Lesson 17 – Analog-to-Digital Conversion, Part I

Learning Outcomes

1. Understand the advantages and disadvantages of digital signals
2. Understand sampling rate in an analog–to-digital conversion system and its engineering impacts.
3. Understand voltage parameters of an analog-to-digital conversion system and their engineering impacts.
4. Understand resolution in an analog-to-digital conversion system and its engineering impacts.

(Intentionally Left Blank)

Lesson 17 - Analog to Digital Conversion (Part I)

As humans, we use *analog* senses to perceive the world. Our eyes detect subtle variations in the color and intensity of light. Our ears discriminate between a wide and continuous range of frequencies and sound levels. Our senses of taste, touch, and smell also differentiate between a continuous range of inputs.

By definition, analog values are continuous, where they can have any value between a defined maximum and minimum value. Instead of black and white, we have infinite shades of gray.

Football scores, on the other hand, are *digital*, which means they can only have certain *discrete* values. Imagine someone telling about a football game with a final score of 3 to 1. You don't have to know much about the game to realize it is not possible to only score 1 point. It just can't happen. Likewise, if the score was reported to have been 12.3 to 13.43, we wouldn't believe it. Football scores must be whole numbers.

Football scores are digital.

The game's outcome is also digital. A certain team can win, lose, tie, or the game can be canceled. No matter what happens, the outcome has to be one of a predefined and finite set of possibilities. That's what makes it digital.

One special type of digital value is called *binary*, which only allows two possible values. If the game continued until one team won and there was no chance of cancelling, then the win-lose outcome would be both digital (because there are a predefined and finite set of possibilities) and binary (because there are only two possible outcomes).

Black and white, with no gray. Computers (and many of the things that we call digital) use collections of binary values for information.

A huge amount of electrical and computer engineering is spent bridging the gap between the physical world's analog information and digital technology. Sit back and think of all the technologies which have been digitized over the last 15 years - cameras, cell phone communications, and TVs are just a few examples. Why are many technologies now digital?

- Digital is less susceptible to noise (the static that you hear in older radios and cell phones)
- Digital can be easily stored and recovered (especially when it's a collection of binary values because computers and their binary memory can be used)
- Digital allows for easier encryption and processing (mostly because of computers again).

Digital has disadvantages though which is why many technologies start off with analog implementations (did you know computers were originally analog?). When you convert your original analog information into a digital format, you lose information. There's no way to go from a continuous range to discrete levels without throwing something away. Minimizing this information loss requires you to increase your set of possible discrete values (e.g., take your digital camera from a measly 1 Mpixels to 10.2 Mpixels), but this also increases the amount of digital information that you have to save, process, and transmit (increased digital bandwidth requirements, more memory, etc.)

Today, we'll discuss the trade-offs that must be balanced in order to convert analog signals into digital data.

Sampling

The first thing we need to do to convert from analog to digital is *sample* the analog signal by taking a set number of measurements each second. The rate at which we take these measurements, or the *sampling rate*, is simply how often we take a snapshot of whatever it is we're trying to sample within a second. Think about watching a show on TV. Even though the movement on the screen looks fairly natural, it really has a sampling rate of 30 frames per second, or 30 Hz. Every 1/30 of a second, the camera takes a picture (HDTV increases the sampling rate to as high as 60 Hz, by the way).

But what if we only got 1 picture per second, would that be enough? What about 1 picture per minute?

Obviously, there is some minimum standard for how often we need to sample whatever we're trying to digitize. For video, the limit is based on how many frames per second the eye can see. For analog signals, the limit is based on the Nyquist rate, which is simply two times the highest frequency in the signal:

$$f_S = 2f_{High}$$

If we sample at a frequency below the Nyquist rate, we get a distortion in the signal known as *aliasing* that once it occurs, can't be fixed. We can correct a lot of problems but aliased data is unrecoverable!

One simple visual example of aliasing is watching a wagon wheel on a TV stagecoach. Even though the coach is obviously moving forward, the wheel sometimes looks as if it is rolling backwards. This is because the wheel is spinning at a rate higher than 15 Hz (or 15 rpm). Let's arbitrarily say the wheel spins at a rate of 20 Hz (or rpm). The required sampling rate would then need to be

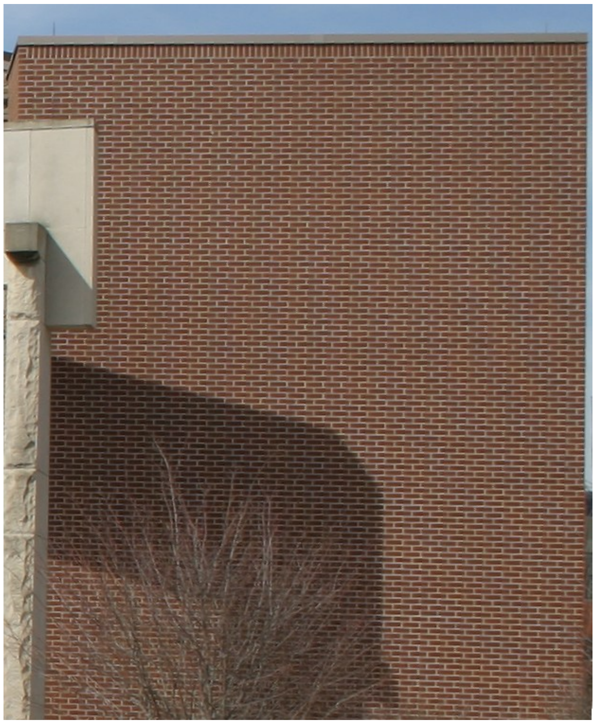
$$f_S > 2f_{High}$$

$$f_S > 2 * 20\ Hz$$

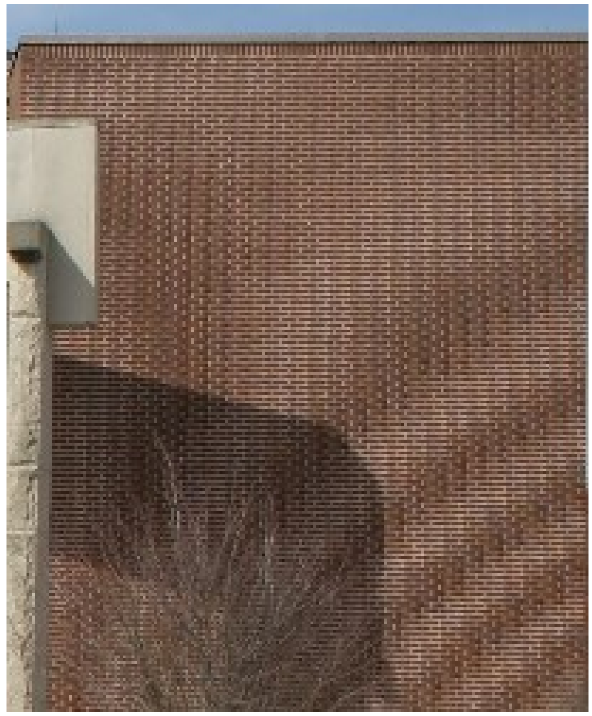
$$f_S > 40\ Hz$$

Since the actual TV sampling rate is less than 40 Hz, we get aliasing and the wheel looks like it is moving backwards. While explaining why aliasing occurs is beyond the scope of this course, it literally changes the frequency content of the signal so that it's indistinguishable from other signals (or an alias of another signal).

The pictures below show another example of aliasing. Did you realize that edges in images are a form of high frequency content? The image on the left is sampled well above the Nyquist minimum. When we reduce the number of pixels (or samples), we get aliasing in the form of the Moiré pattern in the lower right corner.



http://en.wikipedia.org/wiki/File:Moire_pattern_of_bricks.jpg



http://en.wikipedia.org/wiki/File:Moire_pattern_of_bricks_small.jpg

Last lesson, when we told you the two different bandwidths (analog and digital) were related, this is one of the fundamental reasons why. If you want to capture a signal whose highest frequency range is 40 kHz, you must sample, **at a minimum,** at 80 kHz. If you increase the bandwidth to 80 kHz, you have to take twice as many samples a second to avoid aliasing!

Example Problem: A typical human voice has a maximum frequency of 3.4 kHz. If we wish to digitize a voice signal, at what frequency must it be sampled to prevent aliasing?

Big Picture: Aliasing is the distortion in a signal when we do not sample it at a fast enough rate.

Key Issues: None.

Analysis: We must sample at a rate higher than the Nyquist rate to prevent aliasing. Therefore,

$$f_S > 2f_{High}$$

$$f_S > 2 * 3.4\ kHz$$

$$f_S > 6.8\ kHz$$

Answer: We must sample at a rate higher than 6.8 kHz to prevent aliasing.

We have to sample at a rate greater than the Nyquist rate, but what rate should we use?

A good example of something you are probably familiar with is music CDs, which are simply digital representations of analog music signals. Using filters, the original music signals are limited to frequencies lower than 15 kHz. The Nyquist rate for these signals would be:

$$f_S = 2f_{High}$$

$$f_S = 2 * 15\ kHz$$

$$f_S = 30\ kHz$$

Therefore, CDs need to have a sampling rate higher than 30 kHz. The actual sampling rate for CDs is 44.1 kHz, which is only about 1.5 times the Nyquist rate.

If 44.1 kHz is good, wouldn't 50 kHz or even 100 kHz be even better?

Yes and no. Again, we have an engineering trade off. The key is to sample at a high enough frequency to adequately reproduce the signal when needed and a low enough frequency to not exceed our memory capacity or digital bandwidth. For a CD, each sample is stored as a 16-bit binary number (2 bytes). A sampling rate of 44.1 kHz allows up to 80 minutes of uncompressed music to be loaded onto a single disc. While the quality of the music would increase slightly with higher sampling rates, it would require larger discs or reduced play time.

Before we move on, let's look quickly at what a sampled signal looks like.

Example Problem: A sinusoidal signal $V_M(t) = 2\cos(360°\ 3kt)\ V$ needs to be converted into a digital signal. What would the sampled signal look like if we used a 30 kHz sampling frequency?

Big Picture: Before we start sampling, we must *always* check the Nyquist rate to make sure we won't distort the signal. If our sampling frequency is higher than the Nyquist rate, we can start collecting samples.

Key Issues: None.

Analysis: We must sample at a rate higher than the Nyquist rate to prevent aliasing. Therefore,

$$f_S > 2f_{High}$$

$$f_S > 2 * 3\ kHz$$

$$f_S > 6\ kHz$$

Since 30 kHz is greater than 6 kHz, we can start taking samples. How would we do that? Well, we have to solve for the sample period (the time between samples) using the frequency equation:

$$t_s = \frac{1}{f_s} = \frac{1}{30\ kHz} = 33.33\ \mu\text{sec}$$

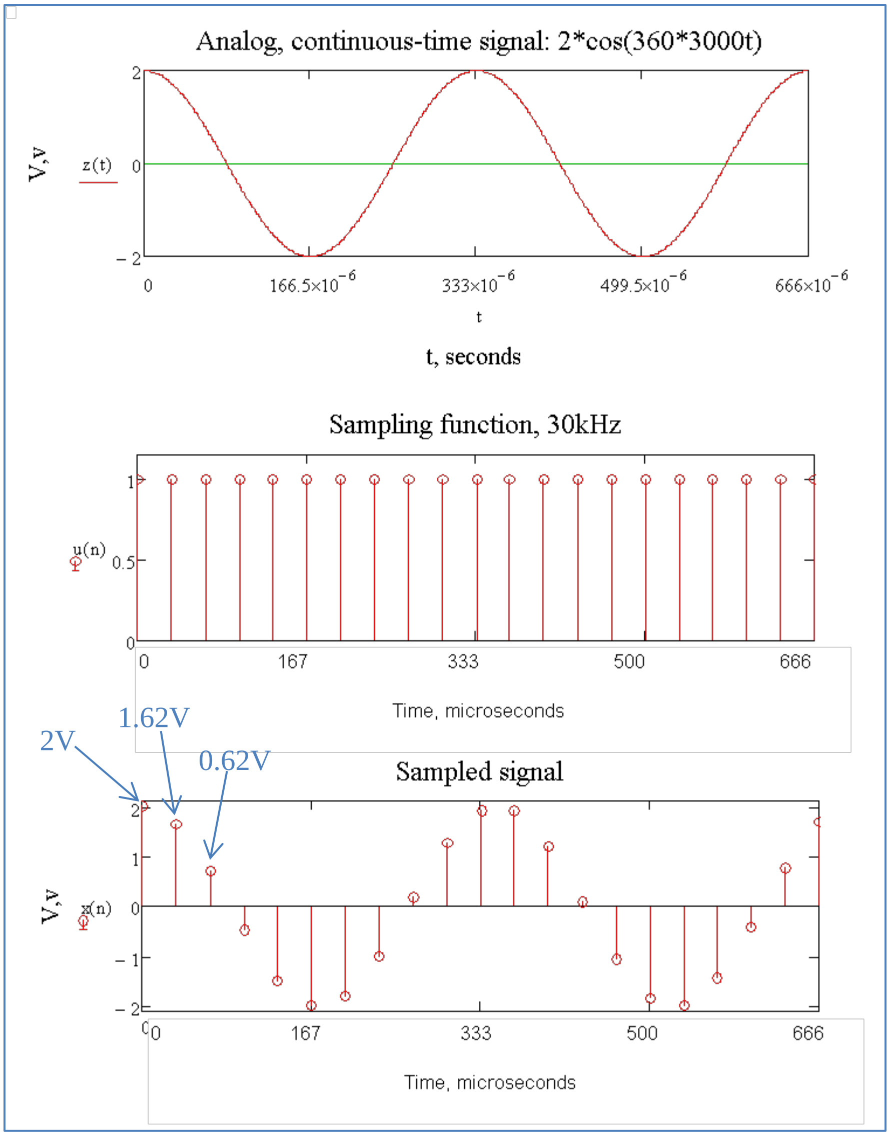
So we will take a sample every 33.33 μsec. Let's solve for the first three samples:

$$V_M(0\ sec) = 2\cos(360°\ 3k(0))\ V = 2\ V$$

$$V_M(33.33\ \mu sec) = 2\cos(360°\ 3k(33.33\ \mu sec))\ V = 1.62\ V$$

$$V_M(2\ x\ 33.33\ \mu sec) = 2\cos(360°\ 3k(66.66\ \mu sec))\ V = 0.62\ V$$

Look at the following diagram. As you can hopefully see, at 30 kHz, we've captured a fairly good representation of the sinusoidal curve we started with. It's not a perfect representation, but depending on what we use it for, it will most likely be good enough.



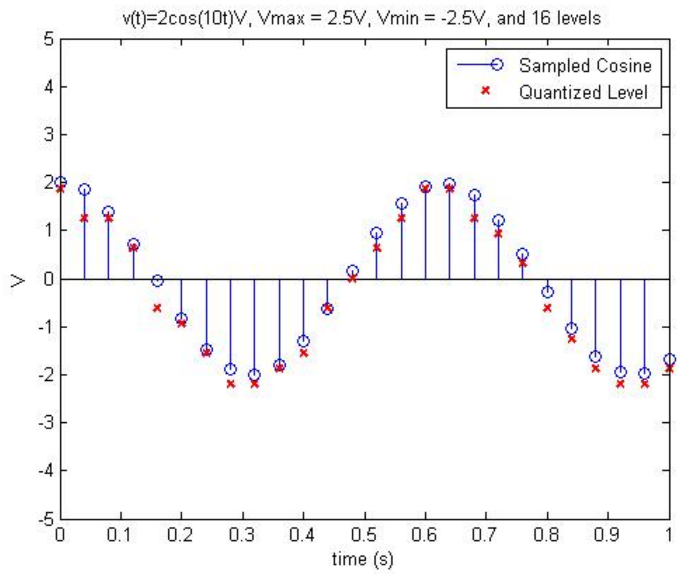
Resolution

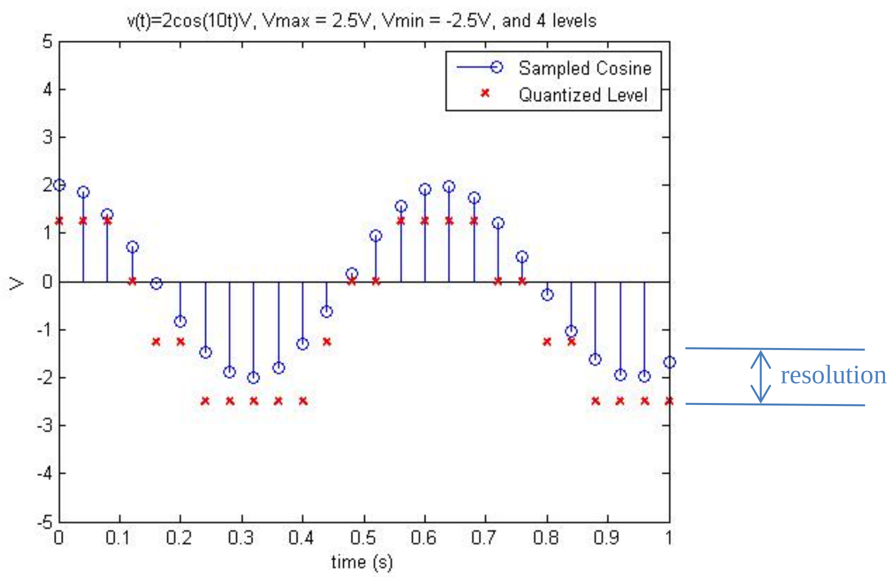
By being continuous, analog signals contain an infinite range of magnitude values. We must convert these values to a finite number of digital outputs. After an analog signal is sampled, we must then determine which output to associate with the sampled value. In order to calculate the discrete level of any given sample, we need to set voltage limits (V_{max} and V_{min}) and a number of bits, b, for the process:

- The maximum voltage, V_{max}, is the highest input voltage that will be correctly converted. Input voltages higher than V_{max} will be treated as though they were V_{max}.
- The minimum voltage, V_{min}, is the lowest input voltage that will be correctly converted. Voltages lower than V_{min} will be treated as though they were at V_{min}.
- The number of bits, b, that will be used to represent the final digital value determines that total number of possible values for the process:

$$levels = 2^b$$

Let's look at two different settings for converting $v(t) = 2\cos(10t)\,V$. In the top example, we allowed each sample to be assigned to one of 16 possible levels while in the example on the bottom, we only allowed 4 levels (easier to see the individual levels). Obviously, the levels assigned to each sample in the top graph are a truer representation of the original signal. An important point here is the sampling rate and the resolution are **completely** independent. Sampling is dividing the signal along the x-axis, while resolution involves dividing the signal along the y-axis.





Resolution is the smallest voltage change that can be measured by the analog to digital conversion process and is equal to:

$$\Delta V = \frac{V_{max} - V_{min}}{levels}$$

In the examples above, our resolutions were 312.5 mV/level (top graph) and 1.25 V/level (bottom graph).

Key Concepts: Some important points need to be emphasized before we move on:

1. Resolution is determined by the analog to digital converter (ADC), not necessarily the analog signal being converted. While we want V_{max} and V_{min} to encompass the range of our input signals, the two ranges are not the same.
2. A **smaller resolution** is a **better** resolution because you can resolve finer details from your analog signal.
3. The units for resolution are volts per level.

Example Problem: A 6-bit ADC has a V_{max} of 5 V and a V_{min} of -1 V. What is the resolution of this ADC?

Big Picture: The maximum and minimum voltages do not have to be symmetric around 0 V.

Key Issues: None.

Analysis: Using our resolution equation:

$$\Delta V = \frac{V_{max} - V_{min}}{2^b} = \frac{5\ V - (-1\ V)}{2^6\ levels} = \frac{6\ V}{64\ levels}$$

$$\Delta V = 0.09375\ \frac{V}{level} = 93.75\ \frac{mV}{level}$$

Answer: The resolution for the given ADC is 93.75 mV per level.

Example Problem: An ADC is to be used with V_{max} = 4 V and V_{min} = -2 V. If the worst acceptable resolution is 200 mV per level, what is the minimum number of bits that can be used?

Big Picture: The more bits we have the better (smaller) the resolution is.

Key Issues: If we solve the resolution equation for 2^b, we get

$$2^b = \frac{V_{max} - V_{min}}{\Delta V}$$

Analysis: If we plug in the worst case resolution, we get the minimum number of levels

$$2^b = \frac{V_{max} - V_{min}}{\Delta V} = \frac{4\ V - (-2\ V)}{200\ mV/level} = 30\ levels$$

There's two different ways to solve this for b. The easiest way is to simply list the powers of 2 until we get one higher than thirty: 2, 4, 8, 16, 32. 32 is the fifth power of 2 ($2^5 = 32$), so therefore 5 bits will give us an acceptable resolution.

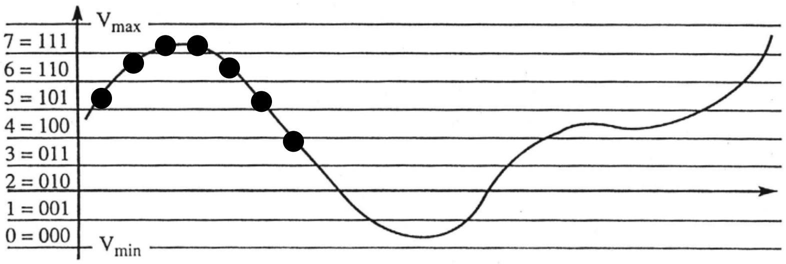
The second way is to use logarithms, where

$$b = \frac{\log(required\ levels)}{\log 2} = \frac{\log 30}{\log 2} = 4.907.$$

Since we want a better resolution than 200 mV per level, we need more than 4.907 bits. Therefore, 5 is the minimum number of allowable bits.

Answer: To get a better resolution than 200 mV per level, 5 bits are required.

So how do we convert a sample into a binary number? Look at the following diagram:



The black dots on this diagram represent 7 samples taken from an analog signal. Notice each dot falls into one of 8 levels. The first dot clearly falls into level 5, and therefore is encoded as 101. The second dot is in level 6, or 110. The next two dots are in level 7 (the highest level) and are therefore encoded as 111. The final three dots are in levels 6, 5, and 3 and are encoded as 110, 101, and 011 respectively.

The binary values resulting from these 7 samples would be:

101 110 111 111 110 101 011

While graphing might work with small graphs such as this, they are not practical for real-world systems. As mentioned earlier, a music CD uses 16 bits per sample. The number of levels would therefore be

$$levels = 2^b = 2^{16} = 65{,}536\ levels.$$

Fortunately, there is a simple equation to determine which level a given signal should be assigned (Please note that while you will not need to use this equation in this course, it is good to understand that this is how we assign sample values to specific levels):

$$Expected\ Level = E.L. = \frac{V_{in}(t) - V_{min}}{\Delta V}$$

Once you take a sample ($V_{in}(t)$), you can solve for the expected level and then ***truncate*** (or round down) the result to get an integer value which can be converted into a binary number for saving on a computer. Remember though, when you throw away the decimal part of the expected level, you are introducing error into your result.

So what is the engineering trade-off when choosing a resolution? In this case, we want to minimize the amount of information that is lost when we *quantize* our samples by assigning them an integer level. If you look at the figure above, it should be obvious that the maximum amount of signal that you can lose is a single resolution so our *maximum quantization error* is equal to:

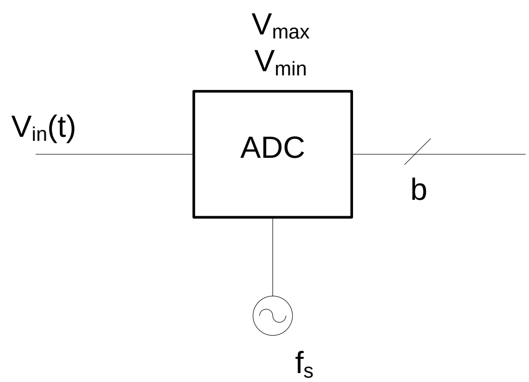
$$QE_{max} = \Delta V$$

We can lower our resolution by either 1) reducing the range of acceptable voltages (by bringing V_{max} and V_{min} closer together) or 2) increasing the number of available levels (by increasing the number of bits used to represent our values).

Since oftentimes we can't reduce the range of acceptable voltages because it would clip our input signal, the best way to minimize lost information is to increase the number of bits used for each sample. The downside though is the same as the one mentioned above: each bit added to our samples increases the amount of memory and digital bandwidth needed to save and transmit our information.

Analog to Digital Conversion

Now that we've talked about the trade-offs involved in deciding how to convert an analog signal into a digital format, let's take a moment to cover how you would actually convert your signal given a specific ADC. Note you will need to know the sampling rate, the range of acceptable voltages (V_{max} and V_{min}), and the number of bits to be used before you can convert the signal (see diagram below).

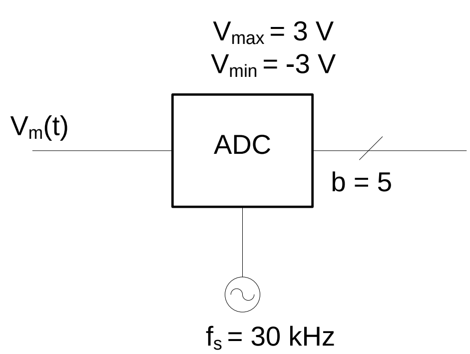


The three steps of the analog to digital conversion process are:

1. Sample: Take snapshots of the signal at the sample rate, f_s.
2. Quantize: Determine the level your sample is associated with.
3. Encode: Convert your quantized level to a binary number that is b bits long.

The easiest way to show these steps is to look at an example. Again, please take note that this example will demonstrate detailed steps of the quantization process that you will not need to do in your classwork, but should be able to understand in order to see what is really happening.

Example Problem: A sinusoidal signal $V_M(t) = 2\cos(360° \, 3kt)\, V$ needs to be converted into a digital signal using the following ADC:



Big Picture: Since we've already sampled the signal above, we now need to quantize and encode the samples.

Key Issues: None.

Analysis:

<u>Sample</u>: We found that the first three samples would be:

$$V_M(0) = 2\cos(360°\,3k(0\,sec))\,V = 2\,V$$

$$V_M(33.33\,\mu sec) = 2\cos(360°\,3k(33.33\,\mu sec))\,V = 1.62\,V$$

$$V_M(2\,x\,33.33\,\mu sec) = 2\cos(360°\,3k(66.66\,\mu sec))\,V = 0.62\,V$$

Before we can do the last two conversion steps, we need to find the ADC's resolution:

$$\Delta V = \frac{V_{max} - V_{min}}{2^b} = \frac{3\,V - (-3\,V)}{2^5\,levels} = \frac{6\,V}{32\,levels} = 187.5\,mV/level$$

<u>Quantize</u>: If we solve for our expected levels, we get:

$$E.L._0 = \frac{V_{in}(0) - V_{min}}{\Delta V} = \frac{2\,V - (-3\,V)}{187.5\,mV/level} = 26.67$$

$$E.L._1 = \frac{V_{in}(33.33\,\mu sec) - V_{min}}{\Delta V} = \frac{1.62\,V - (-3\,V)}{187.5\,mV/level} = 24.64$$

$$E.L._2 = \frac{V_{in}(2\,x\,33.33\,\mu sec) - V_{min}}{\Delta V} = \frac{0.62\,V - (-3\,V)}{187.5\,mV/level} = 19.31$$

We would truncate these values to get our quantized levels:

$$Q.L._0 = 26$$

$$Q.L._1 = 24$$

$$Q.L._2 = 19$$

<u>Encode</u>: These values need to be converted into 5-bit binary numbers:

$$Q.L._0 = 26 \rightarrow 11010$$

$$Q.L._1 = 24 \rightarrow 11000$$

$$Q.L._2 = 19 \rightarrow 10011$$

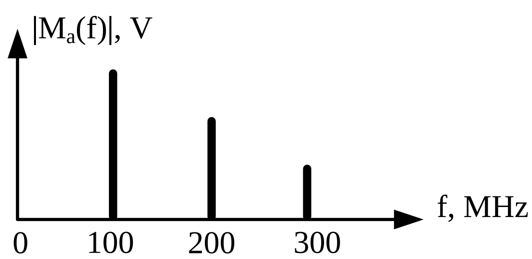
Answer: At t = 0, the ADC would output the binary number, 11010. At t = 33.33 μsec, the ADC would output the binary number, 11000. At t = 66.66 μsec, the ADC would output the binary number, 10011.

Lesson 17 Homework

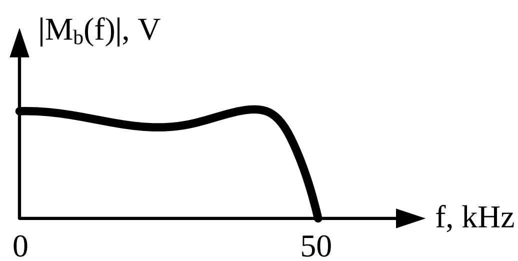
1. What is aliasing and how can it be prevented?

2. What are the three steps for converting an analog signal into digital?

3. Given the following spectra, determine the minimum sampling frequency.



a.



b.

4. Find the number of levels and resolution for an 8-bit ADC with V_{max} = 6 V and V_{min} = -4V.

5. Given a cosine input $v_{in}(t) = 1 + 3cos(360° \, 50kt)V$, answer the following questions:

   a. What is the Nyquist sampling frequency for this input signal?

   b. Which of these sampling frequencies would you use: 75 kHz, 90 kHz, or 120 kHz?

   c. What are the minimum and maximum values of the input signal?

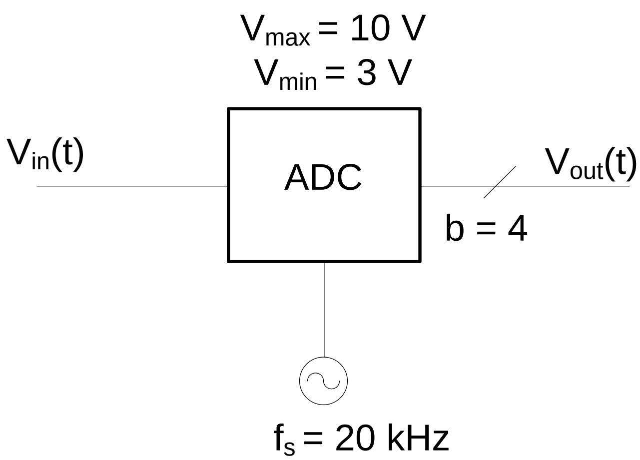
   d. Which of the following ADCs would work with this signal?

- Vmax = 5 V and Vmin = -3 V
- Vmax = 5 V and Vmin = -1 V
- Vmax = 3 V and Vmin = -3 V

   e. Using the V_{max} and V_{min} from part d, how many bits would be required to achieve a resolution of 600 mV or better?

   f. Given the number of bits calculated in part e, what is the actual resolution of the ADC?

6. An input signal, $v_{in}(t) = 6 + 5cos(360°\, 6kt) + 3cos(360°\, 15kt)V$, is to be digitized using the ADC below. Note: the 15 kHz portion of the signal is mainly noise and does not carry any useful information. Is this a good ADC for this signal? Why or why not? If not, what could you do to make this ADC work with this signal?



7. What are the advantages of digital signals?

8. T / F. Decreasing the resolution produces a more accurate digital signal.

9. T / F. Increasing the sampling rate improves resolution.

10. T / F. A smaller resolution is a better resolution.