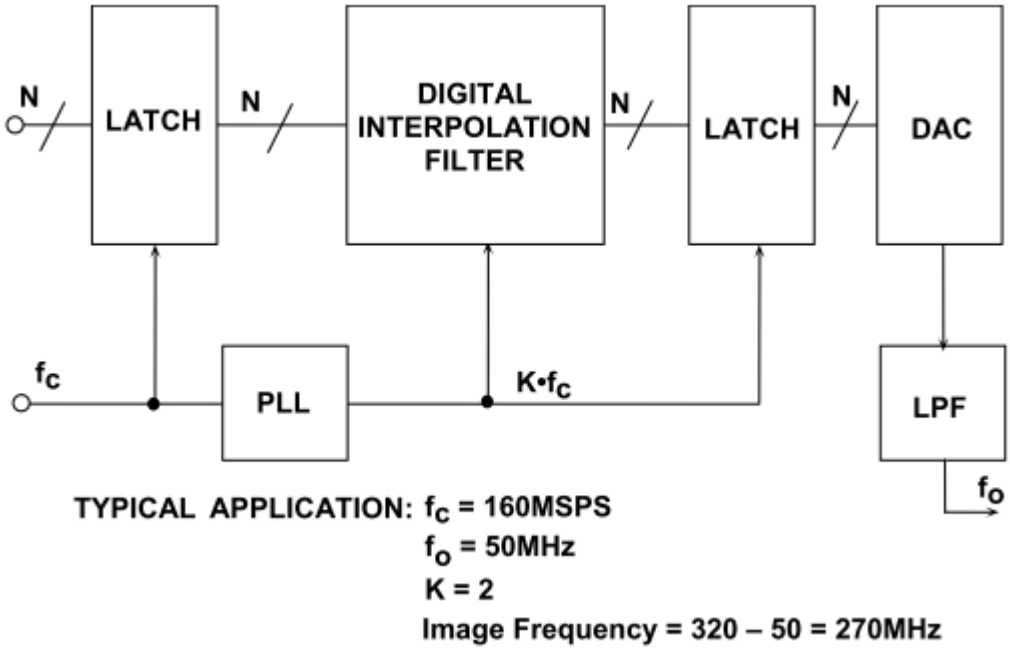
Finding Support for Spectral Decimation Section PhL 3.17.13

www.analog.com/static/imported-files/tutorials/MT-017.pdf

This paper draws this picture,



which suggested to me two things:

(1) there is no decimation here.
(2) my brick wall filter in this context is called an "interpolation filter".

http://www.scribd.com/doc/20203269/Interpolation-filters-for-oversampled-audio-DACs

This paper is cannot be downloaded without paying, but you can see it. Early comment:

Oversampling in principle consists of two processes. The first is to increase the input sampling rate f_{si} by inserting zeros between the existing ones, called zero-stuffing. Then, the signal is band limited using a digital low-pass filter. These are combined using an interpolation filter. The process is shown in the time and frequency-domain for 2x oversampling in fig.1.

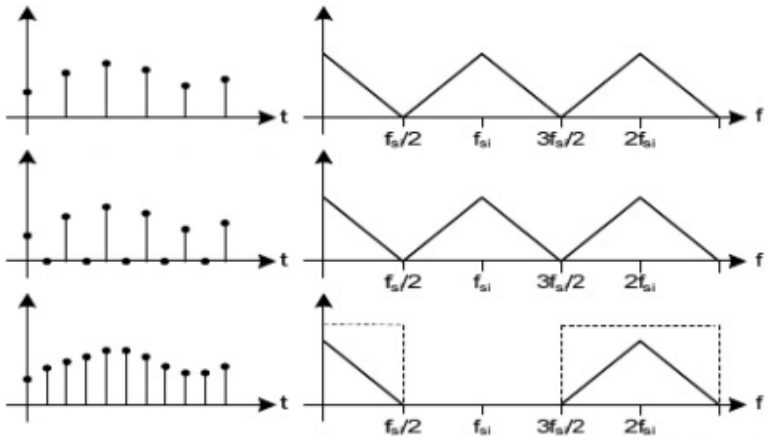


Figure 1: Oversampling in the time and frequency domain

Ideally, the output signal would have exactly the same spectral content as the input signal in its Nyquist-band and consequently the ideal interpolation-filter is characterised by:

$$H_L(f) = \begin{cases} 1 & , 0 < f < \dfrac{f_{si}}{2} \\ 0 & , \dfrac{f_{si}}{2} < f < \dfrac{Lf_{si}}{2} \end{cases} \qquad (1)$$

He refers to decimation as "zero-stuffing" but he does not say why one might do this. Also, it seems wrong that the two signals shown have the same spectrum, I think this might be wrong.

What I find is that decimation broadens the sinc function which has good and bad news. The good news is that you have less sinx/x distortion on the central peak, but the bad news is that you have added a lot of closer in image spectra! That is what decimation does, and I want to find someone who agrees with me.

Another figure is this one:

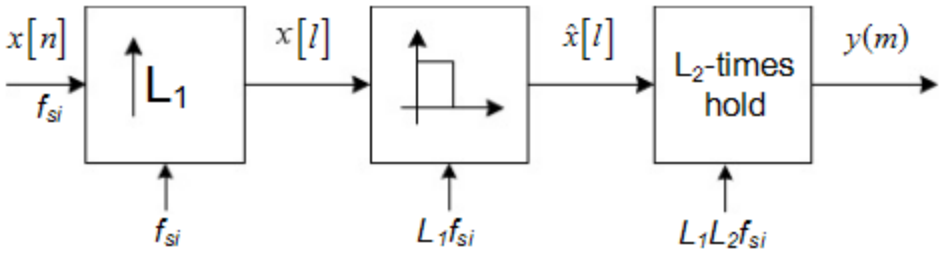


Figure 4: Lower OSR FIR-filter followed by hold-circuit

What do these notations mean? Here it is again

The simplest form of implementation is to directly realize the non-recursive difference equation that is the convolution sum. This is called the direct-form structure and is shown in fig.6. The zero-stuffing is added at the beginning indicating the filter being part of an interpolator with oversampling factor L:

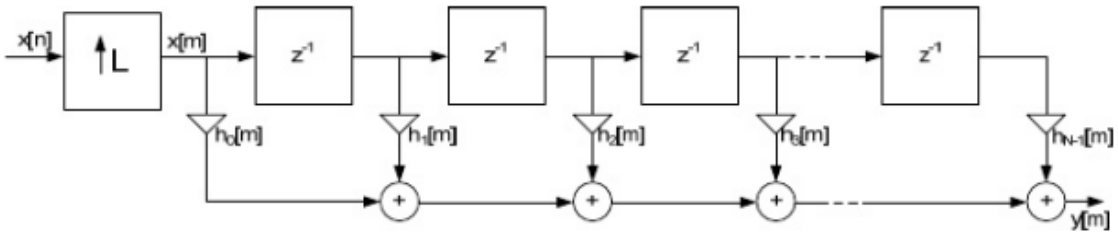


Figure 6: Direct form implementation of FIR filter.

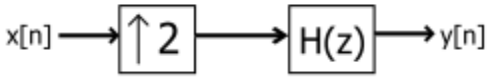
I think the leftmost box does both decimation and Low pass filter. As for filter implementation,

However, the direct-form implementation is disadvantaged in terms of efficiency when used in interpolators. It requires N multiplications per output sample $y[m]$, even though we know the input signal is zero-stuffed. For L times oversampling, only every L^{th} input value is nonzero. Then it is evident that implementing every factor $h[m]$ in the difference equation leads to many redundant multiplication operations.

He shows how you can do this with a polyphase idea to reduce hardware, seems reasonable. It is a good paper, but I need another source.

www.hit.bme.hu/~papay/edu/Conv/pdf/Jonas.pdf

This guy also uses an up arrow



He doesn't say if the box does zero-stuffing or not.

http://books.google.com/books?id=Kvo7cjmaEpkC&pg=PA391&lpg=PA391&dq=decimation+oversampling+DAC&source=bl&ots=77M7ekBiLK&sig=9YGbS-d2F29NKcbIrhzZcLDD3HE&hl=en&sa=X&ei=zytGUceiEuj7ygG_s4DABA&ved=0CGAQ6AEwCA#v=onepage&q=decimation%20oversampling%20DAC&f=false

This one has an interesting idea of doing a filter in stages,

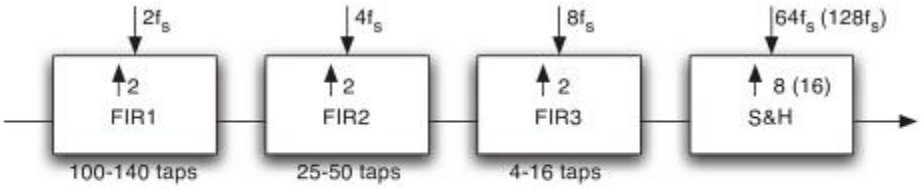


Figure 8.36. Cascade of interpolation sections to obtain interpolation.

Each stage runs 2x faster than the one before. Fast clock uses more power, but filters have fewer taps so less power. I like it. What is the rule of "sample and hold" here?

http://www.dspguru.com/dsp/faqs/multirate/decimation

Loosely speaking, "decimation" is the process of reducing the sampling rate. In practice, this usually implies lowpass-filtering a signal, then throwing away some of its samples.

The most immediate reason to decimate is simply to reduce the sampling rate at the output of one system so a system operating at a lower sampling rate can input the signal. But a much more common motivation for decimation is to reduce the *cost* of

So I am using decimation incorrectly! I am really doing zero-stuffing.