Signal Processing Systems (521279S)
Part 4(b) : Multirate implementation of filters

Version 1.1

November 18, 2025

Goals of the study. In the design task, we consider implementation of narrow-band FIR filters using multirate techniques, which offer considerable savings in the rate of computations.

Contents

History
V1.0 11.10.2019 .
V1.1 18.11.2025 Add reference to multirate IIR filtering.

1 Problem

Let us consider the problem of FIR filter design. When the filter has a narrow passband[1], the number of filter coefficients can be huge as demonstrated by the following example.

Example 1. Consider implementation of a lowpass FIR filter whose specification is

Sampling frequency F_0	48 kHz
Passband cutoff frequency f_p	1000 Hz
Stopband cutoff frequency f_s	1200 Hz
Maximum passband ripple A_p	0.1 dB
Minimum stopband attenuation A_s	60 dB

As cutoff frequencies are much lower than F_0, this is a narrowband filter. Matlab's filter design tool, **fdatool**, gives an equiripple FIR filter whose length is 653 taps (order is 652). The functions **firpmord** and **firpm** function[2] give the length of 661 taps. The number of coefficients is huge as expected. ■

2 Multirate solutions

2.1 Single-stage design

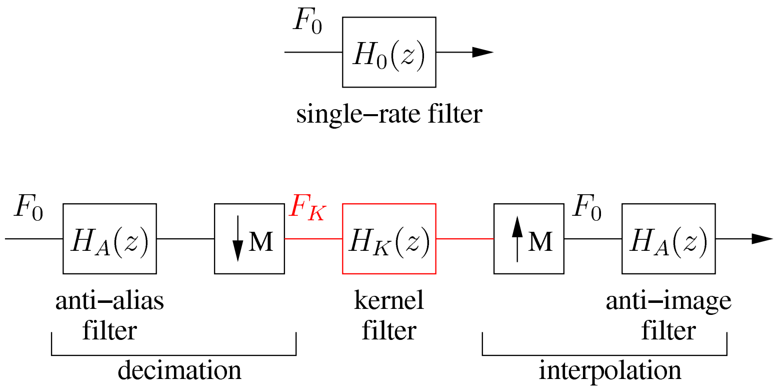
The number of coefficients required to implement narrowband FIR filtering can be reduced by multirate techniques. The idea is to reduce the sample rate by decimation, perform the required filtering at this lower sample rate (it is not any more narrowband), and then increase the sample rate to the original level by interpolation.

Reduction of the sample rate can be done using one or more decimation stages. A single-stage configuration is illustrated in Fig. 1(a). The cutoff frequencies of the kernel filter $H_K(z)$ correspond to the cutoffs of the original filter $H_0(z)$, f_p and f_s. The same filter $H_A(z)$ is used for both antialiasing and anti-imaging. Its passband cutoff is f_p, and the stopband cutoff is chosen so that there are no high-frequency components that can affect the passband of interest. This is illustrated in Fig. 1(b). The maximum stopband cutoff frequency of $H_A(z)$ can be the sample rate of the kernel filter, F_K, minus the stopband frequency f_s.

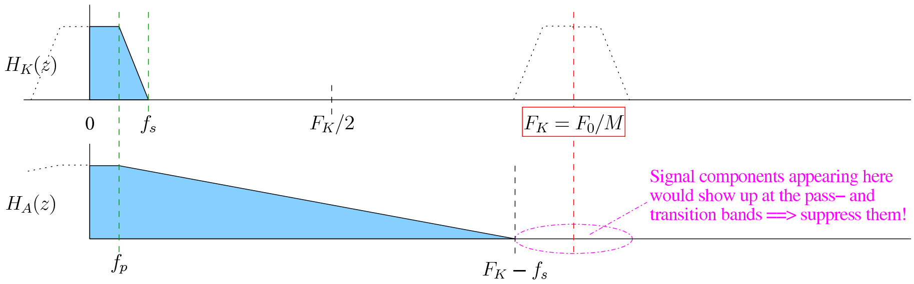
What comes to the choice of the passband ripple, it is obvious that the ripples of $H_K(z)$ and $H_A(z)$ must be lower than the value specified for $H_0(z)$, that is, the

[1]Note: we have the same problem if the stopband is narrow.

[2]These functions are used in the tool script used in the exercise.



(a) Block diagrams (F_0 and F_K denote the sample rates).



(b) Cutoff frequencies of the kernel and antialias/anti-image filters.

Figure 1: Multirate filter design.

ripple is distributed to each of them. The stopband attenuation of $H_K(z)$ is the value specified for $H_0(z)$. The stop-band attenuation of $H_A(z)$ cannot be determined from the corresponding attenuation in the overall filter as it is also related to suppression of aliasing and imaging. So, some experimentation may be needed in order to find out sufficient attenuation. The same holds for the passband ripples.

In the final implementation of the multirate filter design, the number of multiplications and additions can be reduced further by applying efficient techniques to decimation and interpolation stages. The result is illustrated in Fig. 2. As filter coefficients are distributed over component filters $H_\lambda(z)$ ($\lambda = 0, \ldots, M-1$), the number of multiply-accumulate (MAC) operations is $1/M$ times the original value.

Example 2. Let us reconsider implementation of the narrowband lowpass FIR filter of Example 1. For a corresponding multirate design with downsampling factor $M = 4$, the kernel filter $H_K(z)$ is specified as

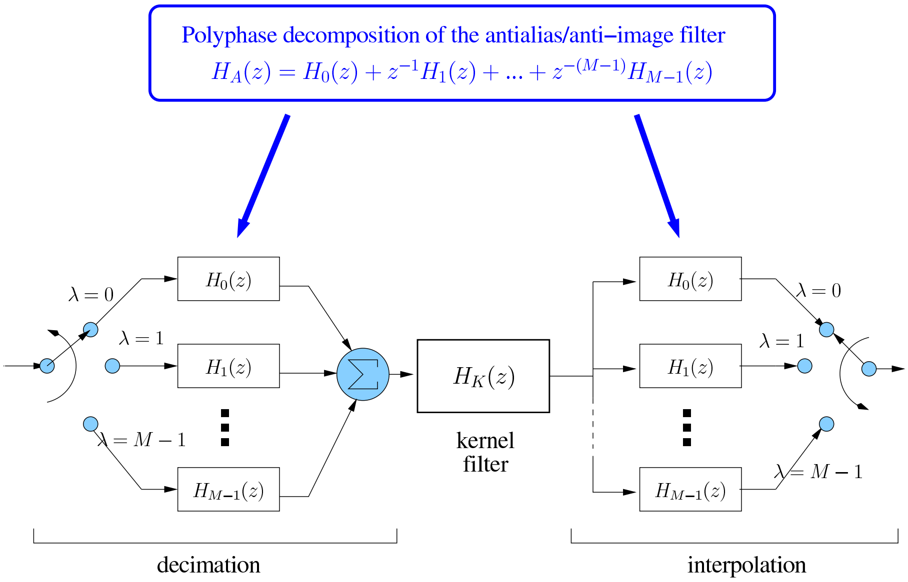


Figure 2: Multirate filter with efficient decimation/interpolation stages.

Sampling frequency $F_K = F_0/M$	12 kHz
Passband cutoff frequency f_p	1000 Hz
Stopband cutoff frequency f_s	1200 Hz
Maximum passband ripple ($< A_p$)	**0.03 dB**
Minimum stopband attenuation A_s	**61 dB**

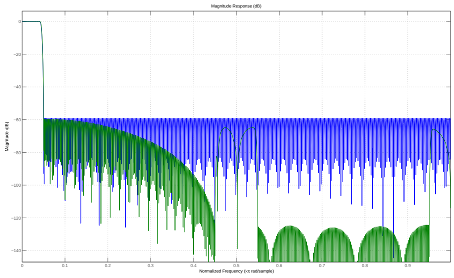
In the specification, passband ripple is lowered from the original value in order to allow ripple in antialias and anti-image filters. **firpm** gives a filter whose length is 191 taps. Compared to Example 1, the length of the filter is much lower.

The passband ripple and stopband attenuation of antialiasing and anti-imaging filters $H_A(z)$ are tuned by trial-and-error, and the final choice is to use

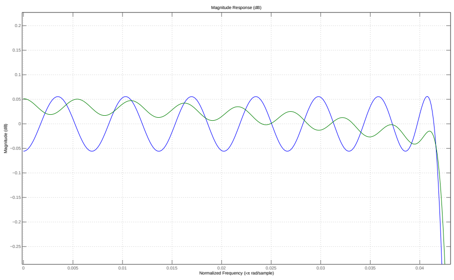
Sampling frequency $F_A = F_0$	48 kHz
Passband cutoff frequency f_p	1000 Hz
Stopband cutoff frequency $F_K - f_s$	10800 Hz
Maximum passband ripple ($< A_p$)	**0.02 dB**
Minimum stopband attenuation	**65 dB**

With this specification, **firpm** gives a filter whose length is 16 taps.

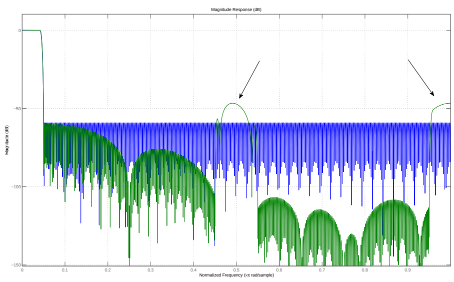
The verification provided in Figs. 3(a) and 3(b) shows the frequency responses of the original single-rate design $H_0(z)$ and the multirate solution, which confirms that the filter works. If the stopband attenuation erroneously is reduced to 50 dB there are aliasing/imaging effects as demonstrated in Fig. 3(c).



(a) Overall magnitude responses.



(b) Passband responses.



(c) Example of stopband failure due to wrong attenuation in $H_A(z)$.

Figure 3: Comparison of single-rate (blue) and single-stage multirate (green) designs.

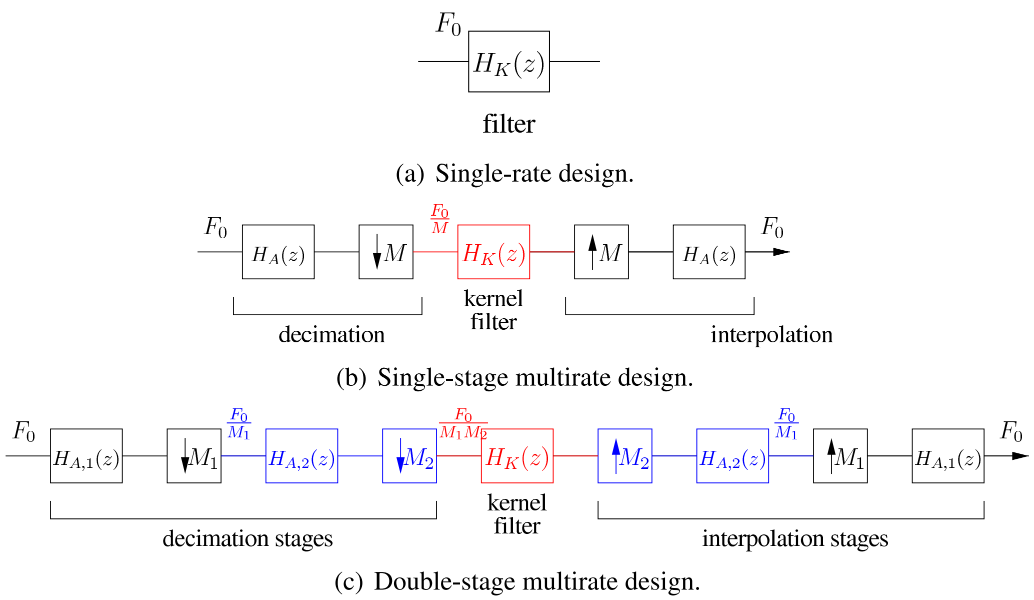
Computational complexity in terms of numbers of multiply-accumulate (MAC) operations is shown in the table below. The single-rate design is the one provided in Example 1. The first multirate design (Multirate/1) does not apply the efficient decimation and interpolation shown in Fig. 2, but already achieves considerable decrease in the rate of MACs. Multirate/2 shows the complexity with the commutator based implementation. The multirate design efficiency is improved about 30% compared to Multirate/1.

Implementation	Rate of MACs	MAC/s	Ratio
Single-rate	$661 \times F_0$	31,728M	100%
Multirate/1	$191 \times F_0/4 + 2 \times 16 \times F_0$	3,828M	12.1%
Multirate/2	$191 \times F_0/4 + (1/4) \times 2 \times 16 \times F_0$	2,676M	8.4%

■

2.2 Multistage design

In practice, decimation and interpolation can be done gradually using multiple stages. Figure 4(c) shows an alternative consisting of two stages. Let us consider if any improvement can be obtained with this alternative in our case.



(a) Single-rate design.

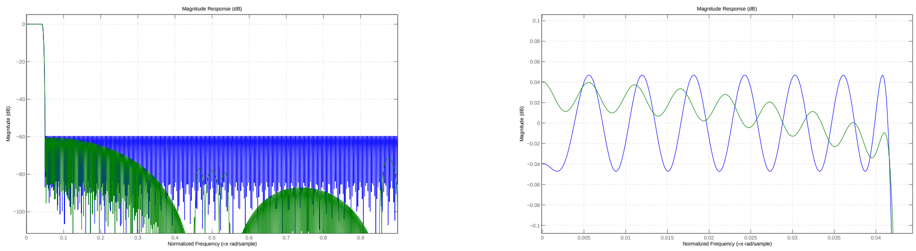
(b) Single-stage multirate design.

(c) Double-stage multirate design.

Figure 4: Implementation alternatives for the filter.

Example 3. In Example 2, $M = 4$ was used. Let us consider a double-stage solution, where $M_1 = 2$ and $M_2 = 2$. As $M_1 M_2 = 4 = M$, we use the kernel filter of Example 2. After some trial-and-error, we use for the first decimation stage $H_{A,1}(z)$ the antialias filter specified as

Sampling frequency $F_{A,1} = F_0$	48 kHz
Passband cutoff frequency f_p	1000 Hz
Stopband cutoff frequency $F_0/M_1 - f_s$	22800 Hz
Maximum passband ripple ($< A_p$)	**0.007 dB**
Minimum stopband attenuation	**78 dB**



(a) Overall magnitude responses.   (b) Passband responses.

Figure 5: Comparison of single-rate (blue) and double-stage multirate (green) designs.

Note that the stopband cutoff frequency is computed using F_0/M_1 and not $F_K = F_0/(M_1M_2)$. The length of the filter given by **firpm** is 6 taps. Compared to the $H_A(z)$ of Example 2 (15 tap filter), the length is shorter here due to wider transition band (from 1,000 to 22,800 Hz).

For the second decimation stage $H_{A,2}(z)$, we design the antialias filter using the specification

Sampling frequency $F_{A,2} = F_0/M_1$	24 kHz
Passband cutoff frequency f_p	1000 Hz
Stopband cutoff frequency $F_K - f_\mathrm{s}$	10800 Hz
Maximum passband ripple ($< A_\mathrm{p}$)	**0.007 dB**
Minimum stopband attenuation	**78 dB**

The length of the filter given by **firpm** is 7 taps.

Fig. 5 shows that the frequency response of the implementation fulfills the requirement. The complexity analysis of the design is provided below, where the implementations Multirate/1 and Multirate/2 correspond to the ones in Example 2.

Implementation	**Rate of MACs**	**MAC/s**	**Ratio**
Single-rate	$661 \times F_0$	31,728M	100%
Multirate/1	$191 \times F_0/4 + 2 \times 6 \times F_0 + 2 \times 7 \times F_0/2$	3,204M	10.1%
Multirate/2	$191 \times F_0/4 + 1/2 \times 2 \times 6 \times F_0 + 1/2 \times 2 \times 7 \times F_0/2$	2,748M	8.7%

According to Multirate/1, the double-stage implementation seems to be better. However, when the commutator based construction is used, the single-stage solution of Example 2 (2,652 MMAC/s, 8.4%) seems to be slightly better. Better tuning of the filter parameters might change the situation. ■

3 Design procedure

In this section, some extra information for doing the design task. For the work, a Matlab tool named **multirateN.m** is provided, which automates some tasks and makes the work easier. Here, we provide more information on choosing the filter parameters, and formulae for evaluating the computational complexity of filter implementations.

3.1 Choice of parameters

Cutoff frequencies. The passband cutoff frequency is always f_p. If the input sample rate to the decimation stage is F and its downsampling factor is M, then the stopband cutoff is $F/M - f_s$. A lower value is too tight and leads to an increase in the filter length. A higher value leads to aliasing.

In a two stage design, the stopband cutoff frequency of the first stage is

$$\frac{F_0}{M_1} - f_s$$

The sampling frequency of the filter is F_0. For the second stage, the stopband cutoff is

$$\frac{F_0}{M_1 M_2} - f_s$$

and the sampling frequency is F_0/M_1.

Passband ripples and stopband attenuations. Distribute the overall ripple given in the specification is, A_p, among filters. For example, in the two-stage design try first the ripple $A_p/5$ as there are five filters. Also all stopband attenuations can be equal to A_s first. Check out the passband and stopband, and reduce the values until the specification is fulfilled.

Choice of downsampling factors. In a design consisting of N downsampling stages, the product of the downsampling factors M_i, $M = \prod_{i=1}^{N} M_i$, must be less than

$$M_{\max} = \frac{F_0}{f_p + f_s}.$$

The reason for this is illustrated in Fig. 6, which also shows that it is not possible to use M close to $M_{\max}$. A rule of thumb is that an appropriate value is somewhere close to the *half of* $M_{\max}$. For example, if $M_{\max} = 15$ then it may be a good idea to try out a two-stage design where $M_1 = 2$ and $M_2 = 4$. Note that the values of M_i do not have to be equal in a multistage design.

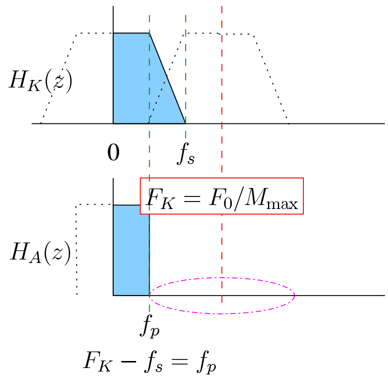


Figure 6: Situation corresponding to Fig. 1(b), when $M = M_{\max}$. Considering the kernel filter $H_K(z)$, there is not yet passband aliasing as $F_K - f_s = f_p$. On the other hand, we see that for a single-stage design it is not possible to realize $H_A(z)$ as the transition band width (step from f_p to $F_K - f_s$) is zero. From the figure, we infer that $F_K > f_p + f_s$ from which the rule follows.

3.2 Complexity analysis

Rules. The rules for evaluating the complexity of multirate filter designs are:

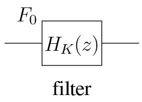
1. Computational complexity of the kernel filter is the product of the input sample rate F and the filter length N.

2. Computational complexity of the decimation filter is the product of the input sample rate F, the filter length N, and the inverse of the downsampling factor M. The last term is due to the use of the commutator.

3. Computational complexity of the interpolation filter is the product of the output sample rate F, the filter length N, and the inverse of the upsampling factor M. Again, the last term is due to the use of the commutator.

The computational requirement of the corresponding decimation and interpolation stages is equal. These rules are applied in the following where we computate of the number of MAC operations per second, R_{mac}, for different kinds of implementations. The input sample rate is F_0.

1. Single-rate design.

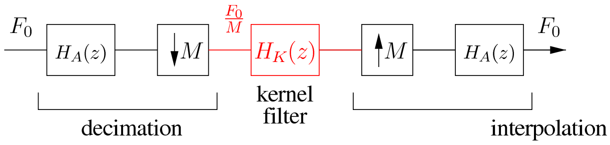


The complexity is

$$R_{\text{mac}} = F_0 \times N_K \tag{1}$$

where N_K is the number of filter coefficients.

2. Single-stage multirate design.

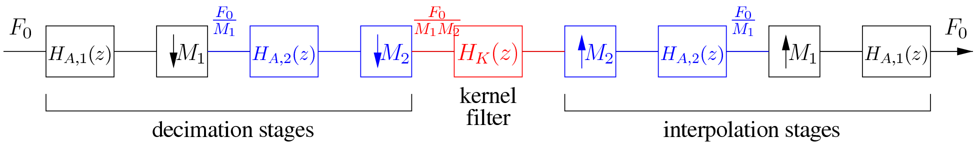


A design with single decimation/interpolation stage has complexity

$$\begin{aligned} R_{\text{mac}} &= F_0 \times N_A \times (1/M) && \text{decimation} \\ &\quad + (F_0/M) \times N_K && \text{kernel} \\ &\quad + F_0 \times N_A \times (1/M) && \text{interpolation} \\ &= F_0 \times (N_K/M + 2N_A/M) \end{aligned} \tag{2}$$

where N_K is the length of the kernel filter, N_A is the length of the anti-alias and anti-image filters, and M is the downsampling/upsampling factor.

3. Double-stage multirate design.



A design with two decimation/interpolation stages has complexity

$$\begin{aligned} R_{\text{mac}} &= F_0 \times N_{A,1} \times (1/M_1) && \text{decimation 1} \\ &\quad + (F_0/M_1) \times N_{A,2} \times (1/M_2) && \text{decimation 2} \\ &\quad + (F_0/(M_1 M_2)) \times N_K && \text{kernel} \\ &\quad + (F_0/M_1) \times N_{A,2} \times (1/M_2) && \text{interpolation 2} \\ &\quad + F_0 \times N_{A,1} \times (1/M_1) && \text{interpolation 1} \\ &= F_0 \times (N_K/(M_1 M_2) + 2N_{A,1}/M_1 + 2N_{A,2}/(M_1 M_2)) \end{aligned} \tag{3}$$

where N_K is the length of the kernel filter, $N_{A,1}$ and $N_{A,2}$ are the lengths of the anti-alias and anti-image filters, and M_1 and M_2 are the downsampling/upsampling factors.

4 Summary

It has been shown above how sample rate changes (downsampling/upsampling) can be used to find out more efficient narrowband FIR filter implementation, and how noble identities combined with polyphase decomposition can be used to improve efficiency of decimation and interpolation. Multirate techniques can also be used to implement other kinds of filters such as wideband and narrow passband filters. For an overview of such implementations, see (Saramäki & Ramstad 2010). Moreover, multirate approach can also be applied to IIR filters. For example, see (Johansson 2003).

References

Johansson, H. (2003) Multirate iir filter structures for arbitrary bandwidths. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 50:1515–1529.

Saramäki, T. & Ramstad, T. (2010) Design of efficient FIR filters based on multirate and complementary filtering. https://homepages.tuni.fi/tapio.saramaki/Part4_short_article.pdf (accessed 18.11.2025) .