music. The next part of the experiment lets you see what a sampled speech signal looks like.

|  |  |
| --- | --- |
| 1. | Disconnect the plugs to the Master Signals module’s *2.08kHz SINE* output. |

|  |  |
| --- | --- |
| 2. | Connect them to the Speech module’s output as shown in Figure 6.  **Remember:** Dotted lines show leads already in place. |

|  |  |
| --- | --- |
| 3. | Set the scope’s *Timebase* control to the *500µs/div* position. |

|  |  |
| --- | --- |
| 4. | Hum and talk into the microphone while watching the scope’s display. |

2-1 How well does the sampled speech replicate the incoming speech signal?

|  |
| --- |
|  |
|  |
|  |

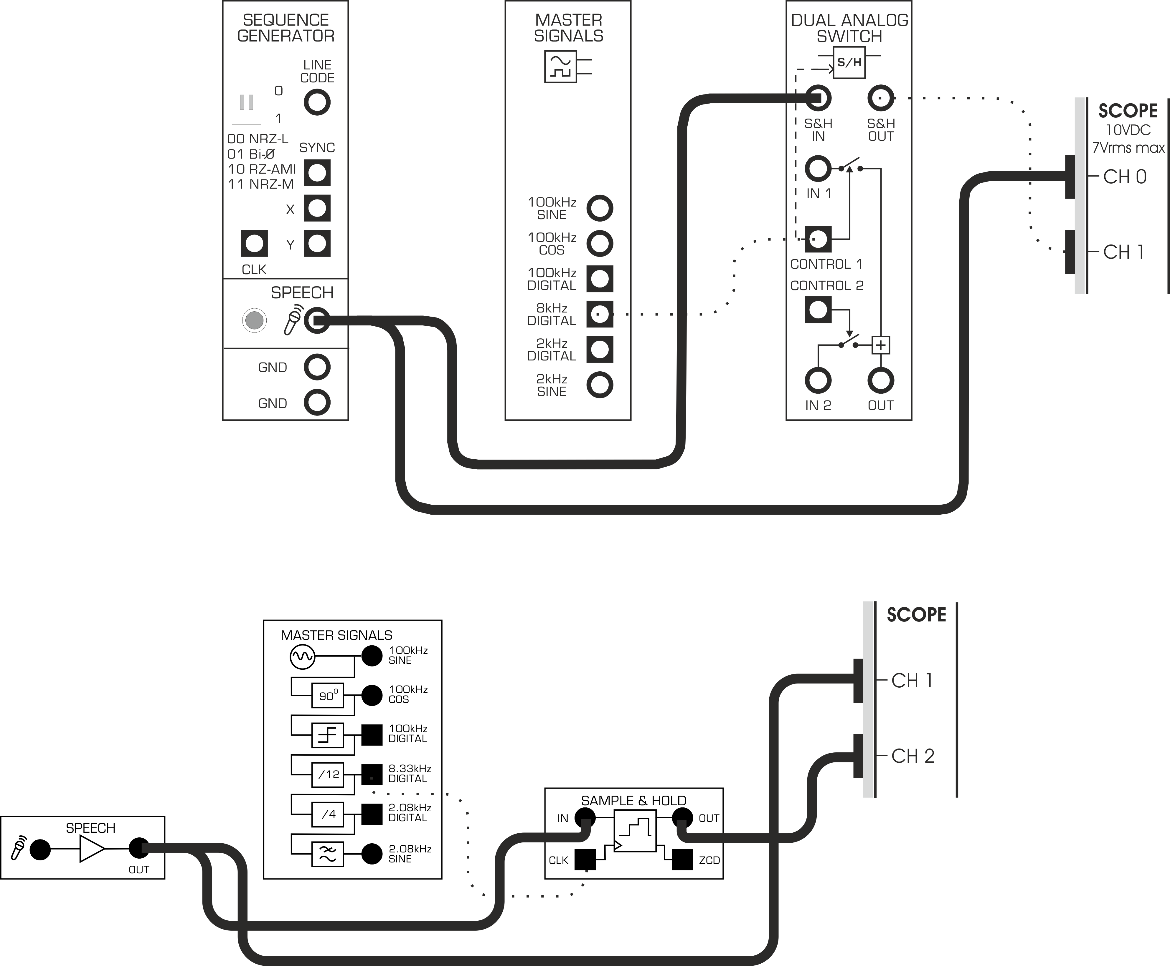


Figure 6: Patching for sampling speech

## Section 3: Observations and measurements of the sampled message in the frequency domain

Recall that the sampled message is made up of many sinewaves. Importantly, for every sinewave in the original message, there’s a sinewave in the sampled message at the same frequency. This can be proven using the FFT display on the Scope instrument. This device performs a mathematical analysis called *Fast Fourier Transform* (FFT) that allows the individual sinewaves that make up a complex waveform to be shown separately on a *frequency-domain* graph. The next part of the experiment lets you observe the sampled message in the frequency domain.

|  |  |
| --- | --- |
| 1. | Return the scope’s *Timebase* control to the *1ms/div* position and set the Volts per division to 500mV for both CH1 and CH2. |

|  |  |
| --- | --- |
| 2. | Disconnect the plugs to the Speech module’s output and reconnect them to the Master Signals module’s *2.08kHz SINE* output.  **Note:** The scope should now resemble the waveform that you drew for Step 12 on page 14. |

|  |  |
| --- | --- |
| 3. | Enable and display the FFT Channel on the Scope, set the source channel for the FFT to Channel 2 and change the FFT Window to 7 Term B Harris. |

|  |  |
| --- | --- |
| 4. | Click on the Scope*Run* control.  **Note:** If the Scope has been set up correctly, its display should look like Figure 7. |

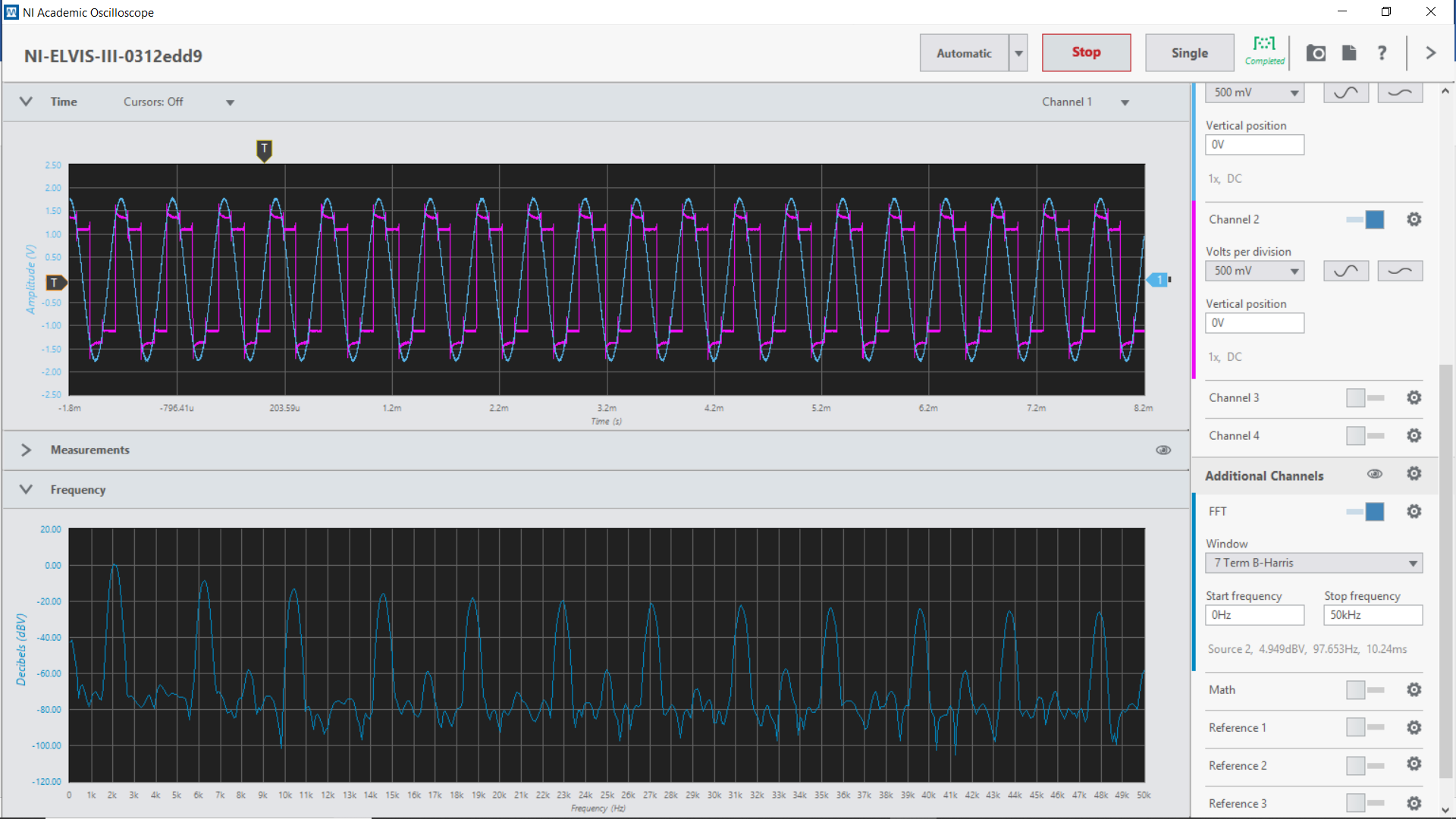


Figure 7: The Scope instrument front panel with both the time display (top plots) and FFT display (lower plot) enabled.

There are two primary displays, with the top display showing the input channels’ voltage levels as a function of time and the bottom display showing the frequency domain representation of one of the input channels.

The humps on the frequency domain plot represent the sinewaves and, as you can see, the sampled input channel consists of many of them. As an aside, these humps should just be sharp straight lines, however, the practical implementation of FFT is not as precise as the theoretical expectation.

Recall that the message signal being sampled is a 2.08kHz sinewave. This means that there should also be a 2.08kHz sinewave in the sampled message.

|  |  |
| --- | --- |
| 5. | Locate the first peak from the left on the FFT display. What frequency does it correspond to? |

As discussed earlier, the frequency of all of the sinewaves in the sampled message can be mathematically predicted. Recall that digital signals like the sampling circuit’s clock signal are made up out of a DC voltage and many sinewaves (the fundamental and harmonics). As this is a sample-and-hold sampling scheme, the digital signal functions as a series of pulses rather than a squarewave. This means that the sampled signal’s spectral composition consists of a DC voltage, a fundamental and both even and odd whole number multiples of the fundamental. For example, the 8.33kHz sampling rate of your set-up consists of a DC voltage, an 8.33kHz sinewave (fs), a 16.66kHz sinewave (2fs), a 25kHz sinewave (3fs) and so on.

The multiplication of the sampling signal’s DC component with the sinewave message gives a sinewave at the same frequency as the message and you have just located this in the sampled signal’s spectrum.

The multiplication of the sampling signal’s fundamental with the sinewave message gives a pair of sinewaves equal to the fundamental frequency plus and minus the message frequency. That is, it gives a 6.25kHz sinewave (8.33kHz – 2.08kHz) and a 10.41kHz sinewave (8.33kHz + 2.08kHz).

In addition to this, the multiplication of the sampling signal’s harmonics with the sinewave message gives pairs of sinewaves equal to the harmonics’ frequency plus and minus the message frequency. That is, the signal also consists of sinewaves at the following frequencies: 14.58kHz (16.66kHz – 2.08kHz), 18.74kHz (16.66kHz + 2.08kHz), 22.92kHz (25kHz – 2.08kHz), 26.08kHz (25kHz + 2.08kHz) and so on.

All of these sum and difference sinewaves in the sampled signal are appropriately known as *aliases*.

|  |  |
| --- | --- |
| 6. | Locate and measure the exact frequency of the sampled signal’s first six aliases. Record your measurements in Table 1.  **Tip:** Their frequencies will be close to those listed above. |

Table 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Alias 1** |  | **Alias 4** |  |
| **Alias 2** |  | **Alias 5** |  |
| **Alias 3** |  | **Alias 6** |  |

## Section 4: Reconstructing a sampled message

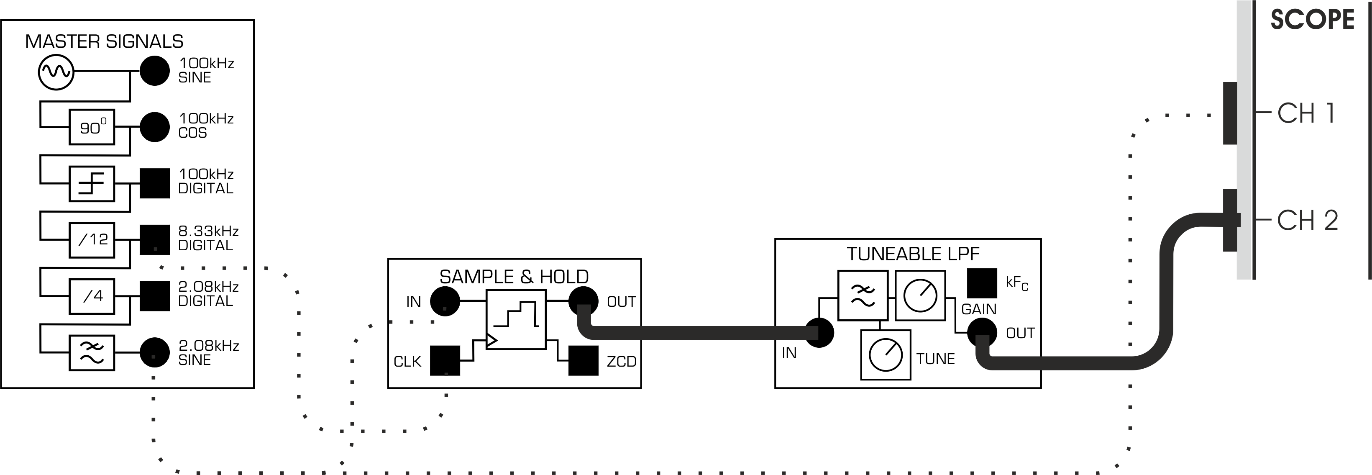
Now that you have proven that the sampled message consists of a sinewave at the original message frequency, it’s easy to understand how a low-pass filter can be used to “reconstruct” the original message. The LPF can pick-out the sinewave at the original message frequency and reject the other higher frequency sinewaves. The next part of the experiment lets you do this.

|  |  |
| --- | --- |
| 1. | Locate the Tuneable Low-pass Filter module and turn its *Gain* control to about the middle of its travel. |

|  |  |
| --- | --- |
| 2. | Turn the Tuneable Low-pass Filter module’s *Tune* control fully anti-clockwise. |

|  |  |
| --- | --- |
| 3. | Modify the set-up as shown in Figure 8. |

Figure 8: Patching for sampling reconstruction



The set-up in Figure 8 can be represented by the block diagram in Figure 9. The Tuneable Low-pass Filter module is used to recover the message. The filter is said to be “tuneable” because the point at which frequencies are rejected (called the *cut-off frequency*) is adjustable.

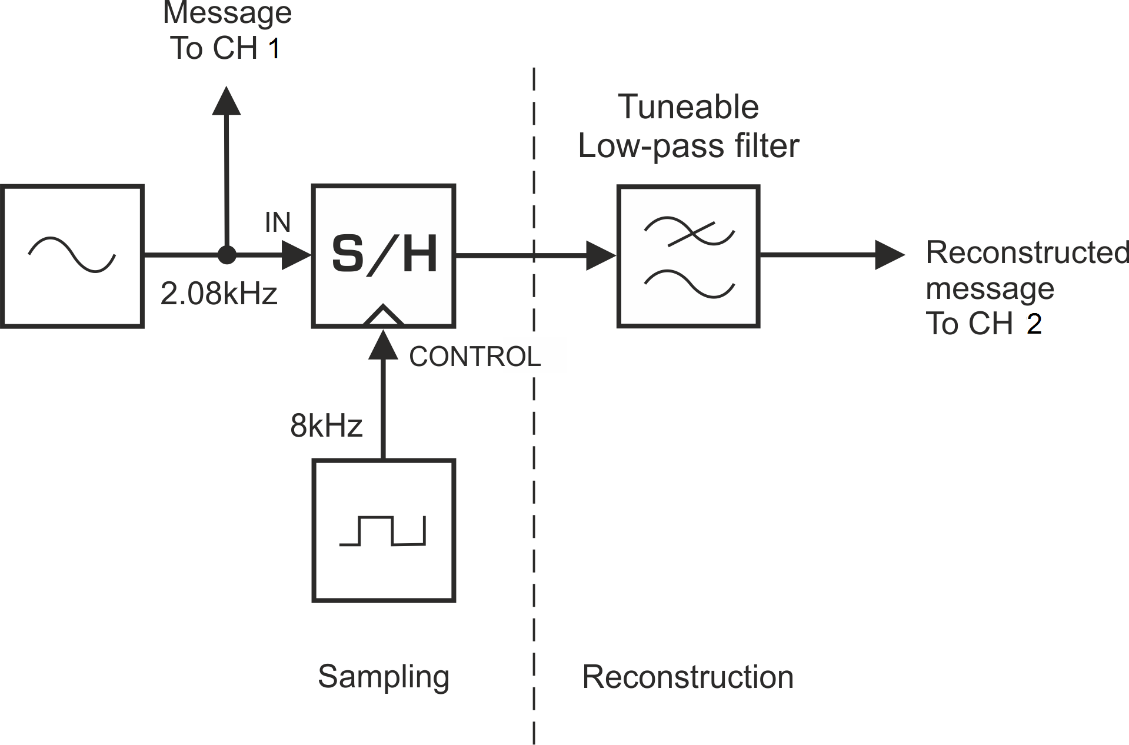


Figure 9: Block diagram for reconstruction

At this point there should be nothing out of the Tuneable Low-pass Filter module. This is because it has been set to reject almost all frequencies, even the message. However, the cut-off frequency can be increased by turning the module’s *Tune* control clockwise.

|  |  |
| --- | --- |
| 4. | Slowly turn the Tuneable Low-pass Filter module’s *Tune* control clockwise and stop when the message signal has been reconstructed and is roughly in phase with the original message. |

## Section 5: Aliasing

At present, the filter is only letting the message signal through to the output. It is comfortably rejecting all of the other sinewaves that make up the sampled message (the aliases). This is only possible because the frequency of these other sinewaves is high enough. Recall from your earlier measurements that the lowest frequency alias is 6.25kHz.

Recall also that the frequency of the aliases is set by the sampling signal’s frequency (for a given message). So, suppose the frequency of the sampling signal is lowered. A copy of the message would still be produced because that’s a function of the sampling signal’s DC component. However, the frequency of the aliases would all go down. Importantly, if the sampling signal’s frequency is low enough, one or more of the aliases pass through the filter along with the message. Obviously, this would distort the reconstructed message which is a problem known as *aliasing*.

To avoid aliasing, the sampling signal’s theoretical minimum frequency is twice the message frequency (or twice the highest frequency in the message if it contains more than one sinewave and is a baseband signal). This figure is known as the *Nyquist Sample Rate* and helps to ensure that the frequency of the non-message sinewaves in the sampled signal is higher than the message’s frequency. That said, filters aren’t perfect. Their rejection of frequencies beyond the cut-off is gradual rather than instantaneous. So in practice the sampling signal’s frequency needs to be a little higher than the Nyquist Sample Rate.

The next part of the experiment lets you vary the sampling signal’s frequency to observe aliasing.

|  |  |
| --- | --- |
| 1. | Launch and run the Function and Arbitrary Waveform Generator instrument and set it up as per the table below. These settings will configure it to generate an 8kHz TTL-level clock source that we will use as a sampling clock. |

Function and Arbitrary Generator Configuration

|  |  |
| --- | --- |
| Channel 1 | Square |
| Frequency | 8kHz |
| Amplitude | 5Vpp |
| DC Offset | 2.5V |

|  |  |
| --- | --- |
| 2. | Modify the set-up as shown in Figure 10. |

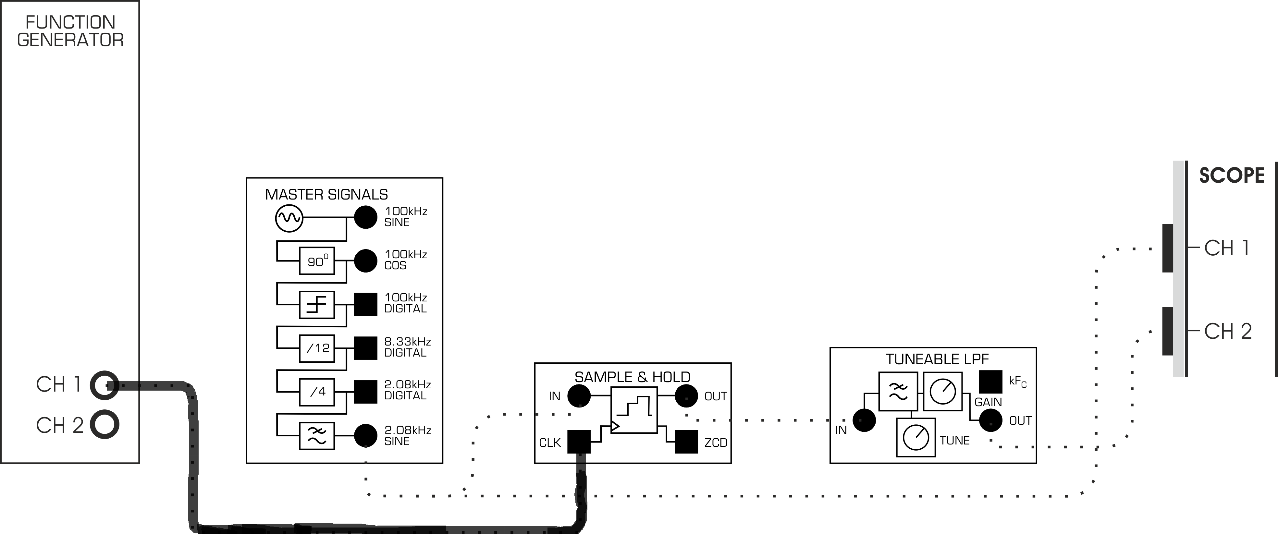


Figure 10: Patching for aliasing

This set-up can be represented by the block diagram in Figure 11. Notice that the sampling ratesignal is now provided by the Function Generator which has an adjustable frequency.

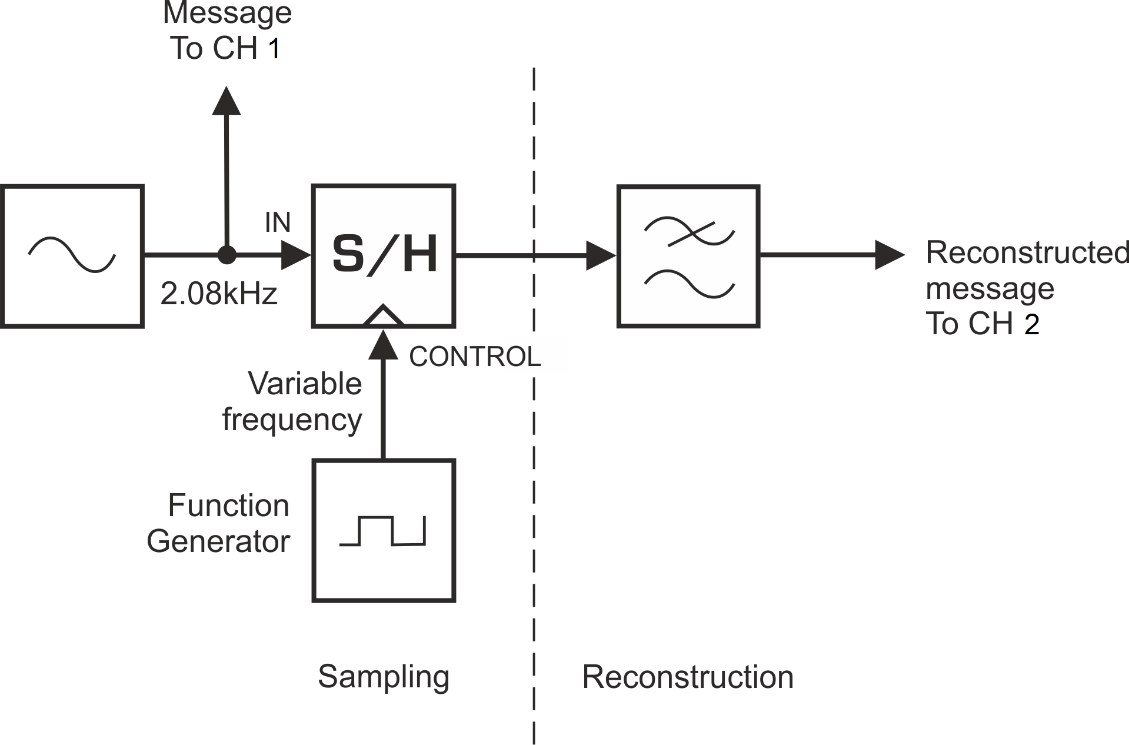


Figure 11: Block diagram for aliasing

At this point, the sampling of the message and its reconstruction should be working as before.

|  |  |
| --- | --- |
| 3. | Set the scope’s *Timebase* control to the *5ms/div* position. |

|  |  |
| --- | --- |
| 4. | Reduce the frequency of the frequency generator’s output by 1kHz and observe the effect this has (if any) on the reconstructed message signal. |

|  |  |
| --- | --- |
| 5. | Disconnect the Scope’s Channel 2 input from the Tuneable Low-pass Filter module’s output and connect it to the Sample & Hold module’s output. |

5-1 What has happened to the sampled signal’s aliases? List several frequencies that you can see peaks in the spectrum.

|  |
| --- |
|  |
|  |
|  |

|  |  |
| --- | --- |
| 6. | Return the Scope’s Channel 2 input to the Tuneable Low-pass Filter module’s output. |

|  |  |
| --- | --- |
| 7. | Repeat Steps 5 to 7 until the Function Generator’s output frequency is 3kHz. |

5-2 What’s the name of the distortion that appears when the sampling frequency is low enough?

|  |
| --- |
|  |
|  |
|  |

5-3 What happens to the sampled signal’s lowest frequency alias when the sampling rate is 4kHz?

|  |
| --- |
|  |
|  |
|  |

|  |  |
| --- | --- |
| 8. | Return the Scope’s Channel 2 input to the Tuneable Low-pass Filter module’s output. |

|  |  |
| --- | --- |
| 9. | Increase the frequency of the Function Generator’s output in 200Hz steps and stop the when the recovered message is a stable, clean copy of the original. |

|  |  |
| --- | --- |
| 10. | Record this frequency in Table 2. |

|  |  |
| --- | --- |
| Table 2 | **Frequency** |
| **Minimum sampling frequency (without aliasing)** |  |

5-4 Given the message is a 2.08kHz sinewave, what’s the theoretical minimum frequency for the sampling signal? **Tip:** If you’re not sure, see the notes on page 16-21.

|  |
| --- |
|  |
|  |
|  |

5-5 Why is the actual minimum sampling frequency to obtain a reconstructed message without aliasing distortion higher than the theoretical minimum that you calculated for Question 5-4?

|  |
| --- |
|  |
|  |
|  |

# Answer Key

1-1 What type of sampling is this an example of?

|  |
| --- |
| This is an example of “natural” sampling. |
|  |

1-2 What two features of the sampled signal confirm this?

|  |
| --- |
| 1) The sample voltages change during sampling. |
| 2) The signal’s voltage returns to zero volts between the samples. |
|  |

1-3 What two features of the sampled signal confirm that the set-up models the sample-and-hold scheme?

|  |
| --- |
| 1) The sample voltages don’t change during sampling periods. |
| 2) There’s no space between the samples. |
|  |

2-1 How well does the sampled speech replicate the incoming speech signal?

|  |
| --- |
| Reasonably well, although there is distortion of the speech signal due to the sample and hold |
| operation. The 8.33kHz sample rate only allows a small portion of the bandwidth of the signal |
| to pass through and the hold operation introduces sharp signal transitions at the sample rate. |
|  |

Table 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Alias 1** | 14.58 kHz | **Alias 4** | 26.08 kHz |
| **Alias 2** | 18.74 kHz | **Alias 5** | 31.24 kHz |
| **Alias 3** | 22.92 kHz | **Alias 6** | 35.40 kHz |

5-1 What has happened to the sampled signal’s aliases? List several frequencies that you can see peaks in the spectrum.

|  |
| --- |
| They are still present, albiet aliased to the new sampling frequency. Aliasing occur at |
| abs(N\*fs – f) where fs is the sampling freq, N is an integer and f is the signal freq, 2.08kHz. |
| At: 8kHz fs, Primary: 2.08kHz, Aliases: 5.9kHz, 10.1kHz, 13.9kHz, 18.1kHz |
| At 7kHz fs, Primary: 2.08kHz, Aliases: 4.9kHz, 9.1kHz, 11.9kHz, 16.1kHz |
| At 6kHz fs, Primary: 2.08kHz, Aliases: 3.9kHz, 8.1kHz, 9.9kHz, 14.1kHz |
| At 5kHz fs, Primary: 2.08kHz, Aliases: 2.9kHz, 7.1kHz, 7.9kHz, 12.1kHz |
| At 4kHz fs, Primary: 2.08kHz, Aliases: 1.9kHz, 5.9kHz, 6.1kHz, 9.9kHz  At 3kHz fs, Primary: 2.08kHz, Aliases: 0.9kHz, 3.9kHz, 5.1kHz, 6.9kHz |

5-2 What’s the name of the distortion that appears when the sampling frequency is low enough?

|  |
| --- |
| Aliasing. |
|  |

|  |  |
| --- | --- |
| Table 2 | **Frequency** |
| **Minimum sampling frequency (without aliasing)** | Approx 7kHz |

5-3 What happens to the sampled signal’s lowest frequency alias when the sampling rate is 4kHz?

|  |
| --- |
| The lowest alias frequency, 1.9kHz, is lower than the sampled signal’s frequency and |
| overlaps it. |
|  |

5-4 Given the message is a 2.08kHz sinewave, what’s the theoretical minimum frequency for the sampling signal? Tip: If you’re not sure, see the notes on page 16-21.

|  |
| --- |
| The thoeretical minimum sampling frequency for a 2.08kHz sinewave is 2 \* 2.08kHz = |
| 4.16kHz. |
|  |

5-5 Why is the actual minimum sampling frequency to obtain a reconstructed message without aliasing distortion higher than the theoretical minimum that you calculated for Question 5-4?

|  |
| --- |
| The actual minimum sampling frequency to obtain a reconstructed message without aliasing |
| distortion is higher than the theoretical minimum because the reconstruction filter is not |
| perfect and has a finite rolloff that allows some level of higher frequency signal components |
| through, causing distortion. |