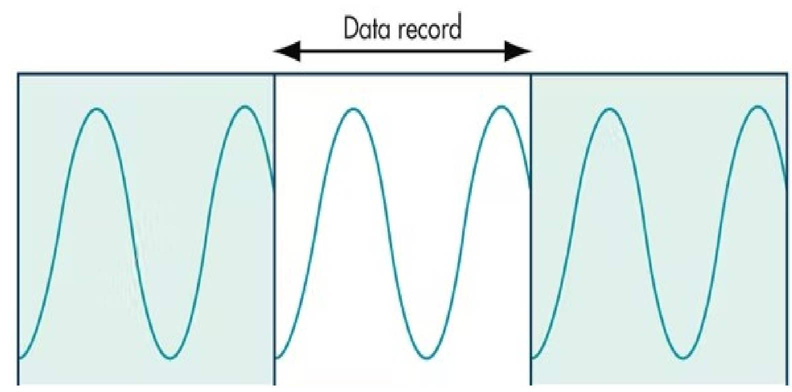
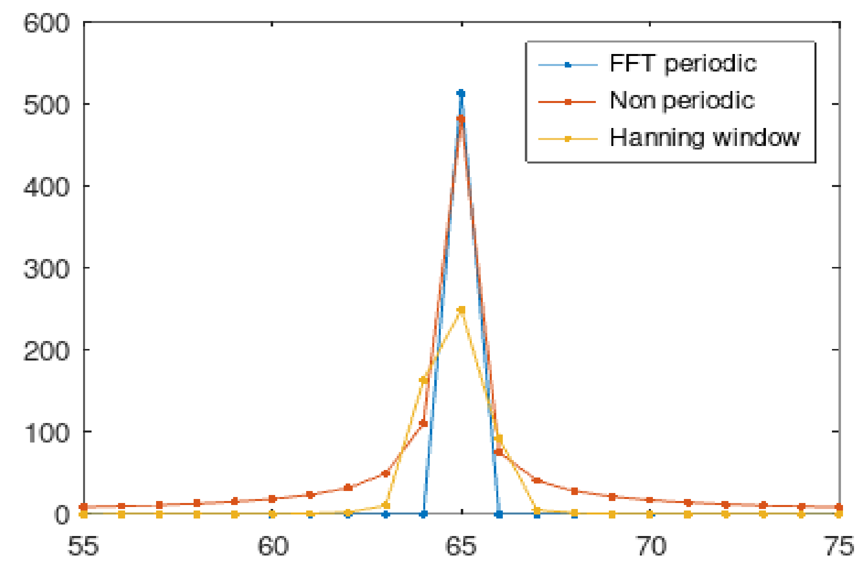
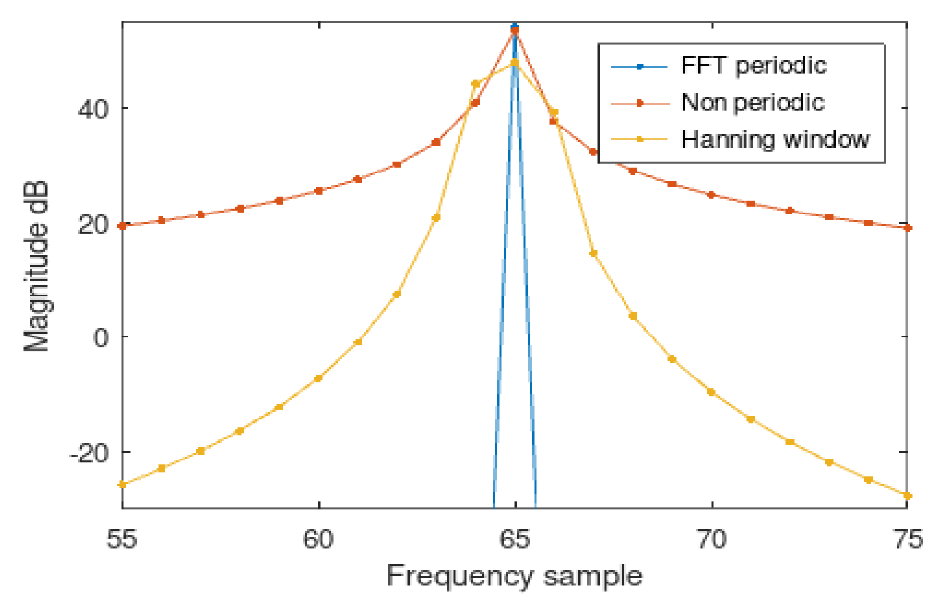
Fast Fourier Transform (FFT)

The Fourier transform was created by Joseph Fourier in a manuscript submitted in 1807. The Fourier transform approximates functions with the sum of simple trigonometric functions. The original Fourier transform was for continuous functions of real argument generating continuous function of frequency. The inverse transform was also defined.

The normal computer usage is for discrete-time sampled data. The Discrete Fourier transform (DFT) is for processing this type of data, producing a spectrum of the frequencies found in the input. For a N sample DFT the results are for a infinite series of repetition of the N samples. This will cause errors when processing non-periodic signals or when period doesn't exactly match the number of samples. Left plot below is linear magnitude which is what the PDP-8 is displaying. Difference is more obvious in dB. PDP-8 ADC is 9 bits signed (+-255) so so has approximately 54 dB *20*log10(V)* dynamic range.



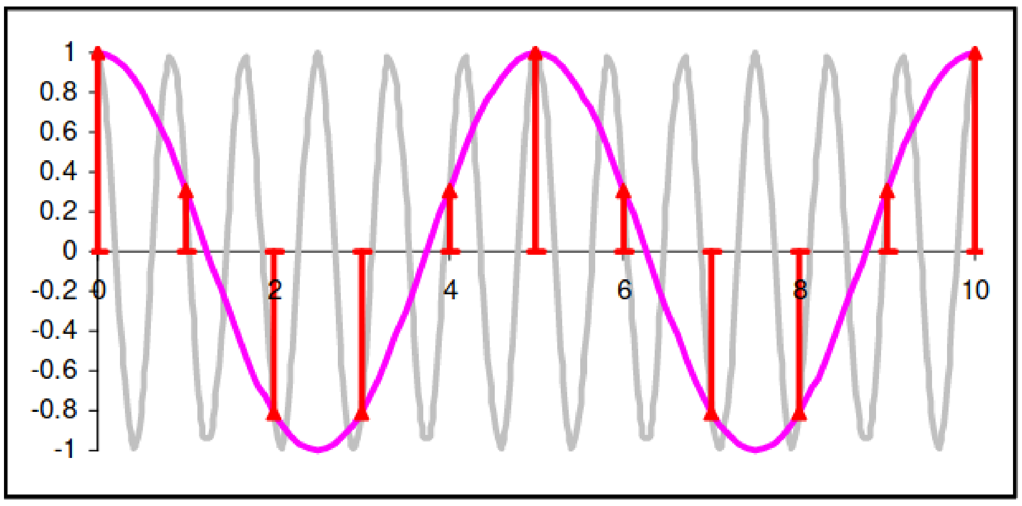




There have been a number of window functions defined that trade off amplitude change, spectral widening, and attenuation of the spectral leakage. The program uses Hanning window which widens the spectrum by 1.5 bins, and attenuates noise by 1.76 dB and coherent signals by 6 dB (half amplitude)

The direct implementation of the DFT equations requires N^2 operations. The Fast Fourier transform (FFT) is an implementation that calculates the transform in N*log(N) operations. The work by J.W. Cooley and J.W. Tukey published in 1965 triggered the efficient implementation on computers. DECUS-8-250 being demoed is from 1970. Looking back in history there were several similar algorithms dating back to Carl Friedrich Gauss in about 1805 though not published during his lifetime. None of the earlier works had widespread usage.

For FFT of real signals the highest signal frequencies must be limited to half the sample rate. For 20 kHz sample rate the signal must not have frequencies above 10 kHz. When you sample a signal that is above the sample rate the signal will alias to a lower frequency. It is easy to get incorrect results with the wrong sample rate. The gray signal will be aliased down to the magenta signal when under sampled.



Ask you you would like me to attempt a clearer descriptions of this gobbledygook.

https://en.wikipedia.org/wiki/Fourier_analysis
https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7389485
https://www.cis.rit.edu/class/simg716/Gauss_History_FFT.pdf