VOL. 31 | NO. 3 | OCT. 2011

A Technical Publication for Advancing the Practice of Operating Asset Condition Monitoring, Diagnostics, and Performance Optimization

MONITOR SYSTEMS

- The 3500 Series Machinery Protection System | pg 09
- ADAPT®.wind Release 1.3 | pg 14
- What’s New with ADRE®? | pg 26
- Wireless Condition Monitoring Update | pg 31
ADRE* 408 DSPi Signal Processing

It has been a long time since analog oscilloscopes and spectrum analyzers were widely used to diagnose machinery problems. These functions have been taken over by computers – which have become increasingly powerful, and have been used more frequently as diagnostic tools. However, when using computers, an important fact has to be considered – they work in a digital world. For this reason, the analog signals coming from the transducers must be transformed into a digital format before they can be used for diagnosing machinery problems.

The term “analog” means that the signal is continuous in nature; it contains an infinite number of amplitude levels, separated by infinitesimal time intervals. A digital signal, on the other hand, is not continuous. It is formed by a finite number of amplitude levels, separated by finite time intervals. The process through which an analog signal is transformed into digital is usually referred to as “sampling”. Figure 1 shows an example of an original analog signal, while Figure 2 shows a digitized version of the same signal.

The quality of the digitalized signal depends on two factors: resolution in amplitude (vertical axis) and resolution in time (horizontal axis). Amplitude resolution is controlled by the design of the Analog to Digital Converter (ADC) of the data acquisition instrument, and time resolution is controlled by the sampling rate.

Gaston Desimone
Latin America Technical Leader
Machinery Diagnostic Services
gaston.desimone@ge.com
The ADC converts the continuously changing amplitude of the analog signal into a digital sample containing a series of discrete values, represented by binary numbers. Each of these binary numbers is formed by a certain number of bits, the basic unit of information in the digital world. The resolution of an ADC, i.e. the smallest available difference between two consecutive values of amplitude for a given full scale range, is determined by the number of bits of each digital sample, which is based on powers of two (4, 8, 16, 32, 64, etc.).

Considering that a bit can have only two states (0 or 1), a particular full scale range will be divided into $2^n$ separate amplitude levels, where $n$ is the number of bits in the sample. The following example shows how amplitude resolution increases when the number of bits is increased, for a given full scale analog signal range of 10 volts (10,000 mV). Dividing the full scale range (10 V) by the number of available discrete amplitude values (number of bits in the sample) gives us the minimum incremental change that can be captured by the ADC:

- Resolution for 8-bit A/D: $10 \text{ V} / 2^8 = 10 \text{ V} / 256 = 39.1 \text{ mV}$
- Resolution for 12-bit A/D: $10 \text{ V} / 2^{12} = 10 \text{ V} / 4096 = 2.44 \text{ mV}$
- Resolution for 16-bit A/D: $10 \text{ V} / 2^{16} = 10 \text{ V} / 65,536 = 0.153 \text{ mV}$
- Resolution for 24-bit A/D: $10 \text{ V} / 2^{24} = 10 \text{ V} / 16,777,216 = 0.000596 \text{ mV}$

The higher the number of bits in the sample, the higher will be the resolution that is provided by the ADC. The tradeoff of using digital samples with more bits is that they require more storage space (memory) and processing time. It is important to observe that the number of bits in each instantaneous value captured by the ADC is not user-configurable, as it is based on the design of the data acquisition device. For the 408 Digital Signal Processing instrument (DSPI), the number of bits in each instantaneous value sampled by the ADC is 24.

The second parameter affecting the quality of the sampled signal is the sample rate, which controls the separation between two consecutive samples in time. Unlike the number of bits processed by the ADC, the sample rate is highly configurable. As such, a certain level of knowledge is required to select appropriate sampling parameters, so that important information can be captured and not lost. We will discuss this parameter in more detail in the next section of this article.

**Signal Sampling in the 408 DSPI**

Why should we be concerned about carefully configuring our 408 DSPI unit before collecting vibration data? Some valid reasons are listed here:

- The quality of the collected vibration data affects the quality of the vibration analysis that is performed using the data.
- When vibration data has not been appropriately sampled, even the best analysis may have no value at all.
- It is important to know that every acquisition instrument has limitations, and to know what these limitations are for the particular instrument being used.
- In most cases, opportunities for data acquisition are limited due to machine availability. The sampling settings must be correct the first time – otherwise, there may not be another opportunity to collect important data.

Because this new piece of equipment is far more versatile than its predecessor, the Data Acquisition Interface Unit (DAIU) 208P, the configuration of sampling parameters has become more complex than it used to be. This requires the user to be quite knowledgeable on signal processing concepts.

**Synchronous & Asynchronous Sampling**

Now that we have discussed amplitude resolution, we will move on to sample rate, which is one of the most important configuration parameters that is available with the 408 DSPI. Two types of sampling are available on the 408 DSPI: Synchronous and Asynchronous. Several important differences exist between the two, so we will
explain them separately. However, before getting into more detail on sampling, let us talk briefly about how data is actually stored in the 408 DSPI hard drives.

The process of digitalizing an analog signal in the 408 is not continuous – not only because the analog signal is divided into discrete parts but also because these parts are grouped in packages, called waveform records, or waveform samples (Figure 3). Typically, the number of instantaneous values included in these packages may vary between 256 and 16384. Both the specific number of instantaneous samples and the way in which these waveform files are filled up with data, depends on whether the sampling process is Synchronous or Asynchronous.

Before getting into more detail on these two types of sampling, it is important to make a clarification. Throughout this article, we will be dealing with the process of digitalizing a signal, both synchronously and asynchronously. However, it is not the purpose of this article to describe the data collection settings of the 408, which control when the samples are taken (sampling events such as delta RPM or delta time, triggering events, the ratio of static samples to waveform samples to be collected, etc.).

Application Example

In this example, we will consider using a synchronous sample rate of 256 samples per revolution (256/8). This parameter will be used in two different machines – one running at 1800 rpm, and the other running at 6000 rpm. In both cases, the data acquisition device will collect 256 samples during each of the required 8 revolutions, in order to fill up a waveform record, or “package” with the required 2048 digital values.

Though the synchronous sample rate is the same in both cases (256 samples per rev), the time required to complete one revolution is very different for the two machines. In our example, the rotor will take 0.033 s to complete one revolution for the machine running at 1800 rpm and only 0.01 s for the machine running at 6000 rpm. A direct consequence is that the time interval between two consecutive samples (Figure 5) will differ from one machine to the other.
**Figure 4A**: Timebase plot generated for a synchronous waveform.

**Figure 4B**: A closer view of the plot header in Figure 4 shows various pieces of important information for the plot. "SMPR: 128/16" means that the sample rate is 128 samples per revolution, collected over a total of 16 revolutions. (128x16=2048). The "Direct" (unfiltered) amplitude value in the upper right corner was derived by a peak-to-peak measurement of the digital waveform.

**Figure 5**: For both machines, each full revolution will include 256 evenly-spaced digital samples (represented by the vertical lines in this illustration. However, the time intervals will be different because of the difference in machine speed.
1800 RPM EXAMPLE
One full revolution (0.033 s) is split into 256 equal parts. The time interval between samples will therefore be 0.033 s / 256 = 0.00013 s. To find the sampling frequency, we divide 1 by the sample interval. 1 / 0.00013 s = 7692 samples per second.

6000 RPM EXAMPLE
One full revolution (0.01 s) is split into 256 equal parts. The time interval between samples will therefore be 0.01 s / 256 = 0.000039 s. To find the sampling frequency, we divide 1 by the sample interval.
1 / 0.000039 s = 25641 samples per second.

From Figure 5, it is evident that the sampling frequency, measured in samples per second [Hz], will depend upon running speed of the machine. The faster the machine runs, the higher will be the sampling frequency used by the 408, in order to satisfy the selected synchronous sampling rate (128/16, 256/8, etc.). Also, the higher the synchronous sampling rate, the more accurately the digitized waveform record will represent the original analog signal.

However, it is important to realize that there is an upper limit on how fast the digital samples can physically be collected. In the 408, the maximum frequency for synchronous sampling is 32 kHz. Using the configured Keyphasor maximum speed, ADRE SKP software calculates and displays the maximum available synchronous sampling rate that can be selected without exceeding the 32000 Hz capability of the ADC.

The result of the synchronous sampling that we have described up to this point is a collection of synchronous waveform samples, each containing 2048 digitized values. These waveform samples represent the raw material that is used in two main groups of plots. These are time domain plots (orbit and timebase), and frequency domain plots (spectrum, cascade and waterfall).

Note: a “full” spectrum, cascade or waterfall plot is an enhanced plot that is produced by using the timebase waveforms from XY transducers. It displays the amplitudes of the forward (same direction as shaft rotation) and reverse (opposite direction to shaft rotation) frequency components of the vibration signal.

Time domain plots constitute a direct representation of the digitized signal, either as a single waveform or as an orbit, which is a combination of two signals coming from a pair of orthogonal XY transducers. In other words, no further processing, other than digitizing the analog signals, is performed when generating these type of plots. In contrast, the frequency domain representation requires additional manipulation of the original waveform files to extract the various frequency components from the signal.

Frequency Domain Analysis
For now, we will take a quick look at the basics of frequency domain analysis as it is used for machinery diagnostics. This will help with understanding the limitations of synchronous sampling, as well as some advantages that asynchronous sampling has for frequency domain analysis. Later, when we talk about asynchronous sampling, we will come back to the frequency domain considerations in more detail.

Once the analog signal has been sampled (transformed into digital format) the Fourier Transform is applied to the sample, through a computational algorithm called Fast Fourier Transform or “FFT”. The main purpose of this transformation is to go from a time domain (waveform) representation of the signal, to the frequency domain representation of that signal, in order to display a conventional spectrum plot (a representation of all the frequency components that exist in the original complex signal, along with their respective amplitudes).

One of the properties associated with the FFT algorithm, is that it takes equally spaced N samples from the time domain, and transforms them into N/2 equally spaced samples in the frequency domain, called “lines”.
• The number of spectral lines is one half the number of digitized samples in the waveform record.

The frequency resolution of the resulting spectrum plot, that is, how close together these lines are to each other, can be calculated as follows:

\[
\text{Frequency resolution} = \frac{\text{Frequency span}}{\text{Number of lines}}
\]

Consequently, if we apply an FFT to a synchronous waveform containing 2048 samples, we will get a 1024-line spectrum plot. Since all ADRE synchronous waveform records contain a fixed number of 2048 samples, there will always be 1024 lines available from such a sample.

In order to calculate the frequency resolution, we need to know which frequency span we are using. This brings up an important question: Is it possible to independently configure the frequency span in Synchronous sampling? The answer to this question is no. Frequency span is automatically set to half the selected synchronous sample rate.

Though this may sound arbitrary, it is not. It is based on the Nyquist sampling theorem, which states that in order to accurately extract the frequency information from the original signal, the sampling rate must be at least twice the highest frequency of interest in the original signal.

This requirement reduces the possibility of erroneously representing the original signal as a different signal with lower frequency. The presence of such spurious frequency components is referred to as “aliasing,” and will be discussed in more detail when we explore asynchronous sampling. By highlighting the phrase “at least,” we are emphasizing the fact that it is a minimum required sampling rate. When we discuss asynchronous sampling, we will see that a higher sampling rate is generally used.

**Frequency Span**

Let us go back to the first example scenario of Figure 5 to present some actual numbers. In this case, we saw that a synchronous sampling rate of 256/8 translates into 7692 samples per second (Hz), when the machine is running at 1800 RPM. As defined by the Nyquist criterion, sampling at 7692 Hz, or 256 samples per revolution, will allow us to accurately “see” frequency components in the digital signal up to 3846 Hz (half of the sampling rate described as frequency units), or 128 times running speed (half of the sampling rate described as “orders” of shaft rotation speed).

- 7692 Hz / 2 = 3846 Hz
- 256X / 2 = 128X

In summary, once the synchronous sampling rate has been selected, the frequency span will automatically be set to half that value. It is evident from this example that the resulting frequency span can tend to be much higher than is needed for a typical frequency analysis on rotating equipment – especially for fluid film turbo machines that do not generate very high orders of vibration relative to shaft speed (Figure 6).

If we also consider that this typically high frequency span is spread over a fixed number of only 1024 spectral lines, we will conclude that there are reasons to avoid the use of synchronous sampling in frequency domain analysis with the available options in the 408 DSPi. Synchronous sampling excels for time domain analysis, where the ability to collect many samples per shaft rotation results in a very accurate representation of the original analog signal. In other words, we will most often use synchronous sampling in those cases where the quality of the waveform itself is critical, either in timebase waveform plots or orbit plots.
Observe that due to the excessive frequency span of 64X, a relatively poor frequency resolution is achieved in this example: 64X/1024 lines = 0.0625X line.

Recall: 1X frequency is simply the shaft rotative speed in rpm divided by 60 seconds/minute. So 1X = 2998/60 = 49.97 Hz.

Converting frequency resolution in this example to other units:
- (0.0625)|(49.97 Hz) = 3.12 Hz
- (3.12 Hz)|60 s/min) = or 187.2 cycles per minute (cpm)

Synchronous Sampling Equations

Before we discuss asynchronous sampling in detail, here are some useful equations related to synchronous sampling. These can be very handy when configuring your 408 DSPi for collecting waveform samples.

\[
\text{Number of revolutions for completing one waveform} = \frac{\text{record size (2048)}}{\text{Configured sampling rate}}
\]

\[
\text{Time for capturing one waveform} = \frac{\text{Number of revolutions in record}}{\text{Running speed (rev/sec)}}
\]

\[
\text{Sampling rate (Hz)} = \frac{\text{Machine speed (rev/sec)} \times \text{Sampling rate (samples/rev)}}{	ext{2}}
\]

\[
\text{Frequency span (nX)} = \frac{\text{Sampling rate (samples/rev)}}{2}
\]

\[
\text{where } X \text{ represents running speed}
\]

\[
\text{Spectrum resolution (nX/line)} = \frac{\text{Frequency span (nX)}}{1024}
\]
Asynchronous Sampling

In the previous section, we discussed how synchronous sampling was performed in the 408 DSPi, emphasizing its dependency on the phase reference pulse. Asynchronous sampling, however, does not depend on this pulse at all. The main difference here is that all of the samples included in the waveform file are equally spaced in time, and the sample rate is independent from changes in rotating speed or the signal frequency.

Also, this type of sampling differs from the synchronous type in the number of samples within each waveform record. There are several options available, and the number of individual digitized values within a waveform record varies based on the number of spectral lines, and the Frequency Span, which are both configurable in this type of sampling. Let’s review each of these parameters separately.

Frequency Span (FS)

Recall – The Nyquist Theorem says that in order to properly identify all the frequency components within a defined frequency range, the data acquisition device must collect samples at a frequency of “at least” twice the highest frequency of interest. With the 408 DSPi (as well as most data acquisition devices), once a particular FS has been selected, the asynchronous sampling rate is automatically set to 2.56 times that value.

For example, if we wanted to examine the frequency content of a signal up to 1000 Hz, once we have selected a FS of 1000 Hz, the 408 DSPi sets a sampling rate of (1000) \(\times 2.56\) = 2560 Hz. Using a factor higher than 2 relates to the use of special low pass filters called “anti-aliasing” filters in this type of sampling (later we will talk a bit more about Nyquist and anti-aliasing filters). Though 2.5 would be an adequate multiplier to determine sample rate, 2.56 is the multiplier that is normally used, in order to comply with the digital sampling constraints of the computer world.

The 408 DSPi offers FS options ranging from 50 to 50,000 Hz, which result in sampling rates from 128 to 128,000 Hz.

Number of Spectral Lines

Numbers of spectral lines ranging from 100 to 6400 are available when using asynchronous sampling in the 408 DSPi. This capability, along with a configurable FS, allows us to achieve the desired frequency resolution, by using the same expression we introduced in synchronous sampling:

\[
\text{Frequency Resolution (Hz/line)} = \frac{\text{Frequency Span (Hz)}}{\text{number of lines}}
\]

At this point, it is important to state that this expression should be used very carefully. The next few pages explain the reason for this cautionary statement.

As we discussed in the synchronous sampling section, the number of spectral lines of the FFT (samples in the frequency domain) depends on the size of the corresponding waveform, that is, the amount of samples included in the waveform record. Though we also stated that the FFT delivers N/2 lines from N samples in the waveform, the 408 DSPi, as well as most data acquisition devices, uses N/2.56 when applying the FFT to asynchronous waveforms.

We could rephrase this as the following relationship:

\[
\text{Number of samples in the waveform} = \text{Number of lines} \times 2.56
\]

Again, using 2.56 instead of 2 in this expression relates to the use of the anti-aliasing filters, combined with a digital environment. Since the number of spectral lines is configurable for asynchronous sampling with the 408 DSPi, this equation becomes important, as the required amount of samples in the waveform will change based on the number of spectral lines that are specified. The list below shows all the available options for numbers of lines in the 408 DSPi, along with the number of digital samples in the waveform records.

- 100 lines: 256 samples in the waveform record
- 200 lines: 512 samples in the waveform record
- 400 lines: 1024 samples in the waveform record
800 lines: 2048 samples in the waveform record
1600 lines: 4096 samples in the waveform record
3200 lines: 8192 samples in the waveform record
6400 lines: 16384 samples in the waveform record

Relationship between Frequency Span (FS) and number of spectral lines

We have already discussed the two main configurable parameters involved with asynchronous sampling. Now let’s see how they combine with each other to determine the quality of the collected data.

The most important aspect of selecting a particular FS is not frequency span itself, but how fast the samples are collected to fill up a waveform file. As we stated earlier, the sampling rate is 2.56 times the selected FS. Also, we learned that the number of lines determines how many individual samples are required to be in the waveform record in order to generate the corresponding spectrum plot.

All of these relationships interact to determine how much time the data collection will take. If we want to apply an FFT algorithm to a waveform or time record, the time that is required to collect the samples may be very important. The reason for this is that the FFT algorithm assumes that the digitized signal is “periodic” throughout the time that was sampled.

In the real world, however, we should keep in mind that things may change with time. When dealing with rotating machines, the perfect example for changing conditions would be transient events such as startups, shutdowns, or speed changes (for variable-speed machines). To illustrate this point, let’s take a look at the following scenario. Hopefully, you will see that the shorter the collection time, the better is the possibility for acquiring a periodic signal.

Example

We want to collect vibration data during the shutdown of a machine. Since the machine is running at 4899 RPM (1X = 82 Hz), a frequency span of 500 Hz would be enough to examine the frequency content from a typical proximity probe signal.

We also set 400 lines for the corresponding spectrum plot, which combined with a FS of 500 Hz will give us 1.25 Hz/line of resolution. Everything looks fine, until we examine the cascade plot generated during the shutdown (Figure 7). There seems to be something wrong with the 1X and 2X frequency components in the range of 4000 – 5000 rpm. They are not exactly 1X and 2X, but are slightly lower frequencies.

So, what exactly is wrong with this scenario? Let’s start by looking at the asynchronous sampling settings. In this case, the selected frequency span of 500 Hz automatically sets the asynchronous sampling rate to 1280 Hz (500 * 2.56). We also know that 400 lines will require a waveform record containing 1024 individual digital samples (400 * 2.56) = 1024.

If the 408 DSPi collects 1280 samples in one second, the time required to fill up a waveform record with 1024 samples will be 0.8 seconds. If we want to generate a spectrum plot from a specific asynchronous waveform, we should verify that the amplitude and the frequency of the signal included in those 0.8 seconds do not change significantly.

In our example, the machine speed during shutdown followed the pattern shown in Figure 8. It changed rapidly at the beginning, and then more slowly. At the beginning of the coast-down, rotating speed dropped from 4842 to 4010 rpm in one second. Our 0.8 s sample time will probably include some changes in vibration amplitude, and definitely in frequency. As a result, the FFT will produce smearing in the corresponding spectrum plot.

The tradeoff of increasing the frequency span is a decrease in frequency resolution. In most cases, a compromise between the two will have to be considered. In this example, increasing the FS from 500 to 2500 Hz and keeping 400 spectral lines caused a decrease in frequency resolution from 1.25 to 6.25 Hz per line (Figure 9).
Figure 7: Full cascade plot for a machine shutdown, using a frequency span of 500 Hz.

Figure 8: RPM trend plot for a typical machine shutdown. Speed drops rapidly at first, and then more slowly as the machine coasts to a stop.
Nyquist Theorem and Aliasing

When we talked about synchronous sampling we briefly introduced the Nyquist Theorem and an extremely important concept related to signal sampling – Aliasing. We will now take a deeper look at these concepts, as they are accounted for in asynchronous sampling. In order to do this, we will consider the following scenario:

We need to digitally sample the analog signal shown in Figure 10. For the purpose of this example, we will assume that the signal to be sampled is a single frequency sinusoidal signal. In previous sections of this article, we learned about a particular rule we need to follow when sampling signals, the Nyquist Theorem. We will stick to it by taking samples at twice the frequency of the analog signal.

Figure 11 shows black dots overlaid on the original signal. These represent the discrete samples collected by the acquisition device. By collecting two samples per signal cycle (which is the same as collecting samples at twice the signal frequency), we make sure that we can identify alternating amplitudes at the frequency of the original signal. In other words, we can detect the frequency of the original signal.

Now let us see what happens if we collect samples at a lower frequency. Figure 12 again shows black dots representing the digital samples, which are now being captured at different locations in the original signal. A direct consequence of this is that when we try to reconstruct the digitized signal, we will see a different signal (green). By comparing it to the original signal (red), it is evident that the frequency of the reconstructed signal is lower than the original. In other words, the original high frequency signal has been “aliased” to a lower frequency signal. It is important to state that once the aliasing of a signal has occurred, there is no way to know whether the reconstructed signal accurately represents the original signal or not.

Figure 9: Full cascade plot for a shutdown, this time using a frequency span of 2500 Hz.
Figure 10: Simple example of an original analog signal.

Figure 11: Digitized samples collected at twice the original signal frequency.

Figure 12: Digitized waveform (green) shows aliasing caused by sampling at too low a sample rate.
Anti-aliasing Filters

Even when the Nyquist theorem ensures that we detect the frequencies of interest, undesired higher frequency components may still generate aliased components that will show up in our selected frequency span. Let us suppose that we want to examine a frequency span of 0 to 1000 Hz. The Nyquist theorem will force us to sample at twice the maximum desired frequency, in this case, 2000 Hz. This will ensure that any frequency component up to 1000 Hz will appear accurately in the reconstructed signal.

However, what would happen if the analog signal included a random 1600 Hz component? Signal sampling theory tells us that any frequency component f0 above half the sampling frequency will generate an aliased component at sampling frequency − f0. In our example, a 1600 Hz component will generate an alias at 2000Hz − 1600Hz = 400Hz. When this happens, we will observe a 400Hz frequency component that was not in the original signal. This can cause a tremendous waste of time trying to relate it to a real problem.

In order to avoid this problem, anti-aliasing filters are usually applied to the original signal before the digital sampling process is performed. These are basically low-pass filters that remove frequency components above our selected frequency span. By attenuating signal components that are beyond our identified frequency span, these filters reduce the chance of aliasing of any higher frequencies that might be in the original signal.

Ideally, an anti-aliasing filter would have the perfect characteristics shown in Figure 13. All signal components lower than the selected cutoff frequency are in the “pass-band” and are allowed to pass through the filter completely unaffected. All signal components higher than the cutoff frequency are in the “stop-band” and are perfectly rejected, or prevented from passing through the filter. The cutoff frequency for such a filter would be set at the selected value of frequency span for the sample to be collected, and the sample rate could be as low as exactly two times the cutoff frequency.

![Figure 13](image1.png)

Figure 13: This example shows the behavior of an ideal low-pass filter used for anti-aliasing.

![Figure 14](image2.png)

Figure 14: Real anti-aliasing filters have a sloped attenuation response rather than a sharp cutoff corner. This creates a transition band with only partial signal rejection.

![Figure 15](image3.png)

Figure 15: The actual frequency span that must be accommodated by the digital sampling process is defined by the high-frequency end of the transition band, rather than by the FS that has been selected in the configuration properties.
Unfortunately, real filters behave as shown in Figure 14. Here, a transition band is present, starting at the cutoff frequency, and with a sloping “roll-off” characteristic rather than a sharp corner. Just above the cutoff frequency, the signal is barely rejected at all. The attenuation increases as frequency increases within the transition band, eventually being complete where the stop-band is reached. If we set the cutoff frequency to the selected frequency span for the sample to be collected, and set the sample rate at exactly two times the cutoff frequency, any incompletely-attenuated signal components in the transition band would be aliased down into the digitized waveform file.

To avoid the possibility of components within this transition band aliasing into our selected frequency span, the sampling rate must be set to twice the highest frequency of the transition band. This point defines the actual FS that must be accommodated by the sampling process, rather the configured value of FS (Figure 15). The typical asynchronous sampling rate used in most data acquisition devices working in a digital environment, including the 408 DSPi, is 2.56 times the configured value of FS.

The asynchronous sampling rate used by the 408 DSPi is 2.56 times the selected value for FS. This sample rate is high enough to accommodate the width of the transition band, and prevent aliasing of signals beyond the configured setting for FS. It also means that more digital samples are collected in the waveform record than are displayed in the resulting spectrum plot.

**Example**

If we select 400 spectral lines, and FS = 1000 Hz, the number of collected samples in the waveform record is actually $\left(\frac{400}{2.56}\right) = 1024$ samples, as shown in this relationship for asynchronous sampling:

$$\text{Number of samples in the waveform} = \text{Number of lines} \times 2.56$$

There are three "Laws of the Universe" that deal with digitally sampling waveforms and computing spectra with a Fast Fourier Transform (FFT):

- The sample rate governs the frequency span.
- The total sample time governs the frequency resolution.
- The number of samples governs the number of lines.
1024 samples are actually enough to create 512 spectral lines. Also, since the actual sample rate is \((1000)(2.56) = 2560\) Hz, the actual FS is 2560/2 or 1280 Hz, rather than our configured value of 1000 Hz.

Because we are only interested in the original frequency span (1000 Hz), and the selected number of spectral lines (400), our process truncates the spectrum to discard the highest frequency data (from 1000 to 1280 Hz) and the 112 spectral lines beyond 400. This process provides additional assurance against aliasing.

The 2.56 factor also accommodates the binary storage capability of digital buffers (512, 1024, 2048, etc.), and user preferences for traditional “round numbers” of spectral line options (400, 800, 1600, 3200, etc.).

**Advantages of Asynchronous Sampling**

In many cases, two characteristics of the 408 DSPi combine to make asynchronous sampling more appropriate than synchronous sampling for frequency domain analysis:

- The 408 DSPi allows both the frequency span and the number of spectral lines to be configured individually for asynchronous sampling. This allows more flexibility to select settings that correspond to the conditions of sample collection.
- The 408 DSPi applies anti-aliasing filters for asynchronous sampling. This reduces the chance of spurious frequency components showing up in the resulting spectrum plots.

Just as we did for synchronous sampling, we include some useful relationships for asynchronous sampling here.

### Sampling rate [samples/sec] = Frequency Span [Hz] * 2.56

### Time required for capturing a waveform = \(\text{Waveform size [samples in the waveform]} \div \text{Sampling rate [sample/second]}\)

### Number of samples in the waveform = \(\text{Number of lines} \times 2.56\)

---

**Conclusion**

There are three “Laws of the Universe” that deal with digitally sampling waveforms and computing spectra with a Fast Fourier Transform (FFT):

- The sample rate governs the frequency span.
- The total sample time governs the frequency resolution.
- The number of samples governs the number of lines (Reference 1).

We can generally say that synchronous sampling is often preferable for time domain analysis, and asynchronous sampling for frequency domain analysis. However, it is important to understand the relationships between these three laws, so that we can select settings that are an appropriate compromise for the machine conditions that exist at the time of sampling. (Reference 2)

*Denotes a trademark of Bently Nevada, Inc., a wholly owned subsidiary of General Electric Company.

**References**

1. Sampling waveforms and computing spectra, by Don Southwick. ORBIT Vol. 14, No.3, pg. 12, September 1993. This article is one of our classics on the topic of basic digital signal processing.

2. 3200 line spectrum – when shouldn’t you use it? ORBIT Vol. 14 No. 3, June 1998. This article includes several examples of the tradeoffs that occur when collecting waveform samples with higher sample sizes – especially when the data collection may occur during machine speed transients, which can cause spectral smearing.

Both of these references are available for viewing or downloading at our Orbit Archive Directory: www.orbit-magazine.com