

Audio Math in QA40xPlot

Dec 19, 2025

Fundamental Signal RMS Voltage

The RMS voltage at the fundamental frequency is simply the height of the fundamental frequency's bin in the FFT.

V @ Frequency

Signal RMS Voltage

The RMS voltage is calculated one of two ways. For time domain data it's just the RMS definition

$$V_{RMS} = \frac{\sqrt{\left(\sum_{t=0}^T V_t^2\right)}}{n}$$

For bandlimited frequency data it's a bit different since frequencies values are like densities.

$$V_{RMS} = \frac{\sqrt{\left(\sum_{fmin}^{fmax} V_f^2\right)}}{ENBW(windowing)}$$

ENBW is the equivalent noise bandwidth of the fft windowing method. In some ways it's a measure of how much the signal is smeared into adjacent channels. It can be calculated by

$$ENBW = \frac{\sqrt{\left(\sum_{t=0}^T W_t^2\right)}}{\sum_{t=0}^T W_t}$$

Where W_t is the fft weight at time t . Note this is scale-independent of the weights.

Harmonic Distortion

Using the frequency domain results the distortion voltage, in VRMS, of each harmonic is simply the height of the harmonic bin, V @ Harmonic Frequency.

The total harmonic distortion (THD) is the RMS total distortion divided by fundamental.

$$THD = \frac{\sqrt{\left(\sum_{t=2}^N V_{Ft}^2\right)}}{V_F}$$

Currently QA40xPlot uses N=7. Note that distortion spectra are clearly in-phase but the above approximation is good enough.

Intermodulation Distortion

CCIF style math for IMD

When close together fundamentals ($f_H/f_L < 2$) use the 2nd order CCIF2 or 3rd order CCIF3. QA40xPlot uses CCIF3.

CCIF2 uses a single value

$$CCIF2 \text{ IMD} = \frac{V_{f_H-f_L}}{V_{f_H}+V_{f_L}}$$

CCIF3 uses a different single value

$$CCIF3 \text{ IMD} = \frac{\sqrt{V_{f_H-f_L}^2 + (V_{2f_L-f_H} + V_{2f_H-f_L})^2}}{V_{f_H}+V_{f_L}}$$

SMPTE/DIN IMD (or MOD IMD)

When the fundamentals are far apart ($f_H/f_L > 7$) use SMPTE/DIN math

$$SMPTE/DIN \text{ IMD} = \frac{\sqrt{(V_{f_H-f_L} + V_{f_H+f_L})^2 + (V_{f_H-2f_L} + V_{f_H+2f_L})^2}}{V_{f_H}}$$

RMS Power IMD

Finally, when $2 < f_H/f_L < 7$ use IMD RMS power methods using RMS addition

$$POWER \text{ IMD} = \frac{\sqrt{V_{f_H-f_L}^2 + V_{f_H+f_L}^2 + V_{f_L-2f_H}^2 + V_{f_L+2f_H}^2 + V_{f_H-2f_L}^2 + V_{f_H+2f_L}^2}}{\sqrt{V_{f_H}^2 + V_{f_L}^2}}$$

Noise Weighting Curves

From [A-weighting - Wikipedia](#)

The function(f) for C weighting is:

$$W_c(f) = \frac{12194^2 * f^2}{(f^2 + 20.6^2)(f^2 + 1219^2)}$$

Adding two real-axis poles to the C-weighting transfer function gives us A-weighting:

$$W_A(f) = \frac{12194^2 * f^4}{(f^2 + 20.6^2)(f^2 + 1219^2) \sqrt{(f^2 + 107.7^2)(f^2 + 737.9^2)}}$$

And unweighted effectively is

$$W_z(f) = 1.0$$

The weights are normalized to 1.0 at 1KHz.

Biquad Filters

A Biquad filter is the way to generate an RIAA equalization curve. It's a ratio of quadratic polynomials.

$$H(s) = \frac{(1 + Z_1 S) * (1 + Z_2 S)}{(1 + P_1 S) * (1 + P_2 S)}$$

Where Z1 and Z2 are the zeros and P1 and P2 are the poles expressed as time points. So, RIAA uses 3 [or 4] time points where: $T = 1 / (2 \pi F)$. The fourth point (the 50KHz pole) is to avoid overflows at high frequencies where there's no audio anyway.

RIAA Preemphasis

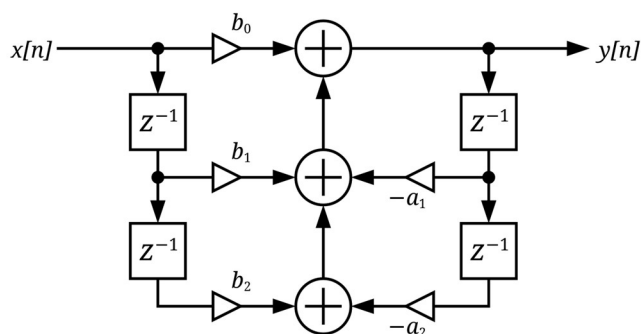
Variable	Value	Frequency
Z1	3180 μ S	50.05 Hz
Z2	75 μ S	2123 Hz
P1	318 μ S	500.5 Hz
P2	3.18 μ S	50050 Hz

Filtering in the frequency (FFT) domain use $S=j\omega$ in the complex domain. Where $\omega=2 \pi F$, so

$$H(f) = \frac{(1 + jZ_1 * 2\pi F) * (1 + jZ_2 * 2\pi F)}{(1 + jP_1 * 2\pi F) * (1 + jP_2 * 2\pi F)}$$

Time Domain Filtering

With the above biquad equation, the digital Z domain filter (where Z-1 is last step, Z-2 is 2 steps ago...) becomes



$$\text{Or } Y[t] = b_0 X_t + b_1 X_{t-1} + b_2 X_{t-2} - a_1 Y_{t-1} - a_2 Y_{t-2}$$

The filter coefficients are calculated from the transform $T' = e^{-\frac{1}{f_s T}}$

From [https://linearaudio.net/sites/linearaudio.net/files/v10 sw app1 table a-1.xlsx](https://linearaudio.net/sites/linearaudio.net/files/v10%20sw%20app1%20table%20a-1.xlsx)

Biquad IIR Coefficients for Popular Sampling Frequencies	poles zeros	a coefficients b coefficients	Gain at 1kHz (dB)
48 kHz	[52.5257285337, 2287.22026941] [21833.6972506, 536.805364274]	[1. -1.73273489, 0.73451925] [1. -0.75549731, -0.16463057]	13.16179286
With the added 50 kHz zero	[52.3147934774, 2274.69993092] [20053.9676045, 533.822795553]	[1. -1.7340031, 0.73577185] [1. -0.79733573, -0.1260212]	12.87643103
96 kHz	[50.1482157346, 2125.92919847] [38433.3251281, 501.903676141]	[1. -1.866632, 0.86705829] [1. -0.85352776, -0.11046426]	18.62517791
With the added 50 kHz zero	[50.1411966368, 2125.49371853] [30478.7992823, 501.800206324]	[1. -1.86665737, 0.86708352] [1. -0.96898135, 0.00125169]	17.67778023
192 kHz	[50.0537159311, 2121.63154226] [75313.5235611, 500.558587601]	[1. -1.93126252, 0.93137234] [1. -0.87968338, -0.10237808]	24.35100343
With the added 50 kHz zero	[50.0537545743, 2121.60617514] [41656.8730375, 500.552807373]	[1. -1.93126329, 0.93137311] [1. -1.17308964, 0.18626087]	21.66968409