

## A practical take on FFTs

Michael Dunn - November 26, 2013

Virtually all scopes today provide the FFT (Fast Fourier Transform) as a standard feature, but in my experience, few users know how, when, or why to use it. And if they do, they rarely use it well.

Why use an FFT? The list of answers is almost as long as the one for "Why use a scope?" Measuring distortion is a classic example, whether it's at 50/60Hz, audio, or RF frequency. Perhaps you want to see the spectral characteristics of some noise, or maybe the occupied spectrum of a serial data signal.

As an aside, the first real FFT equipment I got my own paws on was a lovely HP dedicated audio signal analyzer in the mid-1980s.



The HP 3561A Dynamic Signal Analyzer.

Unfortunately, this spoiled me in terms of UI and functionality, and I've yet to meet a scope that comes close. *Factoid:* This instrument used [bubble memory](#) to store setups and data (think *hard disk on a chip*).

Reviewing the utter basics: An FFT takes a signal in the time domain and converts it (*transforms* it, you might say) to the frequency domain. A 60Hz sinewave (time domain) transforms to a spike at 60Hz (frequency domain). If there are spikes at 120Hz, 180Hz, and so on, then you know there's

distortion.

And reviewing our basic waveshapes: A square wave has only odd harmonics, falling off at  $1/N$ . A triangle has only odd harmonics, falling off at  $1/N^2$ . A ramp has all harmonics, at  $1/N$ . A pulse-train can have missing harmonics. For example, a 10 percent duty-cycle pulse will lack the 10th, 20th, 30th harmonics, and so on.

OK, we want to look at the FFT of a signal on our scope. What parameters will sculpt the view? There's:

- Record size
- Ratio of signal period to record time
- Windowing function

**Record size** is limited by the scope's buffer size – and your patience. Well, patience is becoming less of an issue, because a modern scope can churn out million-point FFTs in the blink of an eye. The main effect of record size is on frequency resolution. For example, an 8k record FFT will generate 4k frequency *bins* (it's always half). FFTs always start at 0Hz, so if the frequency span is 20kHz, our resolution is  $20k/4k = 5\text{Hz}$ . That's pretty good on a linear scale, but if you prefer to view your frequency axis logarithmically, and you don't want the low end to be too coarse, more points may be called for.

The **ratio of signal period to record time** determines where on the X-axis frequency scale the signal components appear. If you time things such that exactly one cycle fills the FFT's input record, its fundamental spectral line will be in the second bin (the first being DC). Any harmonics in the signal will fill up successive bins. It might seem neat and tidy, but it is not usually possible.

First off, in the real world, it's virtually impossible to get exactly one cycle in the buffer. Let's say you get 1.01 cycles. This will cause every spectral line to be *smeared* into its neighbors. It's usable but messy. You're much better off capturing perhaps 10 cycles or 100 (depending on the FFT size and the signal's harmonic content). All the spectral lines will be moved up and spread out. That's much better.

I mentioned frequency span, but not in the parameter list. That's because it's a secondary value. Span simply equals half the sampling rate. Therefore, we can redefine frequency resolution as the sampling rate divided by FFT size.

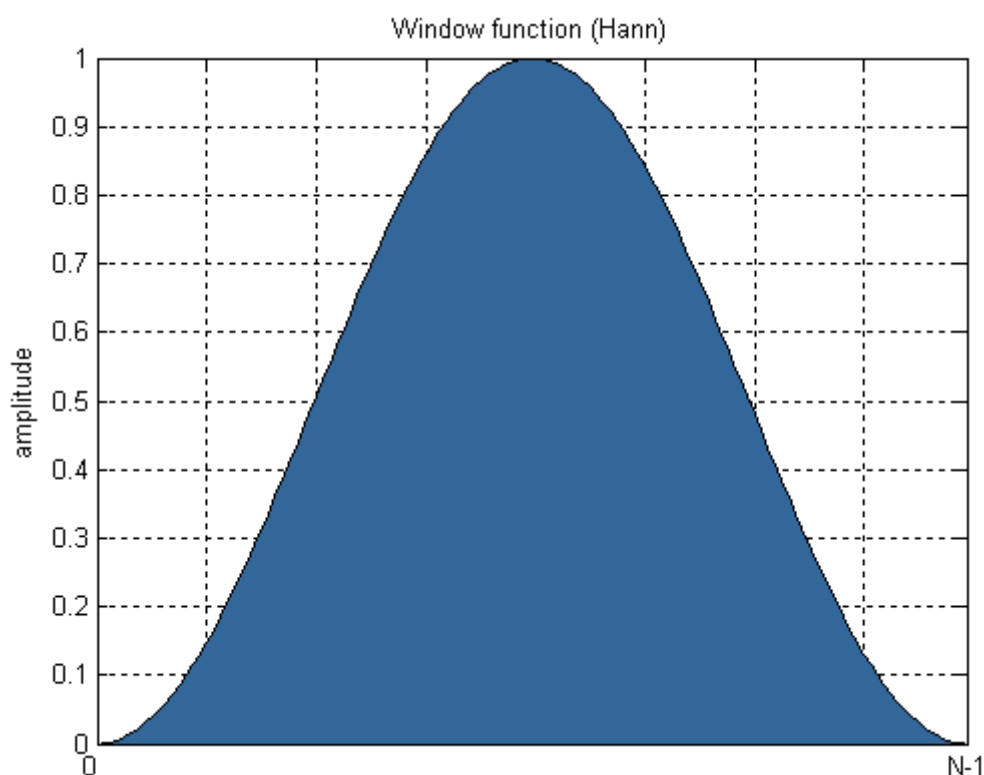
In normal scope operation, there is usually some interaction between sampling rate, sweep speed, and record size. If the FFT UI isn't well thought out, these interactions can drive one to drink (or worse), as you adjust controls and find your careful FFT settings ruined by a well-meaning turn of a knob. If the FFT is something you'll be using a lot on a new scope, make sure to try before you buy.

The last item under our control is the **window function**. "What nonsense is this?" you ask. It's an unfortunate necessity in dealing with real-world signals, and it's actually pretty ingenious.

The Fourier transform, regardless of flavour, assumes a continuous signal over infinite time. Discrete versions of the transform still adhere to this rule – it's assumed the finite waveform record repeats infinitely. So, imagine multiple records spliced together: The end of the record must splice smoothly to the beginning of the next – identical – record (recall the previous talk about fitting exactly one cycle in your FFT input). If we could do this – feed a perfectly integral number of cycles to the FFT – all would be simple (and this also assumes we're only interested in analyzing

harmonically related signals).

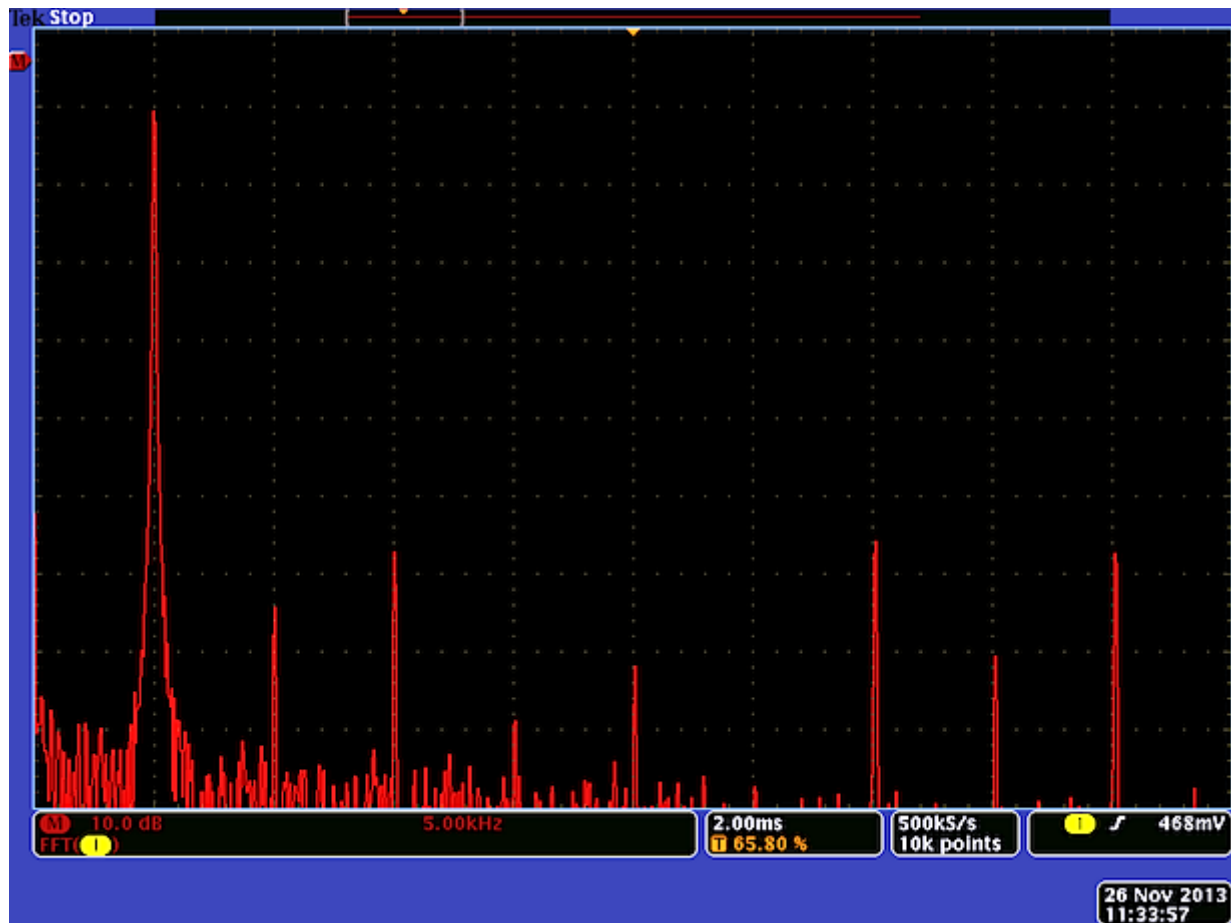
But we both know it ain't gonna work out that way. The inevitable glitch at the join point between the end and start of the record will cause a whole slew of nasty *faux* harmonics to appear. Try it. That's where the [window function](#) comes in.



A von Hann, or Hanning, window.

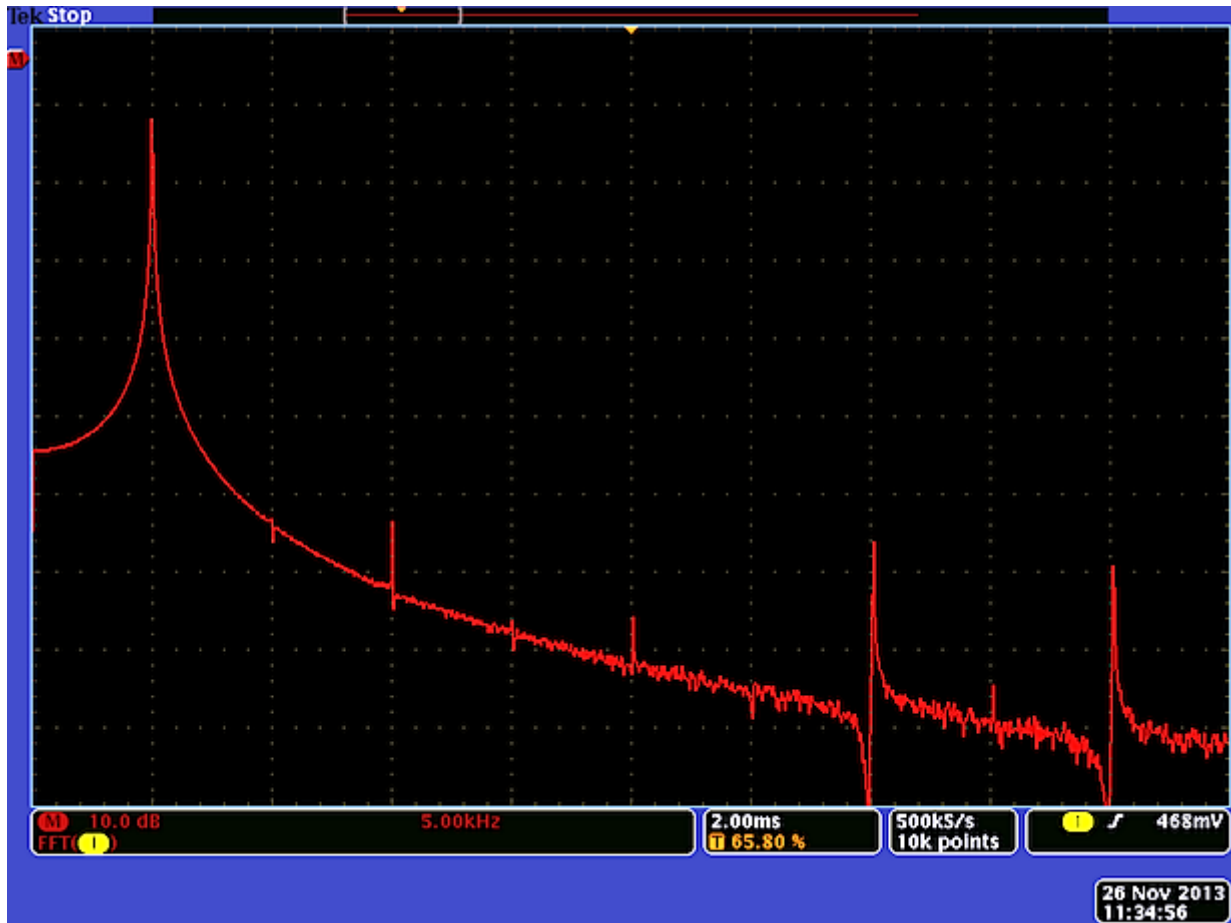
The window above is a commonly used one, and its basic shape is typical. This curve is aligned with the captured record, and is used to scale each point by the curve value at that X-position. The resulting record looks like a tone burst, and as the record is assumed to repeat infinitely, the FFT performs its magic on, in effect, this repeating tone burst. "And that's better!?" you exclaim.

Yes it is. True, the window imposes its own harmonic smearing and other artefacts on the spectrum, but they're predictable. If your main interest is reading accurate harmonic levels, a window with a flat top can be chosen. If better frequency resolution and dynamic range is desired, the "Hanning" window pictured above, or one similar, works well. Often, one uses two or three different window functions during an analysis session. Don't worry. It quickly becomes intuitive. The references listed below will help you get started.

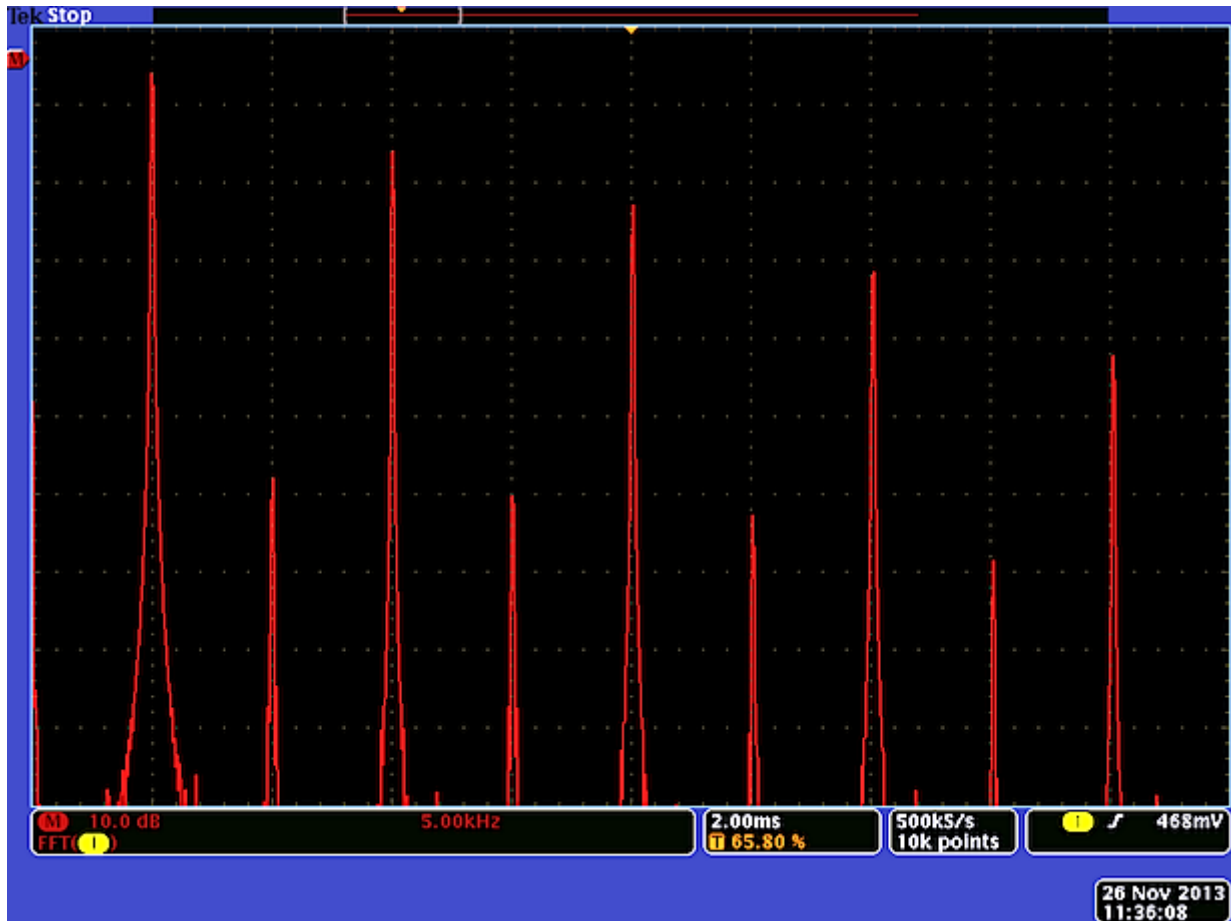


FFT of a 5kHz sine. The on-screen information shows that the vertical scale is 10dB/div, and the horizontal, 5kHz/div. The acquisition settings are 2ms/div, 500kSa/s (high-res), & 10k record. The Hanning window is selected. Also, the display is zoomed in, as shown by the zoom indicator at the top of the screen. Total displayed span is 50kHz, but up to 250kHz is available.

The signal's strongest distortion harmonics are the 3rd, 7th, and 9th, all around -56dBc.



The exact same signal as above, but with a rectangular window (i.e., no window at all)! Note the spectral "smear" and raised noise floor.



Back to a Hanning window, but with the signal amplitude raised to cause scope input clipping. The spectrum starts to resemble that of a square wave.

What else can be done to improve your FFT experience? There should be an averaging mode available to help tame all the noise and pull out low-level frequency components. The waveform itself could be acquired with averaging on too. I strongly recommend this and/or the use of high-resolution mode, which will do wonders to improve the dynamic range of the FFT display.

Have you had frustrating FFT experiences? I hope this article will improve your future ones.

#### Also see:

- [Understanding FFT plots](#)
- [Windowing Functions Improve FFT Results, Part I](#)
- [Know the Limitations of Your Scope's FFT Capabilities](#)
- [The practicing instrumentation engineer's guide to the DFT - Part 1: Understand DFT and FFT implementations](#)
- [Scope Junction blogs](#)

#### References:

- [FFT Applications \(Tek\)](#)
- [Spectral Analysis Using FFT \(Agilent\)](#)
- [More About The FFT \(LeCroy\)](#)

Follow Michael Dunn at 