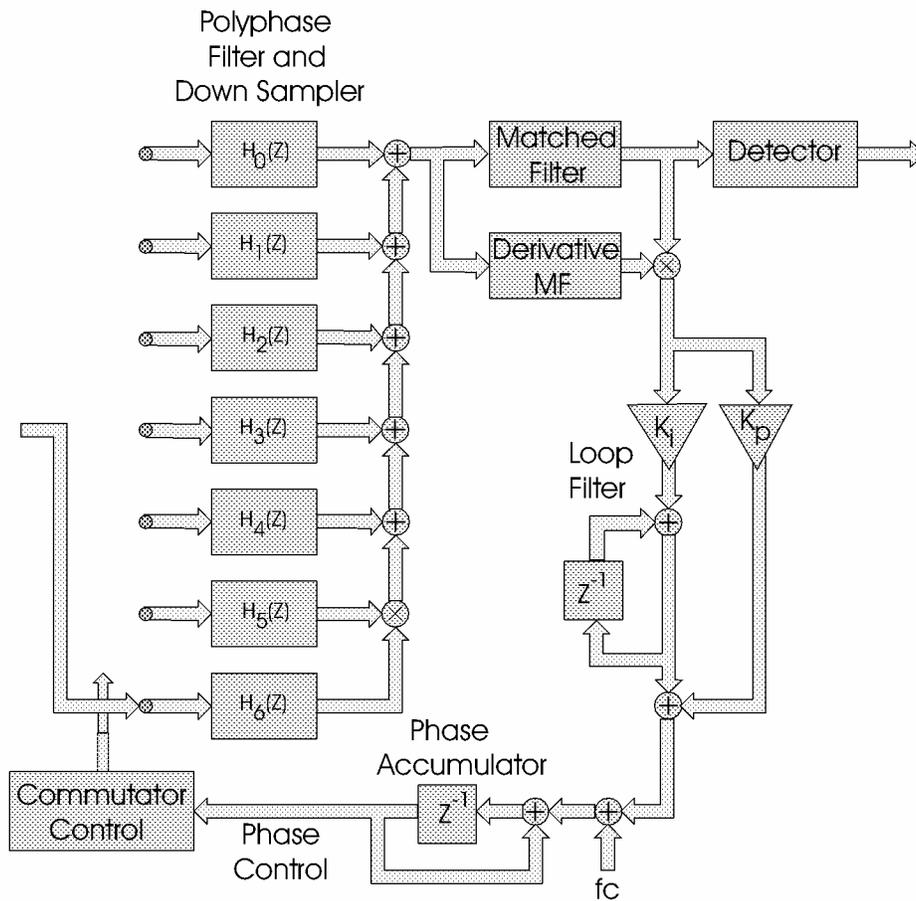


Communication Systems Applications



Communication systems make liberal use of multirate filters in several ways. Multirate processing finds application in shaping filters, in channelizers, in interpolators, in efficient bandwidth and sample rate reduction schemes, in anti-alias filtering, and in many other applications. This chapter deals with applications that fall under the other category. It is a potpourri of applications we have found over many years of applying signal processing techniques to communication systems. Some of these applications are standard and are well known among practitioners but are included to bring the neophyte into the inner circle. Other applications are unique and will quicken the heart of even the most seasoned practitioner.

13.1 CONVENTIONAL DIGITAL DOWN CONVERTERS

A radio receiver down converts and demodulates a narrowband radio frequency (RF) signal embedded in a block of frequencies assigned to a particular radio service. For instance, the commercial FM band spans the frequency span from 88 MHz to 108 MHz with multiple 200-kHz channels. Similarly cable TV modems select, down convert, and demodulate 5-MHz symbol rate QAM signals spanning 54 to 216 MHz with 6 MHz channels interspersed with legacy frequency gaps. The traditional architecture of a radio receiver that performs this task is shown in Figure 13.1. This standard architecture performs two frequency translations and is called a dual-conversion receiver. The receiver passes the input signal through an image reject filter, amplifies, and then down converts a selected RF channel to an intermediate frequency (IF) filter that performs initial bandwidth limiting. The output of the IF filter is again down converted to baseband by matched quadrature mixers that are followed by matched analog baseband filters that perform final bandwidth control. Each of the quadrature down-converted signals is then converted to its digital representation by a pair of matched analog-to-digital converters (ADC). The output of the ADCs is processed by DSP engines that perform the required synchronization, equalization, demodulation, detection, and channel decoding.

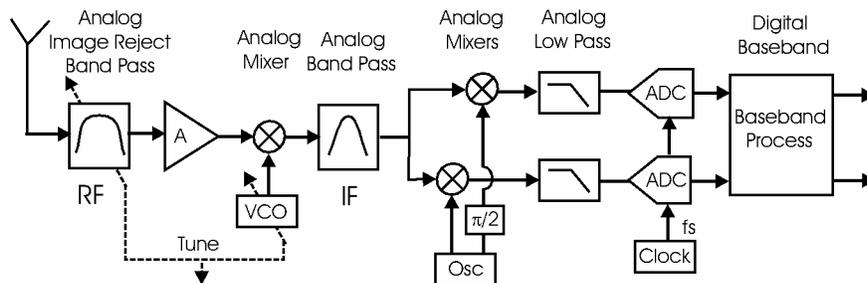


Figure 13.1 Standard Radio Receiver Architecture

Gain and phase imbalance between the two paths containing the quadrature mixers, the analog baseband filters, and the ADCs in the receiver are the cause of cross talk between the in-phase and quadrature (I/Q) components. In addition, the ADCs inject a DC term in the center of the baseband signal, and the analog filters introduce group delay distortion. Adaptive algorithms can remove the imbalances, the DC terms, and the phase distortion as background processing tasks in the DSP segment of the receiver. Rather than repairing the analog defects in the DSP domain, we have high motivation to avoid these distortion effects entirely by performing the entire baseband processing task in the DSP domain. Besides the advantages of avoiding performance degradation due to path imbalance due to analog component tolerance, avoiding performance degradation due to component parameter drift with time and temperature, avoiding the cost of quadrature mixers, and avoiding group delay distortion associated with analog filters, DSP insertion also offers the attraction of flexibility related to filters with programmable bandwidth and sample rates.

Figure 13.2 presents the block diagram of a second generation receiver in which the conversion from analog to digital occurs at IF rather than at baseband. Examining the receiver, we note two significant changes in the processing stream. First, due to the higher center frequency of the IF signal, the ADC must operate at a higher sample rate than in the baseband version. Second, we see that the down conversion of the selected channel is performed by a digital down converter and digital low-pass filter which, because of the higher sample rate, now includes a resampling operation. We are willing to accommodate this extra processing burden to gain the advantage that the DSP-based down conversion is free of imbalance-related distortion terms. A second advantage of the digital translation process is that the digital filters in the process are designed to have linear phase characteristics, a characteristic trivially simple to realize in digital nonrecursive filters. Another option we will examine shortly is the ability to embed the down conversion in the resampling process.

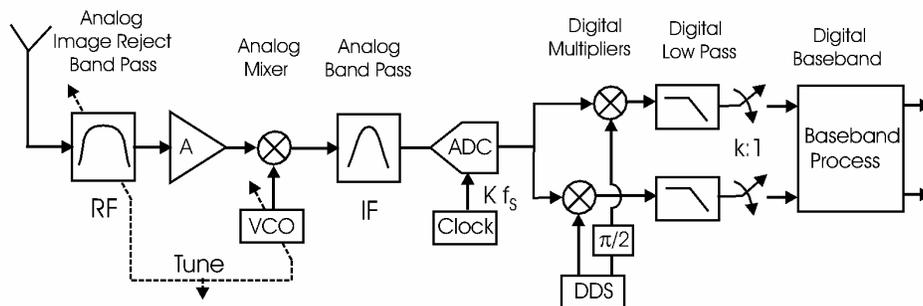


Figure 13.2 Second-Generation Radio Receiver Architecture

To control the computational workload, the filtering and down sampling is usually performed in two stages, a 4- or 5-stage CIC filter that performs filtering without multiplications while performing an internal M-to-1 down sampling to an output rate of $4f_s$. The CIC is followed by two or more half-band filters that correct for the main lobe gain of the CIC

while rejecting the main lobe spectral region containing significant aliased energy due to the M-to-1 down sampling. Figure 13.3 shows a possible realization of the standard second-generation receiver front end implemented by a digital down converter. The converter performs a cascade of simple operations consisting of a quadrature mixer driven by a direct digital synthesizer (DDS), a K-stage CIC filter capable of large integer resampling of M-to-1, a 2-to-1 half-band filter with CIC compensation, a second 2-to-1 half-band to finish the spectral control, and a course gain control.

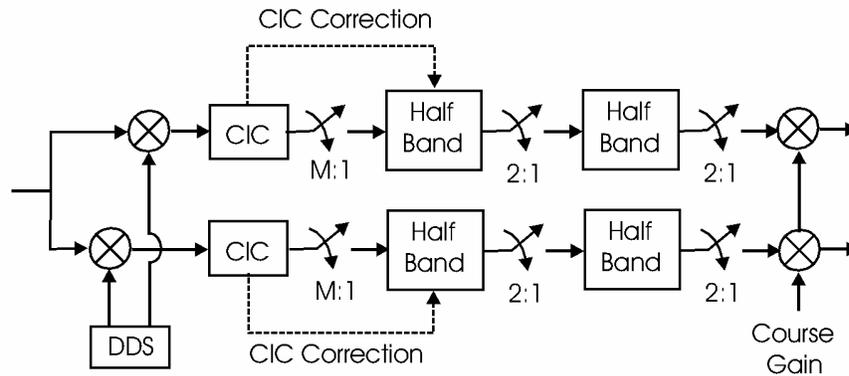


Figure 13.3 Standard Digital Down Converter with CIC and Half-band Filters

Figure 13.4 shows the log magnitude frequency response of a 5-stage CIC filter operating as a 10-to-1 resampling filter. The spectrum is shown at the input and output sample rates. The spectrum at the output rate shows the aliased main-lobe folding back into the main lobe spectral interval due to the 10-to-1 down sampling. The alias-free region, approximately one-fourth of the output sample rate, is extracted from the main lobe by the half-band filters following the CIC. These filters not only extract the desired bandwidth, they correct the droop in pass-band gain due to the curvature of the CIC main-lobe response. Shown, as an overlay on the main lobe, is the desired response of the half-band filters following the CIC filter.

Figure 13.5 shows the frequency response of the CIC filter main lobe along with the frequency response of the compensating first half-band filter and their composite response. Note the spectral peaking of the compensated spectrum beyond the pass-band edges. This peaking is suppressed by the frequency response of the second half-band filter following the compensating half-band filter. The compensating half-band filter was designed using the Remez algorithm with the minor modification that the pass-band gain is the reciprocal of the CIC gain over the normalized frequency interval 0 to 0.5. The gains were set over a grid spanning the normalized frequency interval, and the Remez algorithm does linear interpolation between the grid points. Rather than measure the CIC gain, we computed it at the desired grid points using the first three terms of its Taylor series as indicated in (13.1).

$$gg = \left(1 - \frac{1}{6}\theta^2 + \frac{1}{120}\theta^4 - \frac{1}{5040}\theta^6\right)^5 \quad (13.1)$$

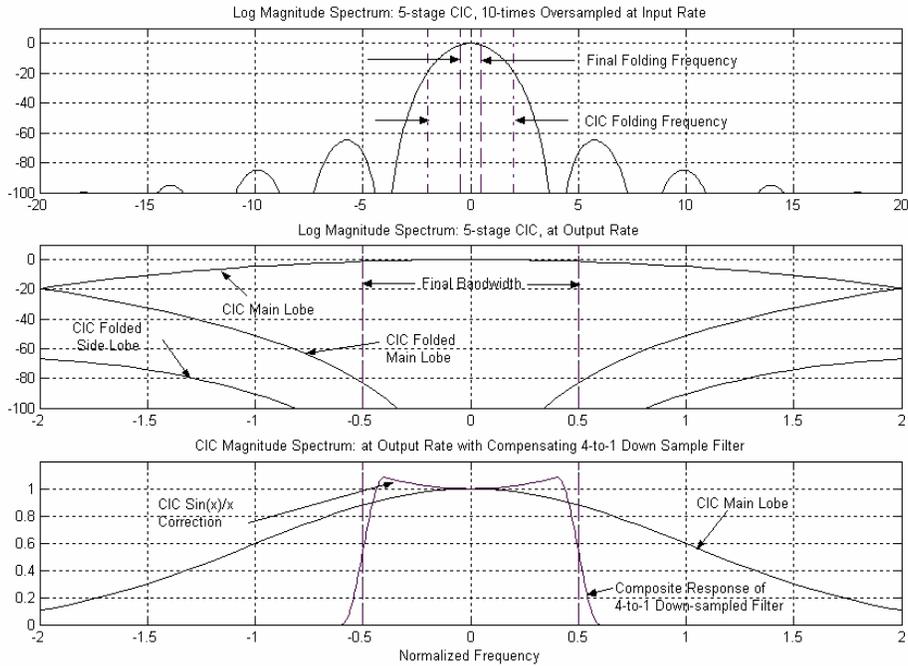


Figure 13.4 Frequency Response of 5-stage CIC Filter at Input Sample Rate, and at Output Sample Rate Illustrating Main-lobe Folding due to 10-to-1 Re-sampling, and Main-lobe Response with Overlaid Compensating 4-to-1 Down-sample filter

The MATLAB call to the Remez routine using this expression for the desired pass-band gain is shown here.

```
phi=[0.00 0.01 0.011 0.02 0.021 0.03 0.031...
      0.04 0.041 0.05 0.051 0.06 0.061...
      0.07 0.071 0.08 0.081 0.09 0.091 0.10];
phi=phi*pi;
tt=((1-(phi.^2)/6+(phi.^4)/120-(phi.^6)/1540)).^5;
hh=remez(20,[phi 0.4 0.5]/0.5,[1./tt 0 0]);
```

MATLAB Call for CIC Compensating Half-band Filter

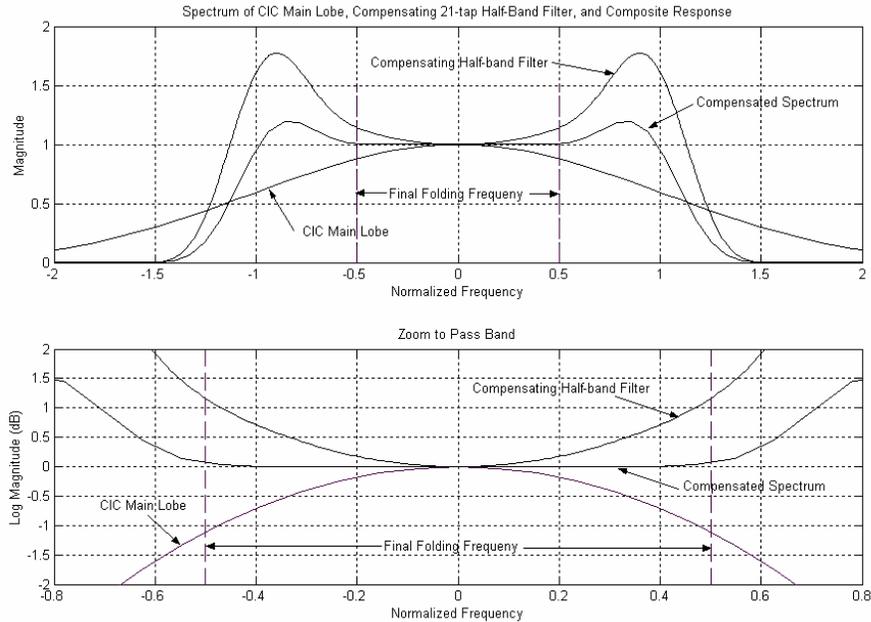


Figure 13.5 Spectra of CIC Main Lobe, of 21-tap Compensating Half-band Filter, and the Composite Response

13.2 ALIASING DIGITAL DOWN CONVERTERS

In the conventional sampled data collection process, the sample rate is selected to satisfy (13.2) where $2f_{BW}$ is the two-sided bandwidth of the analog anti-alias filter and f_{Δ} is the transition bandwidth of the same filter.

$$f_s = 2f_{BW} + f_{\Delta} \quad (13.2)$$

Signals collected from the output of the anti-alias filter at this data rate are said to satisfy the Nyquist sampling criterion. The Nyquist criterion is sometimes stated as: “The sample rate must be greater than twice the highest frequency of the input signal.” This is an over restrictive or a narrow interpretation of the Nyquist criterion. The less restrictive interpretation is that the sample rate must exceed the two-sided bandwidth of the signal. This second interpretation is important when we have a narrow bandwidth signal centered on a high frequency carrier. In a digital receiver, the signal processing following the data collection proc-

ess removes the carrier to extract the complex envelope of the narrowband signal on the carrier. A digital down-conversion process normally performs this task. Since the carrier frequency is discarded as part of the signal extraction, there is no need to preserve it during the data-sampling process. We are thus free to violate the Nyquist criterion for the carrier frequency as long as we satisfy the criterion for the bandwidth of its complex envelope.

This narrowband interpretation of the Nyquist criterion leads to an alternate data collection process known as subsampling or IF sampling. In this process the sample rate is selected to be less than the signal's center frequency to intentionally alias the center frequency to a lower frequency less than the sample rate. Since we are intentionally violating the Nyquist sampling criterion we must condition the analog signal to prevent multiple frequency intervals from aliasing to the same frequency location to which our desired signal component will alias. To minimize the cost of the analog signal-conditioning filter we arrange for the signal band of interest to alias to one-fourth of the selected sample rate. Aliasing to the quarter-sample rate maximizes the separation between the positive frequency alias and the negative frequency alias, which permits the maximum transition bandwidth of the analog band-pass filter. Aliasing the center frequency f_c to the quarter sample rate during the sampling process is assured if the sample rate satisfies (13.3). The $k + 1/4$ option aliases the signal to the positive quarter-sample rate while the $k - 1/4$ option aliases the signal to the negative quarter-sample rate. Use of the two options simply makes available a larger set of possible sample rates with either option equally acceptable.

$$f_c = kf_s \pm \frac{1}{4}f_s \quad (13.3)$$

To better understand the process, we now present an example of IF subsampling with aliasing to the quarter sample rate.

13.2.1 IF Subsampling Example

Signal 2-sided Bandwidth	10 kHz
Center Frequency	450 kHz
Signal Dynamic Range	80-dB
Output Sample Rate	20 kHz

First Option: Our first option is to sample the input signal at 2.0 MHz, placing the band of interest near the quarter-sample rate, then using a standard digital down converter we would perform a complex translation to baseband, filter and down sample 25-to-1 with a CIC, and then filter and down sample 4-to-1 with a pair of half-band filters.

Second Option: Our second option, the focus of this example, is to perform IF sampling at a sample rate satisfying (13.3). A list of possible sample rates that satisfy (13.3) for the center frequency of 450 kHz is presented in Table 13-1.

Table 13-1 List of Possible Sample Rates that Alias 450 kHz to Quarter-sample Rate

Integer k	fs for $450 = (k + 1/4) fs$	fs for $450 = (k - 1/4)fs$
0	1800	1800
1	360	600
2	200	257.143...
3	138.46...	163.37...
4	105.88...	120

We will select the 200-kHz sample for this example and suggest that the reader consider how the solution would be different had we selected 120-kHz as the sample rate. The aliasing caused by sampling the signal centered at 450 kHz at a 200-kHz sample rate is illustrated in Figure 13.6. Also note in this figure the transition bandwidth of the analog IF filter is maximized when the alias is to the quarter-sample rate. Figure 13.7 presents the same spectrum on a frequency-scaled periodic circle. Here we can start at 0 and travel in the positive direction around the 200-kHz circle passing the listed frequencies. After two encirclements, we see that frequency 450 kHz is located at the same location as 50 kHz. This is the desired result that we expected from the IF sampling process.

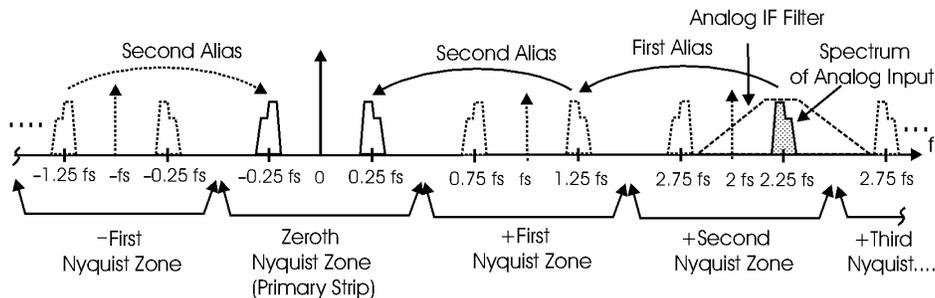


Figure 13.6 Aliasing Spectra at 2.25 fs in Second Nyquist Zone to 0.25 fs in Zeroth Nyquist Zone by IF Sampling at fs

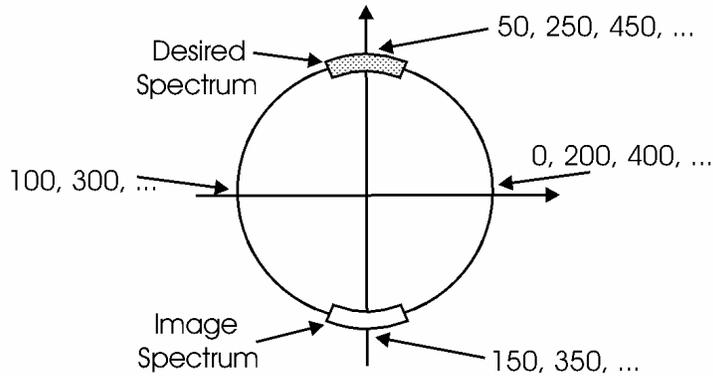


Figure 13.7 Periodic Spectra on Scaled Circle Showing How 450 kHz from Second Nyquist Zone Folds through 250 kHz in First Nyquist Zone to 50 kHz in Zeroth Nyquist Zone

At this point we could return to the standard digital down converter that can translate the spectral region to baseband with a complex heterodyne followed by a 10-to-1 resampling filter. This filter can be a 5-to-1 CIC followed by a 2-to-1 compensating half-band filter. The complex heterodyne required to move the spectrum from the quarter-sample rate is of the form shown in (13.4). The values of the cosine and sine terms in (13.4) are trivially the +1 and -1 and zero values shown in (13.5), so that in fact the heterodyne is trivial.

$$\exp(-j\frac{\pi}{2}n) = \cos(\frac{\pi}{2}n) - j \sin(\frac{\pi}{2}n) \quad (13.4)$$

$$\exp(-j\frac{\pi}{2}n) = \{1, 0, -1, 0, \dots\} + j\{0, -1, 0, 1, \dots\} \quad (13.5)$$

We also note that due to the zero-value samples of the heterodyne, half the data samples in the cosine product are set to zero, and the complementary set of the data samples in the sine heterodyne are also set to zero. These zero-valued terms cannot contribute to the outputs of the filter following the heterodyne. Since we know the location of these zeros, we can discard them without error as long as we account for their effect in shifting the non-zero data samples through the filter. It would be a shame to waste the unique attributes of this heterodyne and data variation on a standard digital down converter. Instead, we continue this example with a number of techniques that take advantage of the signal being located at the quarter-sample rate.

The most obvious option to apply at this point is a 4-path polyphase filter to down sample 4-to-1 and thus alias the quarter-sample rate to DC. A phase rotator at the output of the 4-stage filter would unwrap the aliases and extract samples corresponding to the band centered at the quarter-sample rate. By setting the center frequency of the sines and cosines

at the quarter-sample rate, the phase rotator sequence $\exp[j(\pi/2)r]$, for $r = 0, 1, 2,$ and 3 is a particularly simple sequence of $1, j, -1$ and $-j$. Merging the phase-rotated outputs of the polyphase filter stages proves to be a particularly simple task of identifying real and imaginary output sample, and forming the difference of the two real paths and of the two imaginary paths. The 4-path polyphase filter using these rotators is shown in Figure 13.8.

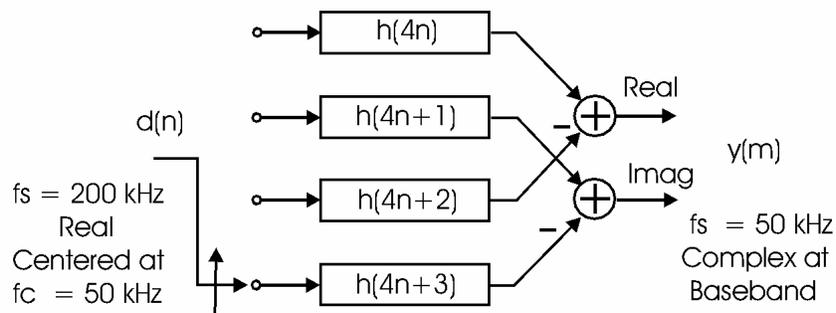


Figure 13.8 Four-path Polyphase Filter: Simultaneously Translates Frequency Band from Quarter-sample Rate to Baseband, Down Samples 4-to-1, and Converts Real Input to Complex Output

The remaining operation is to down sample the output of the 4-path polyphase filter from 50 kHz to the desired 20 kHz. This down-sampling task requires a 2.5-to-1 change in sample rate, which is accomplished as a 1-to-2 up sampling followed by a 5-to-1 down sampling. The 1-to-2 up sampling is performed conceptually by zero packing the input series 1-to-2. Of course the alternate zero-valued samples resulting from the zero packing cannot contribute to the filter output and, as observed earlier, can be discarded providing we account for their data shifting function in the 5-stage polyphase filter. Two successive cycles that deliver five zero-packed data samples to the 5-stage filter are shown in Figure 13.9.

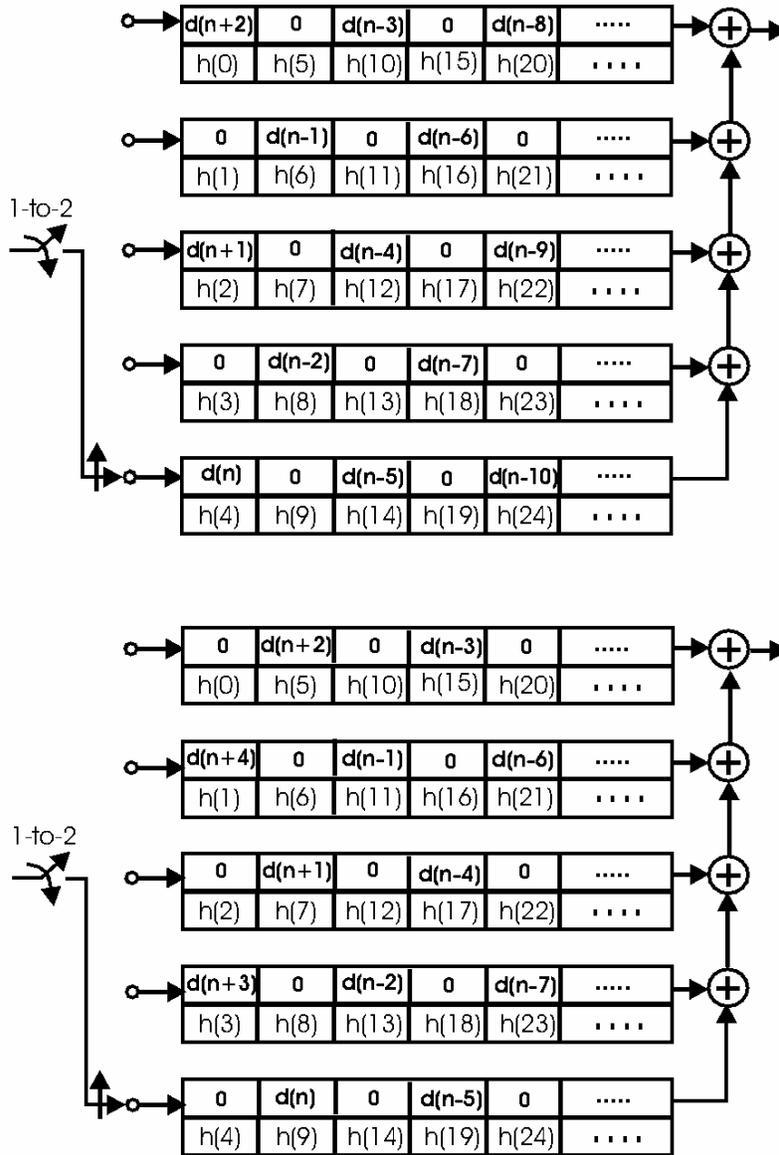


Figure 13.9 Indexing of Successive Zero-packed Data Samples to 5-path Polyphase Filter

Note that in the first cycle we deliver three non-zero samples $d(n)$, $d(n+1)$, and $d(n+2)$ and then compute an output. In the second cycle we deliver two non-zero samples $d(n+3)$ and $d(n+4)$ and then compute the next output. Over two successive input cycles, we deliver 5 inputs and compute 2 outputs for a resampling ratio of 5-to-2 or 2.5-to-1. The inserted zero-valued samples simply move the data samples in their paths one data sample to the right. We can omit the inserted zeros by following the 2-cycle state machine rules listed in Table 13-2.

Table 13-2 Two-state State Machine Input and Inner Product Schedule for 5-path 2.5-to-1 Resampling Filter

Cycle-0	Cycle-1
Input data: $d(n)$ in path-4 $d(n+1)$ in path-2 $d(n+2)$ in path-0	Input data: $d(n+3)$ in path-3 $d(n+4)$ in path-1
Inner products: path-4, data with $h(10n+4)$ path-3, data with $h(10n+8)$ path-2, data with $h(10n+2)$ path-1, data with $h(10n+6)$ path-0, data with $h(10n)$	Inner products: path-4, data with $h(10n+9)$ path-3, data with $h(10n+3)$ path-2, data with $h(10n+7)$ path-1, data with $h(10n+1)$ path-0, data with $h(10n+5)$

Figure 13.10 shows the block diagram of the down converter using undersampling to alias the narrowband spectrum to the quarter-sample rate and then a 4-path polyphase filter to alias it again to baseband for a final down sample of 5-to-2 with a 5-path polyphase filter. The first filter requires 16 coefficients, which means that each of the 4-paths require inner products with 4-coefficients while the second filter requires 40 coefficients that are split evenly between the two cycles of the state machine so that each of the 5-paths also require inner products with 4-coefficients.

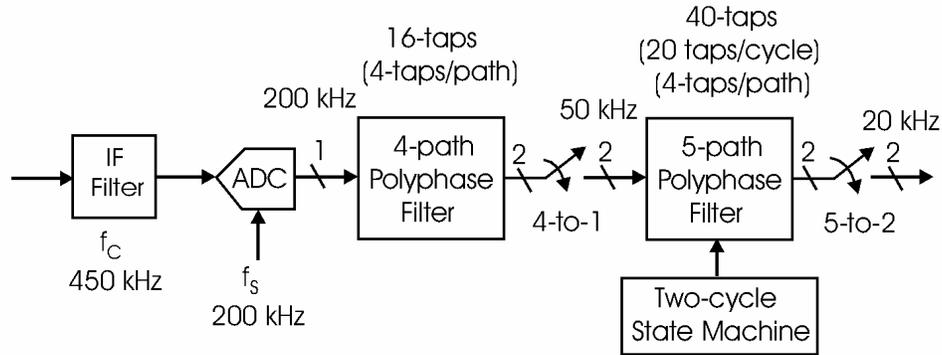


Figure 13.10 Signal Processing Structure of Down Converter Using Aliasing to and from Quarter-sample Rate by Subsampling and Polyphase Filtering

Third Option: In the example we are using to demonstrate subsampling we selected a sample rate of 200 kHz to translate the 450-kHz IF signal to the quarter-sample rate of 50 kHz. We selected the 200 kHz for the ease with which we could perform the remainder of the processing task, filtering and down sampling by a factor of 10 from 200 kHz to 20 kHz. We accomplished the 10-to-1 down sampling in two steps, first the 4-to-1 in the 4-path polyphase filter and then a 5-to-2 in the 5-path polyphase filter. We were guided to the 4-to-1 down sampler because it was natural to alias the quarter sample rate to baseband in this stage. A signal at any center frequency can be aliased to other center frequencies beside baseband under the resampling operation. The quarter-sample rate is quite remarkable in that it is only aliased to the four cardinal directions by other resample ratios besides the obvious 4-to-1. We now consider another option for this example in which we use a 5-to-1 first stage filter followed by a 2-to-1 second stage filter.

We can design a prototype low pass filter that will support a 5-to-1 resampling operation in a 5-path polyphase filter. In the standard way we would apply this filter, we would permit multiples of the output sample rate to alias to baseband where phase rotators would unwrap the desired alias. The problem here is that the signal of interest resides at 50 kHz, which is not one of the multiples of 40-kHz, so the standard polyphase structure appears to be inappropriate for this problem. By a slight modification of how we use the polyphase filter we are able to take advantage of the unique attributes of the quarter-sample rate center frequency. In particular, we translate the prototype low pass filter to the quarter-sample rate prior to the 5-path polyphase partition. We accomplish this translation by the trivial heterodyne of the filter coefficients with the terms $\exp[j(\pi/2)n]$. These heterodyne terms, the periodic sequence $\{1, j, -1, -j\}$, merely guide the output samples to the real or imaginary output and at most effect a sign change in the weighted sum performed in the filtering operation. This form of the filter is a close cousin of the Hilbert transform filter.

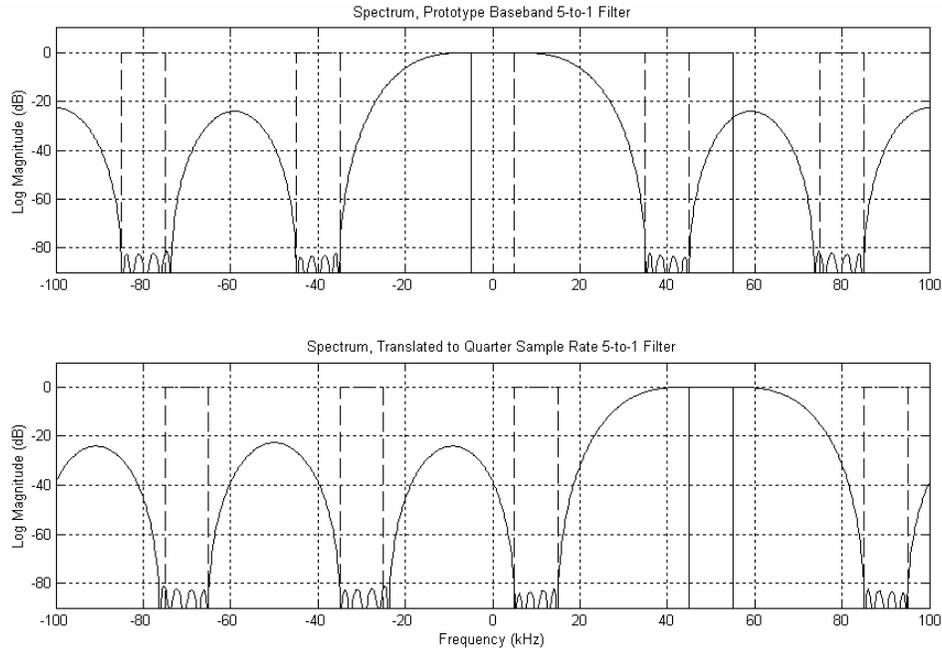


Figure 13.11 Spectrum of 25-tap Prototype Low pass Filter with Frequency Bands that Alias to Baseband when Resampled 5-to-1 (Upper Figure) and Spectrum of Same Filter Translated to Quarter-sample Rate with Frequency Bands that Alias to Quarter-sample Rate when Resampled 5-to-1 (Lower Figure)

When the filter is used in this manner, as a complex polyphase band-pass filter, the resampling operation exhibits an interesting property. This property is that a signal at the quarter-sample rate will always alias to a multiple of the quarter-sample rate under any integer-resampling operation. In particular, for our example, we have a signal centered at 50 kHz and sampled at 200-kHz so that it is located at the quarter-sample rate when it enters the 5-path polyphase filter. Here the signal is resampled 5-to-1 to 40 kHz which aliases the 50 kHz centered signal to 10-kHz, the quarter-sample rate at the output of the filter. Figure 13.11 shows the spectrum of a 25-tap baseband filter with the frequencies that alias to baseband under a 5-to-1 resampling as well as the translated to the input quarter-sample rate filter with the frequencies that alias to the output quarter sample rate when resampled 5-to-1.

We note that the sampled data sequence that enters the 5-path filter is real while the sampled data sequence that leaves the filter is complex and centered at the quarter-sample rate. This output sequence is heterodyned to zero by the trivial heterodyne $(-j)^n$ prior to final processing in the half-band filters. Figure 13.12 presents the block diagram of the signal processing performed by this option.

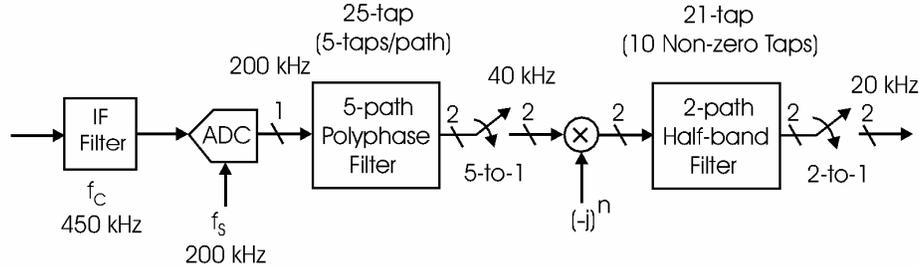


Figure 13.12 Signal Processing Structure of Down Converter Using Aliasing to Quarter-sample Rate by Subsampling and Again by 5-path Polyphase Filter

13.3 TIMING RECOVERY IN A DIGITAL DEMODULATOR

Multirate filters have had significant impact on communication systems, often replacing analog prototype systems with DSP based implementations. Filtering for bandwidth control and quadrature mixing for spectral translation are the obvious examples of this digital insertion. A second instance of this process is the timing recovery mechanism in digital communication receivers. In traditional analog systems, the phase of the sampling clock is controlled by a phase lock loop to obtain phase alignment with the symbol epochs in the received modulation signal. In modern digital systems, the phase alignment occurs not by moving the sample instances to the correct position in the time waveform, but rather by interpolation the waveform samples from the collected sample locations to the desired sample locations.

13.3.1 Background

The modulator of a communication system transmits a sequence of scaled and time translated band-limited waveforms of the form shown in (13.6). The scale factors a_n are selected from a finite list of permitted amplitudes in response to a sequence of binary words delivered at a periodic rate to the modulator. The alphabet may, for instance, contain two elements such as $+1$ and -1 selected by a 1-bit input, or it may contain four elements such as $+3$, $+1$, -1 , and -3 selected by a 2-bit input, and so on.

$$d(t) = \sum_n a_n h(t - nT) \quad (13.6)$$

In an ideal communication system, the demodulator receives a version of the transmitted waveform that has been corrupted by additive white Gaussian noise (AWGN) as shown in (13.7).

$$d(t) = \sum_n a_n h(t - nT) + \mathcal{N}(t) \quad (13.7)$$

To minimize the effects of the received noise in the ensuing decision process, the received noisy signal is passed through a matched filter. The impulse response of the matched filter is, as shown in (13.8), a time reversed and delayed version of the transmitted waveform $h(t)$. Here we assume the wave shape extends over k symbol durations.

$$g(t) = h(kT - t) \quad (13.8)$$

The convolution process performs a running weighted average with the filter's time-reversed impulse response. We purposely time-reversed the filter impulse response in anticipation of the time reversal that will occur in the convolution process. The reversal in the convolution process undoes the initial reversal so that the running weighted average is performed with a replica of the transmitted waveform and the convolver performs a replica correlation. The peak of the correlation function corresponds to the projection of the noisy signal on a noiseless replica of the signal, and this projection exhibits the maximum SNR that can be obtained by any processing of the received signal. In order for the receiver to access these peak values it invokes a timing recovery mechanism to align the frequency and phase of its sampling clock with the epochs of the correlation peaks. The signal flow of the entire process is shown in Figure 13.13.

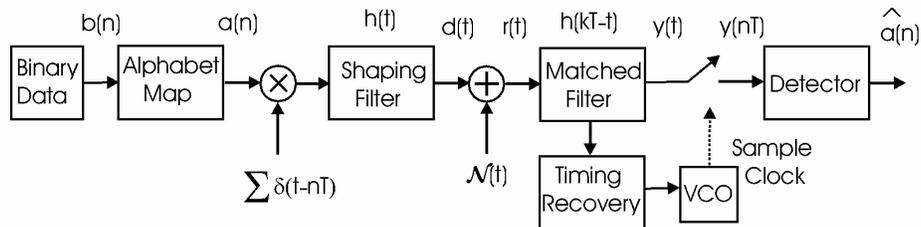


Figure 13.13 Signal Flow in Modulator and Demodulator for Communicating Through Band-limited AWGN Channel

In modern receivers, the matched filter operation is performed in the sampled data domain. The receiver structure changes slightly to accommodate the anti-aliasing filter, the S&H, and the ADC. A first-generation digital demodulator structure is shown in Figure 13.14. Note here that the timing recovery is now controlling the sampling clock to the digital matched filter rather than the sampling at the output of the analog matched filter. The sampling performed at the input to the digital filter must satisfy the Nyquist criterion for the collected wave shape, and in many systems the sample rate is two samples per symbol. The sampling performed at the output of the analog matched filter to supply samples to the detector occurs at the symbol rate or one sample per symbol.

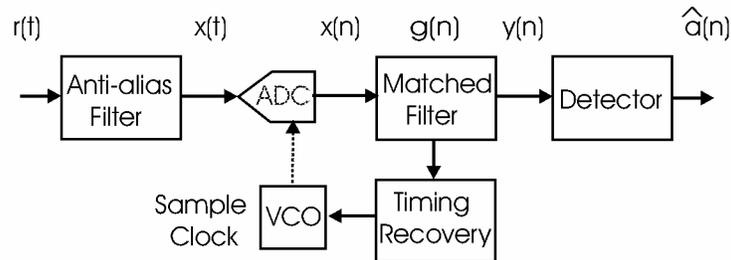


Figure 13.14 Signal Flow in First-generation Digital Demodulator

The timing recovery process must ascertain if the clock sample position is at the correct position or should be advanced or retarded relative to the input time waveform. It does this by posing the question, “How do I know that I am at the local peak of the correlation function?” It knows that at the peak, the correlation function has a zero derivative, so it poses the nearly equivalent question, “What is the derivative of the correlation function zero at this sample time?” Traditionally, auxiliary matched filter outputs, called early and late gates, that are time advanced and time delayed relative to the sample test point, supply an answer to this question as the average difference between the early and late gate output values. This can be visualized in Figure 13.15.

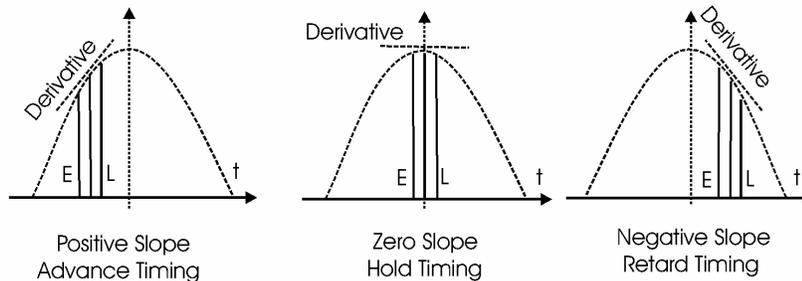


Figure 13.15 Three Samples, Early, Punctual, and Late Samples, on Correlation Function for Three Operating Conditions. (1) Positive Slope: Peak Is Ahead, (2) Zero Slope: At Peak, and (3) Negative Slope: Peak Is Behind

The derivative alone contains insufficient information to determine if the timing should be time advanced, held, or retarded. A conditional piece of information is missing, and this information is the answer to the question “What is the sign of this sample of the correlation function?” As seen in Figure 13.16, a sample set formed prior to the peak will generate a positive slope if the correlation values are positive but will generate a negative slope if the correlation values are negative.

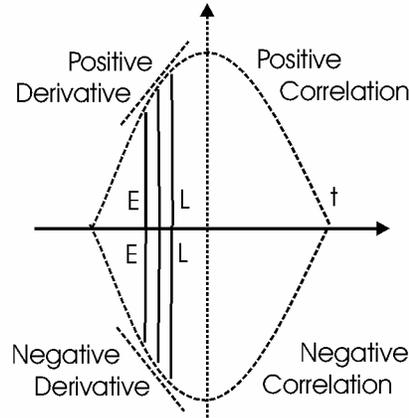


Figure 13.16 Three Samples, Early, Punctual, and Late Samples on Positive and on Negative Correlation

The conditional information is folded into the observable error parameter in the timing recovery process as the products shown in (13.9). When the SNR, the sign of the data sample $y(t)$, is reliable, the $\text{sign}[y(t)]$ is a sufficient modifier. When the SNR is small, the modifier is the data itself rather than the sign of the data. The maximum likelihood timing recovery process minimizes the product of the derivative matched filter output $\dot{y}(t)$ with SNR-conditioned selection of the matched filter output $y(t)$ or sign of the matched filter output $\text{sign}[y(t)]$. Many systems do not bother switching modes and simply use $y(t)\dot{y}(t)$ for the full range of SNR.

$$\begin{aligned} e(t) &= \dot{y}(t) \text{sign}[y(t)] \cong [y(t+\Delta t) - y(t - \Delta t)] \cdot \text{sign}[y(t)], & \text{High SNR} \\ e(t) &= \dot{y}(t) y(t) \cong [y(t+\Delta t) - y(t - \Delta t)] \cdot y(t), & \text{Low SNR} \end{aligned} \quad (13.9)$$

13.3.2 Modern Timing Recovery

Examining Figure 13.14 we note that the timing recovery process controls the phase and frequency of the voltage controlled oscillator (VCO) that is supplying the sampling clock to the ADC. It is likely that the error detector, the $y(t)\dot{y}(t)$ of (13.9), is performed in the sampled data domain as $y(n)\dot{y}(n)$ and that the loop filter that controls the transient and steady state behavior of the timing recovery system is also performed in the sampled data domain. The control signal formed to operate the VCO resides, as a set of numbers in the sampled data domain, while the control signal required to operate the VCO is an analog voltage level. To convert the digital control signal to an analog control signal requires a high-precision DAC, a low-bandwidth analog filter, and a bus and control mechanism to transfer the control words to the DAC's internal register.

Rather than incur the overhead of changing between the sampled data and continuous domains, we can perform the entire timing recovery process in the sampled data domain. We can accomplish this in two different ways. We can either move the data samples to be aligned with the filter coefficients, or we can move the filter coefficients to be aligned with the data samples. In the first method, we use an interpolator to raise the input sample rate by a factor of M , say 32, and then down sample back to the same input rate with a fixed time offset to the sample locations required to be aligned with the impulse response of the matched filter. Figure 13.17 presents the signal flow for this demodulator architecture. In the second method we increase the sample rate of the matched filter and then resample the filter response to the original sample rate with successive time offsets of $1/M$, $2/M$, $3/M$, etcetera, to form a set of M filters matched to different time offsets between the input sample location and the envelope of the received waveform. Figure 13.18 presents the signal flow for this demodulator structure. In both cases we are required to operate only one filter path out of the M -possible paths. In this mode, the interpolator is used as a 1-to- M up sampler followed by an M -to-1 down sampler with the desired time offset between input and output sample locations. The timing recovery process determines which filter path is the one required to align the filter with the signal time offset.

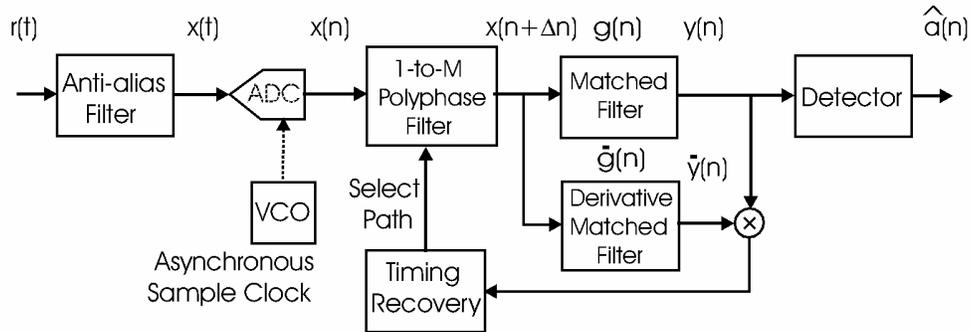


Figure 13.17 Signal Flow for Timing Recovery with Polyphase Interpolator Processing and Shifting Asynchronous Samples to Desired Time Locations

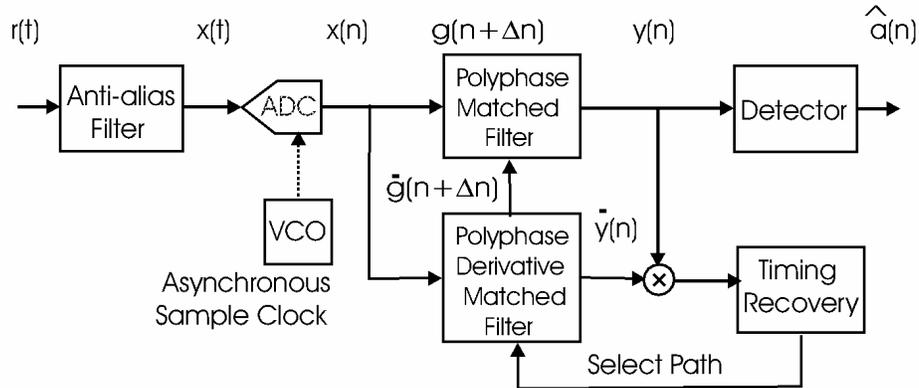


Figure 13.18 Signal Flow for Timing Recovery with Polyphase Matched Filter. Timing Recovery Selects Matched Filter Path Aligned with Input Sample Positions.

In Figure 13.15 we used the analog perspective that used an early and a late gate to determine the derivative at the output of the matched filter. Doing so requires the operation of three filters: early, punctual, and late. Note that the filters corresponding to the early and late gate filters are trivially the polyphase segments $(k - 1)$ and $(k + 1)$ when testing polyphase segment (k) . This is seen in Figure 13.19. As shown in (13.10), the same data is convolved with filters $(k - 1)$ and $(k + 1)$ to form corresponding outputs, which we then subtract to form the derivative estimate. Factoring out the common data, we are left with a filter with coefficients obtained as the difference of the two adjacent filters. We recognize this single filter as one that directly computes the desired derivative. We can now replace the early and late gate filters with the derivative filter and thus reduce the workload required to implement control data for the timing recovery process. Figure 13.20 shows a process using two banks of polyphase filters, one for the signal and one for the derivative, while Figure 13.21 shows a simple structure with two simple filters with a coefficient selection process. It is this later structure that is embedded in the block diagrams shown in Figures 13.17 and 13.18.

$$\begin{aligned}
 \dot{y}(n + k/M) &= y(n) * h_{k+1}(n) - y(n) * h_{k-1}(n) \\
 &= y(n) * [h_{k+1}(n) - h_{k-1}(n)] \\
 &= y(n) * \dot{h}_k(n)
 \end{aligned} \tag{13.10}$$

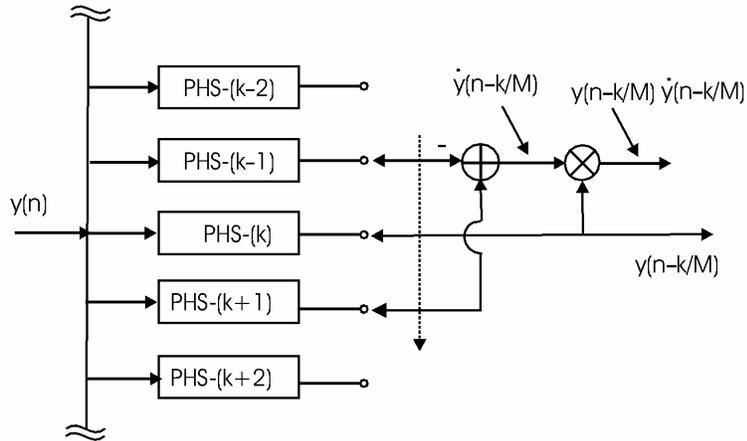


Figure 13.19 Early, Punctual, and Late Gate Filters for Timing Recovery Control Signals

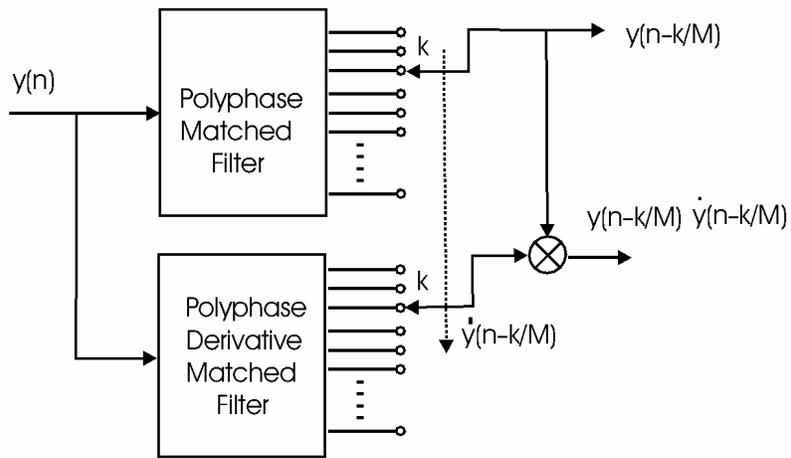


Figure 13.20 Two Polyphase Filter Banks Forming Filter and Derivative Filter Outputs

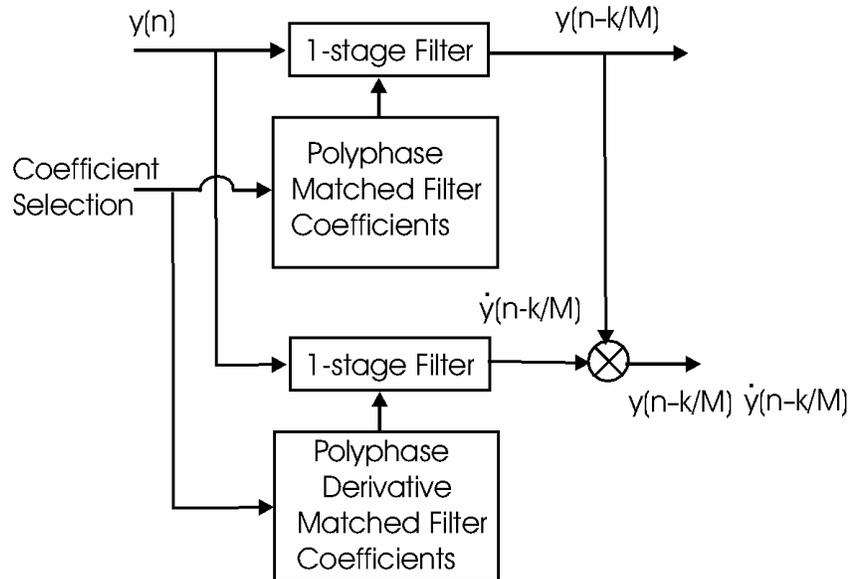


Figure 13.21 Two Single-stage Filters with Selection of Coefficient Sets from Polyphase Filter Bank for Matched Filter and Derivative Matched Filter

Figure 13.22 shows the time history of the address pointer in a modem using a 40-path polyphase-matched filter for timing recovery. Since the input signal is collected at two samples per symbol, one sample, the one with an even index for instance, is chosen as a data sample, and the other is the non data sample sometimes called the timing sample. The data sample is the one delivered to the detector. If the address pointer tries to cross the address boundaries, an address lower than 1 or greater than 40, the address wraps circularly and the input sample identified as the data sample is switched to the odd index. When this happens, the pointer resets from index 0 in interval 1 to index 40 in interval 2. You can see this pointer offset in the lower half of Figure 13.22. Beyond the scope of this discussion is the observation that there may also be data sample insertion (data stuff) or deletion (data skip) at the boundary crossing depending on the direction of the crossing and whether the data sample is an even or odd index input sample.

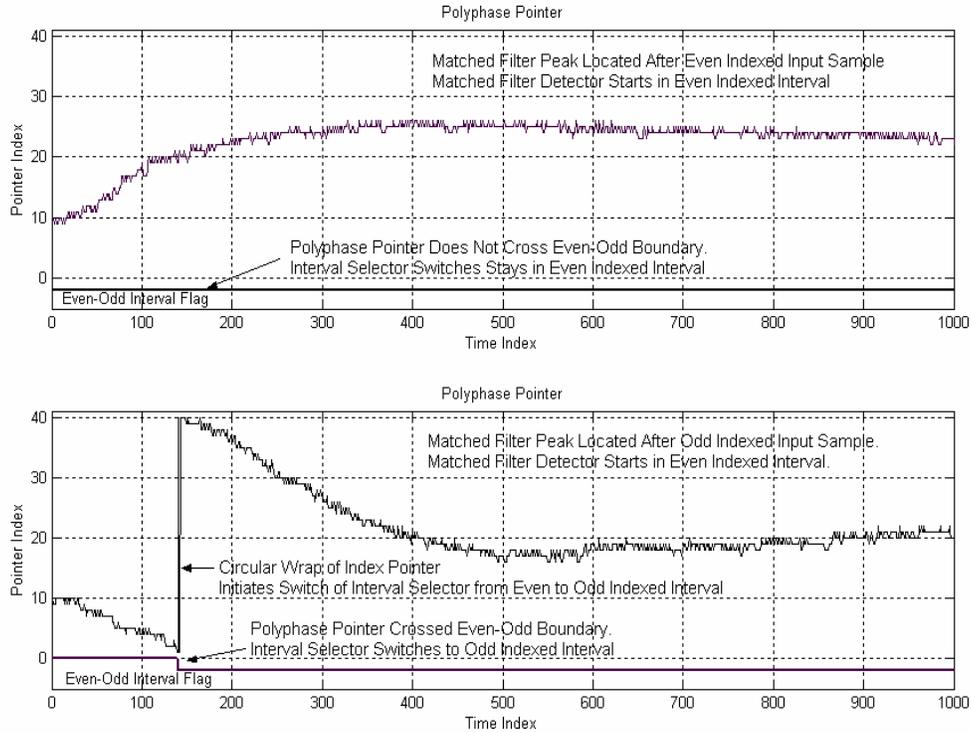


Figure 13.22 Polyphase Address Pointer for Timing Recovery Loop with 40-path Filter and Two Samples per Symbol. In Upper Figure, Pointer Does Not Cross Address Pointer Boundary: In Lower Figure, Pointer Crosses Address Pointer Boundary and Converts Address 0 to Address 40 and Switches from Even-indexed to Odd-indexed Data Sample.

13.4 MODEM CARRIER RECOVERY

Many communication systems use a complex heterodyne to translate a complex baseband communication signal to a center frequency in the band allocated to the particular radio service. We perform spectral translation to different center frequencies for a number of reasons. One reason is the sharing of the radio spectrum by multiple users through use of frequency division multiplexing. Another reason is the relative ease of designing analog circuits when the signal bandwidth occupies a small fraction of the center frequency. Yet another reason is access to smaller or more efficient antennae at higher frequencies as well as access to spectral bands with specific propagation characteristics.

The transmitter in the communication system performs the spectral translation of the baseband signal to the selected carrier center frequency by an up-conversion process. To cull the desired signal from the numerous channels sharing the spectral region the receiver must invert the spectral translation process. In order to accomplish this task, the receiver requires a phase-coherent replica of the unmodulated carrier delivering the signal. In early radio systems, a copy of the carrier was embedded in the modulated signal to enable the down-conversion process. In modern radios the receiver forms its own copy of the carrier from side information residing in the modulated signal. This process is called carrier recovery.

13.4.1 Background

At the simplest level, modulation is simply the translation of a complex baseband modulation envelope $m(t)$ with baseband spectral characteristic $M(\omega)$ to an arbitrary center frequency called the carrier. The relationship between the time and frequency descriptions of the baseband and translated signal is shown in (13.11).

$$\begin{aligned} m(t) &\Leftrightarrow M(\omega) \\ m(t) \exp(j\omega_C t) &\Leftrightarrow M(\omega - \omega_C) \end{aligned} \quad (13.11)$$

Product modulators that modify the amplitude and phase of the carrier perform this translation. To form the signal required for the actual transmission, the conjugate of the complex waveform is added to the expression shown in (13.11) to make the signal real. As shown in (13.12), the real signal is formed as the Cosine heterodyne of the real part of the complex envelope minus the Sine heterodyne of the imaginary part of the complex envelope. The real and imaginary parts of the complex waveform are normally called the In-Phase and Quadrature Phase or the I and Q components. These I, Q designations are inherited from the phase of their respective carriers.

$$\begin{aligned} s(t) &= m(t) \exp(j\omega_C t) + m^*(t) \exp(-j\omega_C t) \\ &= 2 \text{RL}[m(t)] \cos(\omega_C t) - 2 \text{IM}[m(t)] \sin(\omega_C t) \\ &= I(t) \cos(\omega_C t) - Q(t) \sin(\omega_C t) \end{aligned} \quad (13.12)$$

The narrowband signal is launched through the channel by the transmitter and is delivered through a noisy channel at the receiver. The receiver must remove the complex envelope from the carrier and present the noisy versions $I(t)$ and $Q(t)$ to the demodulator. To do so it must align the frequency and phase of its local oscillator to match the frequency and phase of the carrier in the received signal. The oscillator must do this in spite of temperature variation of its frequency-dependent components, in spite of manufacturing tolerance spread, in spite of Doppler-related frequency offsets due to a velocity component between platforms, and in spite of the fact that there is no spectral line at the carrier frequency in the

received signal. It accomplishes this feat with a carrier recovery loop formed around a phase locked loop (PLL). The loop is composed of a phase detector operating on demodulated data, a loop filter, and a controlled oscillator. The high-level block diagram of the up conversion and down conversion performed at the two ends of the channel is shown in Figure 13.23.

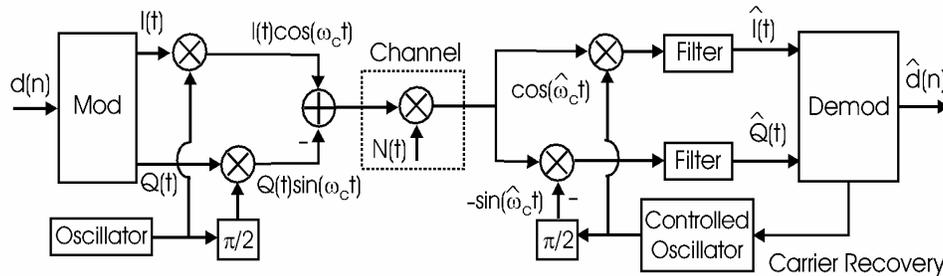


Figure 13.23 Block Diagram of Quadrature Up Converter at Transmitter and Quadrature Down Converter at Receiver.

The PLL has two distinct modes of operation: acquisition and tracking. The tracking bandwidth of a PLL, the range of frequencies over which it can follow frequency offsets, is limited by the control range of the oscillator. The acquisition bandwidth is limited by the pass band bandwidth of the loop filter. The acquisition range is considerably smaller than the tracking range. A receiver is directed to acquire an RF signal by being tuned to the expected center frequency of the signal.

Previous generation, or legacy design, receivers operate in the following manner: If the frequency offset between the local oscillator and the received center frequency is less than the acquisition bandwidth, the carrier recovery loop is able to generate a control signal to shift the local oscillator in the direction to servo the offset to zero, thus accomplishing the acquisition task. If the frequency offset is greater than the acquisition bandwidth, a lock detection signal will not be generated in the time-out interval that monitors the operating state of the receiver. The receiver enters a *failure to acquire* state in which it invokes an acquisition aid. The aid conducts a search for the signal by directing the controlled oscillator to perform a frequency sweep through the expected range of frequency offsets. When the frequency difference between the local oscillator and the errant signal becomes smaller than the acquisition bandwidth, the loop successfully acquires the signal, generates the lock detection signal, and disables the acquisition aid.

13.4.2 Modern Carrier Recovery

A modern receiver will use a maximum likelihood frequency estimator as an aid to pull the local oscillator close to the center frequency of an offset signal and thus bring the offset signal within the acquisition range of the PLL. The derivation of the maximum likelihood fre-

quency estimator is beyond the scope of this text, so we will simply describe its operation and then its implementation. In a manner similar to the maximum likelihood timing-recovery process, we form a filter that is the derivative of the matched filter. In the timing-recovery process we took the time derivative of the matched filter, and then drove the product of the matched filter and the time-derivative matched filter outputs to zero. In the frequency-recovery process we take the frequency derivative of the matched filter and in a similar manner we can drive the product of the matched filter and the frequency-derivative matched filter outputs to zero. The frequency derivative matched filter is called a band-edge filter, and we now describe why it is so called and describe a common variant of its operation.

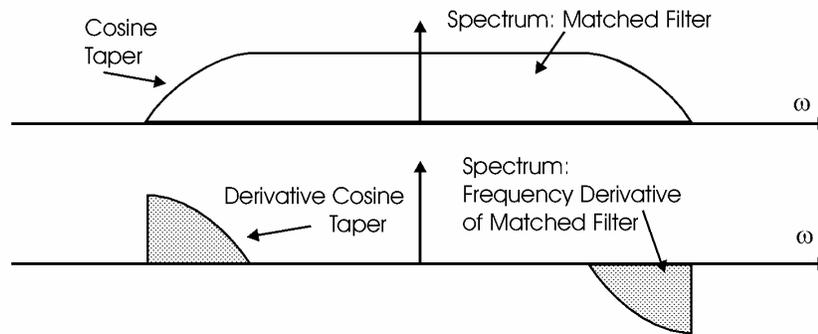


Figure 13.24 Spectra of Matched Filter for Square-root-Nyquist Filter and Frequency-Derivative Matched Filter

Figure 13.24 presents the power spectrum of the matched filter for a square-root-Nyquist filter as well as the frequency derivative of the same filter. Note that the derivative is zero everywhere except in the transition band of the filter. Seeing that the non-zero spectral response of the derivative-matched filter spans the band edges of the matched filter, we can readily understand why it is called a band-edge filter.

Figure 13.25 illustrates how the band-edge filter responds to input signals with significant frequency offsets. The top two figures present the spectrum of an input signal without a frequency offset and the response of the two spectral segments in the band-edge filter. We note that the two band-edge segments contain the same spectral energy. By comparison, the bottom two figures present the spectrum of an input signal with a small frequency offset and the response of the two spectral segments in the band-edge filter. Here we note that, as a result of the spectral shift to the right, the energy contained in the filter segment on the right increased while the energy contained in the filter segment on the left decreased. Energy difference in the two band-edge segments is a simple and accessible observable that can be used to drive the input spectrum to be centered at baseband. The modification to the receiver that incorporates the band-edge filter is shown in Figure 13.26. The polyphase filter following the band-edge filter separates the two band-edge spectral regions, and their outputs become the frequency error detector as the difference of their squared magnitudes.

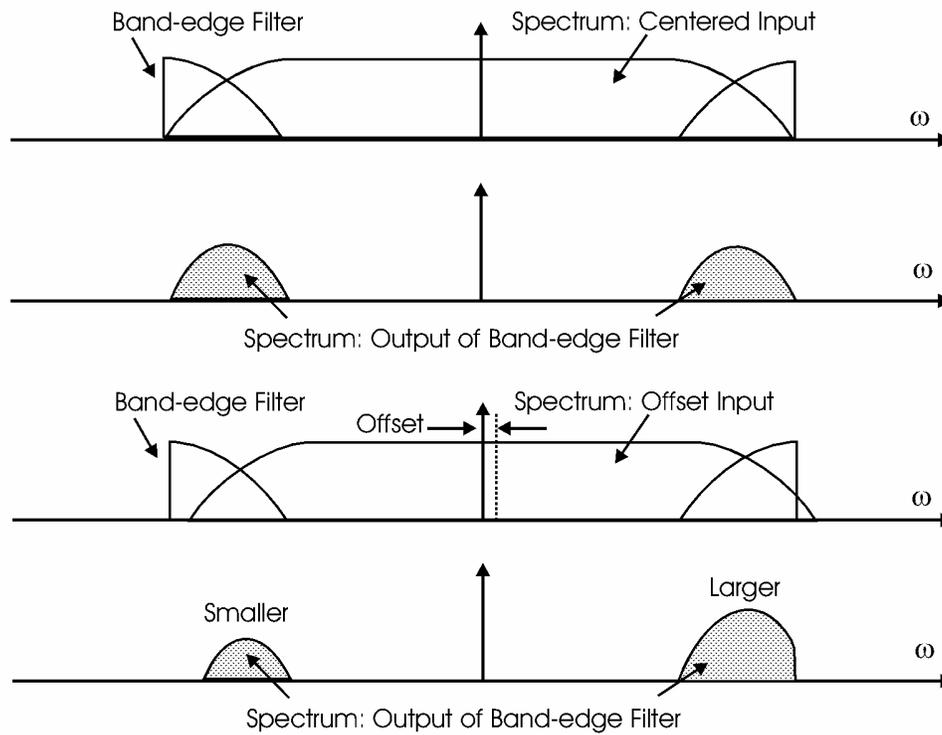


Figure 13.25 Spectra at Input and Output of Band-edge Filter for Centered Input Signal and for Input Signal with Spectral Offset

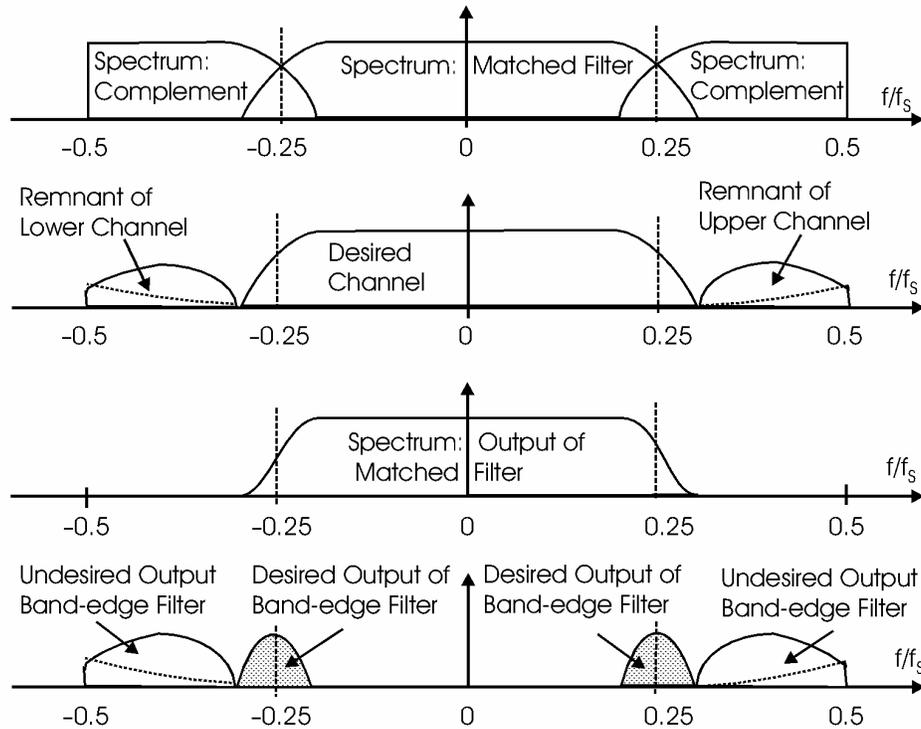


Figure 13.27 Spectrum of Matched Filter and Complement, Spectrum of Composite Signal Presented to Filter Pair, and Spectral Responses from Matched Filter and from Complementary Matched Filter

The polyphase filter following the frequency-matched filter in Figure 13.26 serves three tasks. Its first task is to reject the undesired spectral components adjacent to the band edge spectral regions. This is accomplished by appropriate selection of bandwidth and transition bandwidth of the prototype low pass filter. The next task is that of separating the signal contributions from the positive and negative frequency segments of the spectrum. We can accomplish this by complex translation of the prototype low-pass filter to each of the two spectral locations with trivial heterodynes of the form $\{1, +j, -1, -j\}$. The resultant filters are narrowband Hilbert-transform filters. The third task is an option, and that is to reduce the sample rate because the band-edge filter has a lower bandwidth than the signal bandwidth. Note that the PLL filter that responds to the output of this band-edge signal has a significantly lower bandwidth than the signal bandwidth and is likely operating at a reduced sample rate. The sample rate reduction and the spectral translation from the center frequencies at the quarter-sample rate can both be folded into a polyphase resampling filter. The polyphase filter can be implemented with four or more paths depending on the band-edge bandwidth. This bandwidth is the same as the excess bandwidth of the square-root-Nyquist

matched filter and the input sample rate is twice the signal bandwidth. Thus with a normalized excess bandwidth α equal to 0.25, the band-edge filter is oversampled 8-to-1 and can be down sampled by the same ratio.

There may be applications for which the spectral clean up and channel separation is desired without the sample rate change. Such an example is one in which the band-edge filters are used to generate spectral lines at the symbol rate, which is used as an aid in timing acquisition. For these cases, the polyphase partition can still be applied to the filtering task without invoking the noble identity to resample at the filter input. Figure 13.28 presents the filter chain that implements the matched filter and the band-edge filter pair from the matched filter complement, a polyphase filter, and a pruned 4-point FFT.

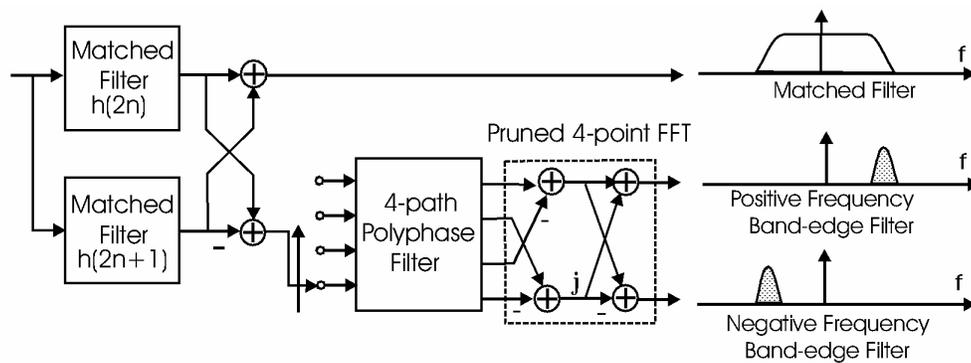


Figure 13.28 Signal Flow for Matched Filter and Upper and Lower Band-edge Filters

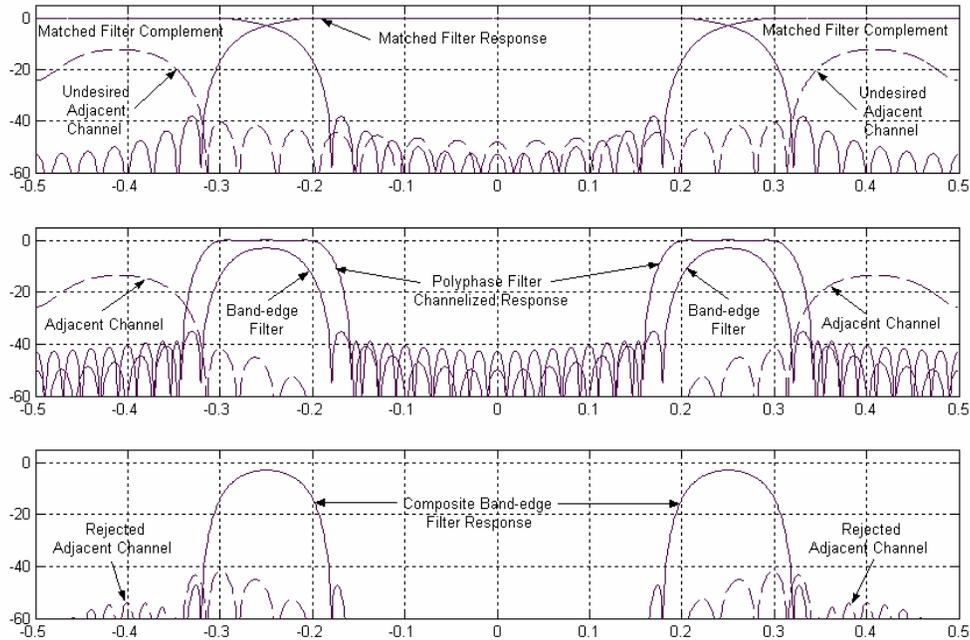


Figure 13.29 Desired and Undesired Spectra at Input to Matched Filter and Complementary Matched Filter, Response of Complementary Matched Filter to Both Inputs, and Response of Polyphase Filter to Both Inputs

Figure 13.29 shows the desired and undesired adjacent channel spectra at input to matched filter and complementary matched filter that forms the initial band-edge filter. We then see the response of the complementary filter to the two input components showing the desired band-edge response and the undesired adjacent channel contribution. Finally we see the spectral response of the 40-tap polyphase filter to the separate components present in the complementary matched filter. As designed, the band-edge filters' responses are isolated and are available and are separated by the polyphase filter.

13.5 DIGITALLY CONTROLLED SAMPLED DATA DELAY

A digitally controlled sampled data delay line described here is implemented with recursive all-pass filter sections. This is in marked contrast to the standard implementation of programmable time delays that use fixed nonrecursive polyphase stages or adjustable Farrow FIR filters. The recursive filter exhibits an equal-ripple approximation to constant group delay. The phase slope is programmable to present a continuously variable time-delay net-

work. Applications of a continuously adjustable, linear-phase time-delay structure offers unique signal processing options to address various communication system tasks. These include timing recovery in DSP-based receivers, adaptive beam forming and steering, communication systems channel modeling, and reverberation modeling in acoustic chambers and instruments.

13.5.1 RECURSIVE ALL-PASS FILTER DELAY LINES

A polyphase M-path filter can be formed with recursive as well as with nonrecursive filter segments. The recursive system is composed of all-pass filters with numerator and denominator formed with reciprocal polynomials in Z^M . For ease of design, and without loss of generality, the polynomials are formed as a cascade of first- and second-order all-pass filter stages. The first-order filter in Z^M forms M-poles and M-zeros with a single multiply while the second-order filter in Z^M forms 2M-poles and 2M-zeros with two multiplies. These filters offer particularly efficient implementations of all-pass transfer functions. Figure 13.30 presents the standard structure for an M-path filter, which because of the Z^M polynomial structure can be used as an M-to-1 down sampler or as a 1-to-M up sampler. When the zero-indexed path in the M-path filter is selected to be pure delay, a particularly simple all-pass filter, every path in the M-path filter becomes an equal-ripple approximation to that path delay. Figure 13.31 presents the structure of a sixth-order recursive all-pass filter that makes up each path in the M-path polyphase filter. Each path contains two first-order polynomials and two second-order polynomials. Each path requires six multiplies per output data point.

Figure 13.32 presents the phase shift of the 10 paths in the 10-path recursive polyphase partition. Note the linear phase as a function of normalized frequency. Figure 13.33 presents the phase slopes for the same 10 paths of the 10-path filter formed with six coefficients per path. Note that the 10 paths present almost linear time delay over $\pm 35\%$ of sample rate. If a signal is sampled at twice its nominal bandwidth, a common practice in many digital receivers, it will experience linear time delay over its whole bandwidth when processed by the 10 stages shown here. For comparison, a 10-stage polyphase FIR filter with the same linear phase shift over the same fractional bandwidth would require 12 taps per path. Figures 13.34 and 13.35 present the phase and phase slopes of the equivalent 10-path FIR filter.

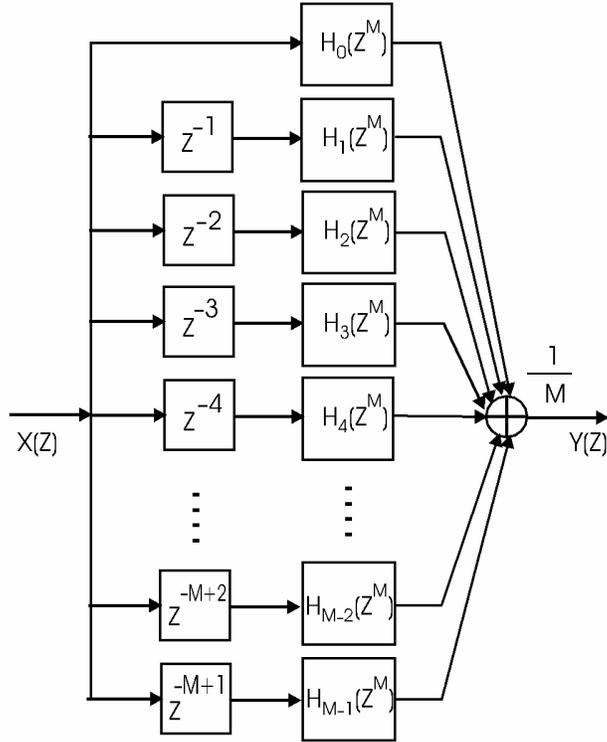


Figure 13.30 M-path Filter Implemented as Recursive All-pass Filter in Z^M

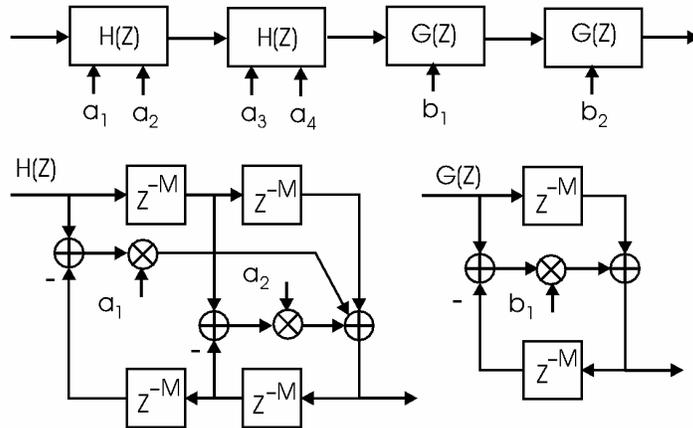


Figure 13.31 Structure of Each Path in M-Path Recursive All-pass Polyphase Filter

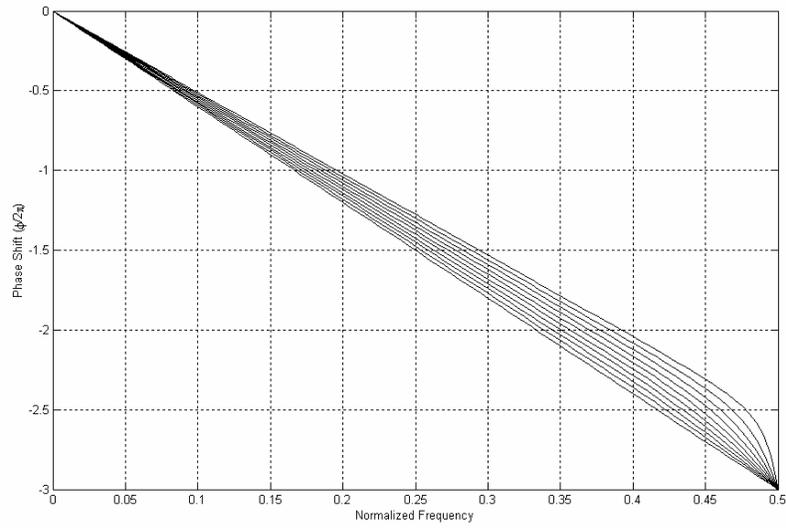


Figure 13.32 Phase Shift for 10 Paths of 10-path Recursive Filter

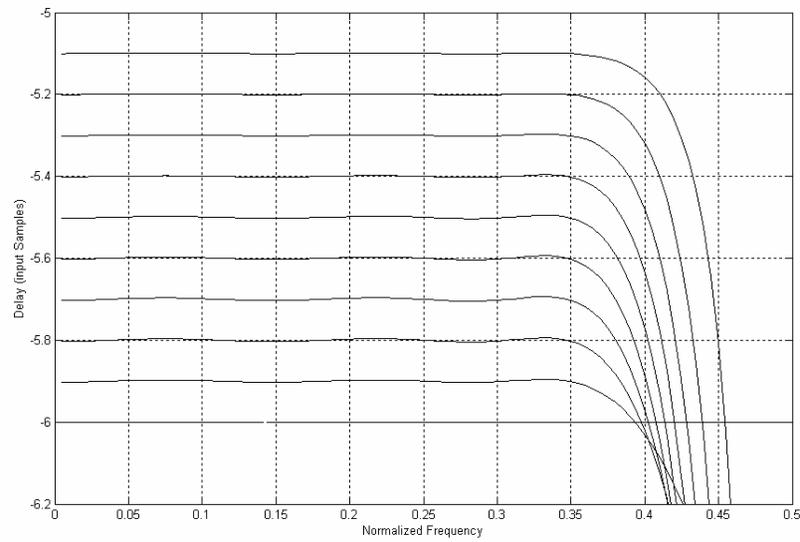


Figure 13.33 Phase Slopes (Group-delay) for 10 Paths of 10-path Recursive Filter

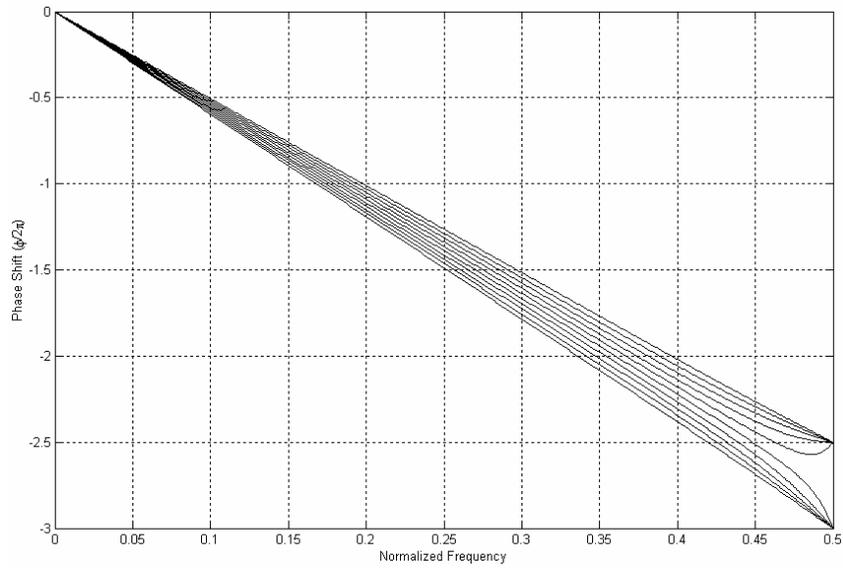


Figure 13.34 Phase Shift for 10 Paths of 10-path Nonrecursive Filter

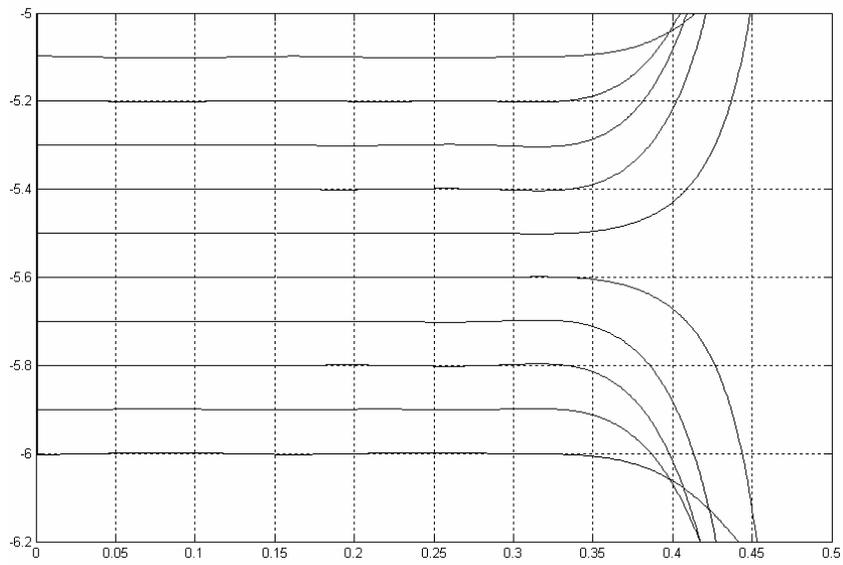


Figure 13.35 Phase Slopes (Group-delay) for 10 Paths of 10-path Nonrecursive Filter

We can imagine that the six coefficients of the all-pass filter on each of the nine non-trivial paths are samples of a smooth continuous curve which, when sampled at increments of 0.1, present the values that define the 9 paths. If we had the smooth continuous functions, we could sample it at any location (besides the multiples of 0.1) to determine the weights required to obtain arbitrary time delay corresponding to the desired sample point. Figure 13.36 presents the coefficients of the 9-path filters and a fifth-order polynomial fitted with equiripple error to the sample values. As seen, the polynomials offer the smooth continuous function that relates weight values to delay. These polynomials can then be sampled at any location to obtain the weights that will present the delay at that sample position. The separate curves are labeled with their parameter values as shown in Figure 13.31.

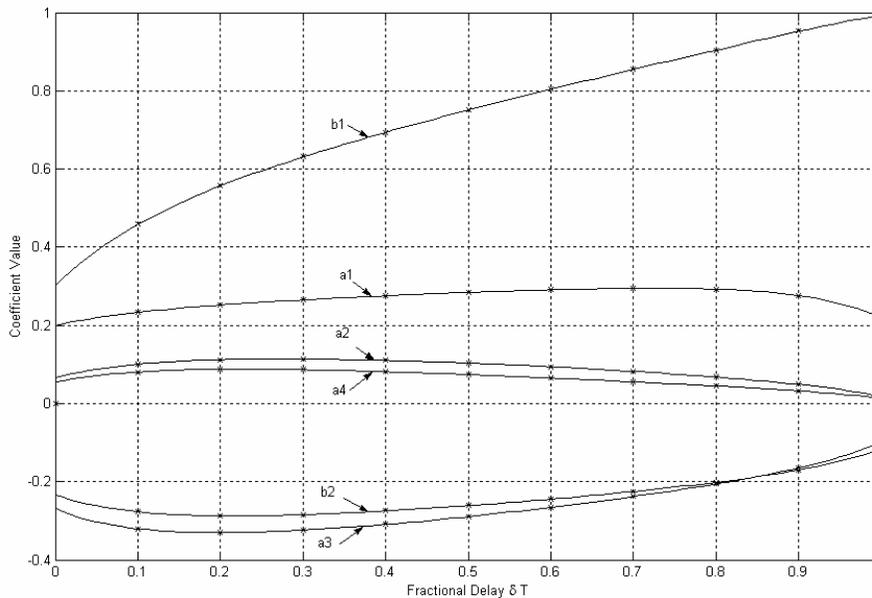


Figure 13.36 Coefficient Values for Non-trivial Paths of the 10-path Filter and Fifth-order Polynomial Fitted to Values

These polynomials are embedded in the structure of an arbitrary time delay network as shown in Figure 13.37. Figure 13.38 presents the original delays plus those obtained by evaluating the polynomials at increments of 0.1 starting from an initial offset of 0.05. These samples correspond to delays midway between those obtained from the original 10-path filter bank. As can be seen, the delays from the coefficients obtained by sampling the polynomials are also linear with frequency.

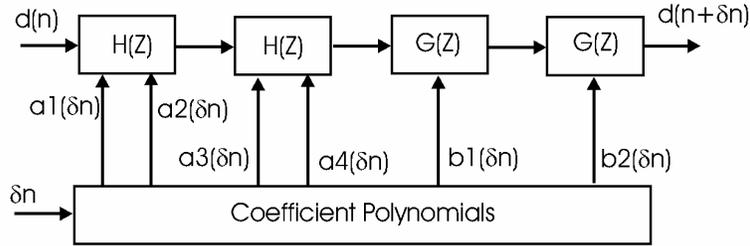


Figure 13.37 Programmable Time-delay Network with Associated Coefficient Generator

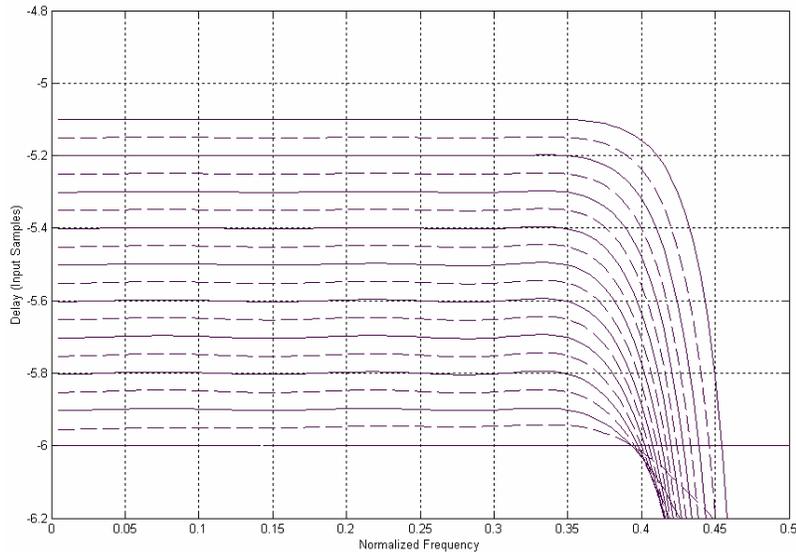


Figure 13.38 Original Delays Plus Midvalue Delays Obtained by Evaluating Coefficient Polynomials Defined by Original 10-path Filter

The digital delay line described here has been used in a timing recovery loop in the receiver structure shown in Figure 13.39. Here the input data is collected at 2-samples per symbol, the envelope is time shifted by the variable digital time delay line, and it is presented to the decision feedback equalizer and detector. Figure 13.40 illustrates various time responses of the receiver structure of Figure 13.39 that uses the variable time-delay network in its timing recovery loop. The variable, labeled timing coefficient, shown in the upper-right corner of Figure 13.40, is the parameter delivered to the coefficient generator shown in Figure 13.37. The timing error is reduced to the minimum level resolvable by the timing error detector. The instantaneous and average time-delay error is shown in the lower-right corner of Figure 13.40.

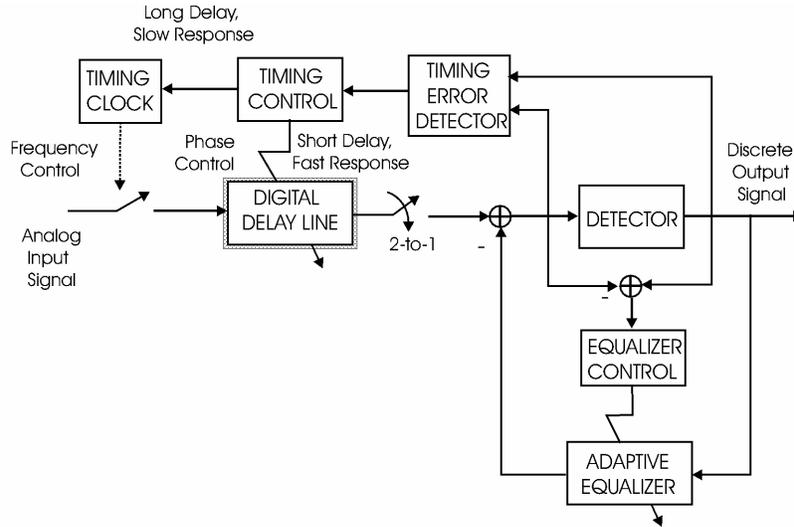


Figure 13.39 Digital Delay Line Embedded in Digital Receiver

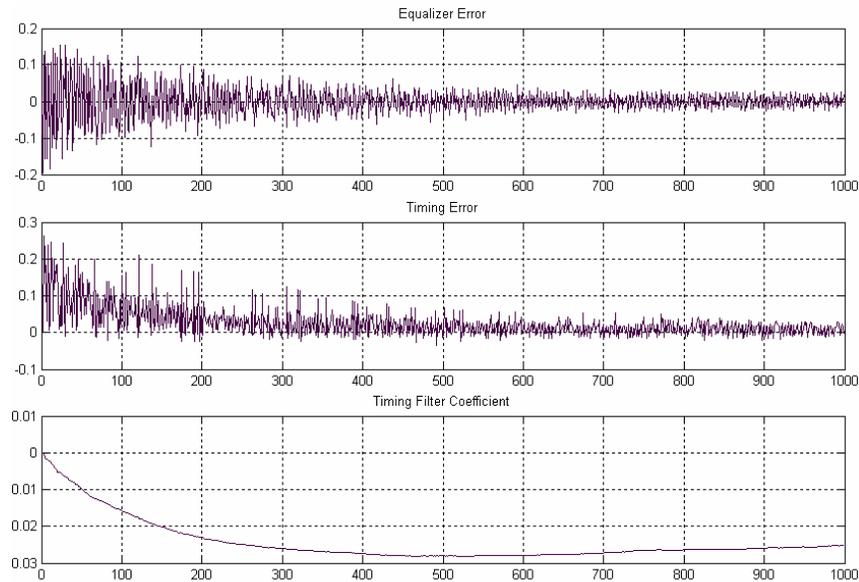


Figure 13.40 Equalizer and Timing Error Response to Timing Coefficient Trajectory of Digital Delay Line Embedded in Digital Receiver

We have demonstrated here that a single path of a standard M-path recursive polyphase filter can be used as a programmable linear-phase time-delay network in a manner similar to the FIR Farrow filter. The example we used to illustrate the performance of this concept used two second-order polynomials in Z^2 and two first-order polynomials in Z^2 , for a total of 6-multiplies per filter path, to obtain the variable time-delay process. Time-delay filters can be formed with fewer stages per arm, and we have tested the performance of systems with 1, 2, or 3 multiplies. These filters, with smaller number of computations, perform very well. They simply exhibit a larger ripple in the time-delay versus frequency characteristic, or exhibit linear group delay over a smaller fraction of the input sample rate due to larger transition bandwidth. Interestingly, when coupled with a decision feedback equalizer, the performance of the time-delay network with a small number of coefficients is enhanced since the equalizer attributes residual group-delay distortion to the channel and compensates appropriately. For those interested in playing with the structure we describe here, Table 13-3 lists the six coefficients of the nine nontrivial paths of a linear-phase 10-path recursive polyphase filter. The MATLAB script file *time_10* that contains the same coefficients and generated the phase portraits is also available in the disc accompanying this text.

Table 13-3 Coefficients of All-pass Filters of Paths 1–9 of 10-path Polyphase Filter

Path #	B1	B2	A1	A2	A3	A4
1	0.458758	-0.276820	0.233603	0.100849	-0.320550	0.080736
2	0.556790	-0.287255	0.252152	0.112364	-0.330782	0.087489
3	0.630817	-0.284168	0.265170	0.113768	-0.324013	0.086214
4	0.693992	-0.274731	0.275704	0.110127	-0.309201	0.081185
5	0.751025	-0.261264	0.284359	0.103237	-0.289275	0.073988
6	0.804237	-0.244646	0.290925	0.093857	-0.265387	0.065386
7	0.854981	-0.224961	0.294424	0.082187	-0.237745	0.055720
8	0.904135	-0.201338	0.292323	0.067839	-0.205482	0.044960
9	0.952320	-0.170104	0.276523	0.048997	-0.164700	0.032264

13.6 INTERPOLATED SHAPING FILTER

Throughout this book we have examined a number of techniques to implement resampling filters. In this section we select a specific task of up sampling by a factor of 8 the impulse response of shaping filter used in a modulator. The particular shaping filter we examine is a square-root cosine-tapered Nyquist filter used in a third-generation cellular phone system. A friend supplied the spectral mask the system had to satisfy and asked us to design the filter and the interpolator to perform the up-sampling task. The question then arose as to what would be an efficient up sampler and what signal degradation effects would be introduced by the up-sampling options. The distortion terms are related to the pre and post echoes that

the interpolating filter contributes to the time response due to ripple in spectral amplitude and spectral phase. We also chose to examine distortion levels related to nonuniform phase characteristics of efficient recursive filters. The measure of distortion that is easy to access is the peak and rms level of ISI at the output of the matched filter. The types of interpolators we examined are listed in Table 13-4, along with an indication of their relative computational complexity and their ISI levels. Also included in this table is the level of ISI contributed by the original shaping filter operating at its design frequency of 2-samples per symbol and then down sampling to symbol rate. The remaining table entries were obtained by operating the filters at 16-samples per symbol and then down sampling the filter output to the symbol rate.

Table 13-4 Interpolator Options and their Comparative Measures

Interpolator Type	Length or Number of Coefficients	Operations per Output Sample	Peak and RMS ISI
Shaping Filter No Interpolation	41-taps	41-M, 41-A	PK; 0.0101 RMS; 0.0036
8-path Polyphase FIR	72-taps, 9-taps/Path	9 M, 9A	PK; 0.0158 RMS; 0.0040
3-stage Half band FIR Filter Set	1st Stage; 21-taps, 2nd Stage; 13-taps 3rd Stage; 9-taps	4.8 M, 4.8 A	PK; 0.0109 RMS; 0.0036
8-path Linear-phase Poly- phase IIR Filter	21-coef. 3-coef./Path	2.7 M, 5.3 A	PK; 0.0128 RMS; 0.0040
3-stage Linear-phase IIR Half-band Filter Set	1st Stage; 4-coef. 2nd Stage; 2-coef. 3rd Stage; 1-coef.	1.5 M, 3.0 A	PK; 0.011 RMS; 0.0036
8-path Nonlinear Phase Polyphase IIR Filter	14-coef. 2-coef./Path	1.8 M, 3.6 A	PK; 0.0817 RMS; 0.0259
3-stage Nonlinear Phase IIR Half-band Filter Set	1st Stage; 3-coef. 2nd Stage; 2-coef. 3rd Stage; 1-coef.	1.4 M, 2.8 A	PK; 0.0808 RMS; 0.0284

Figure 13.41 shows the impulse response and the frequency response of the shaping filter designed to meet the spectral mask indicated on the spectral plot. Also presented in the spectral plot is an insert figure zoomed to show the in-band ripple levels of the shaping filter. The in-band ripple level was specified to be less than 0.1-dB, and here it is seen to be 0.022-dB. We see here that the frequency of the ripple is 10 cycles per interval of symbol bandwidth that has been normalized to 1.0 in this figure. The ripple frequency tells us that the matched filter output will exhibit pre and post echoes at positions ± 10 symbols from the location of its peak value.

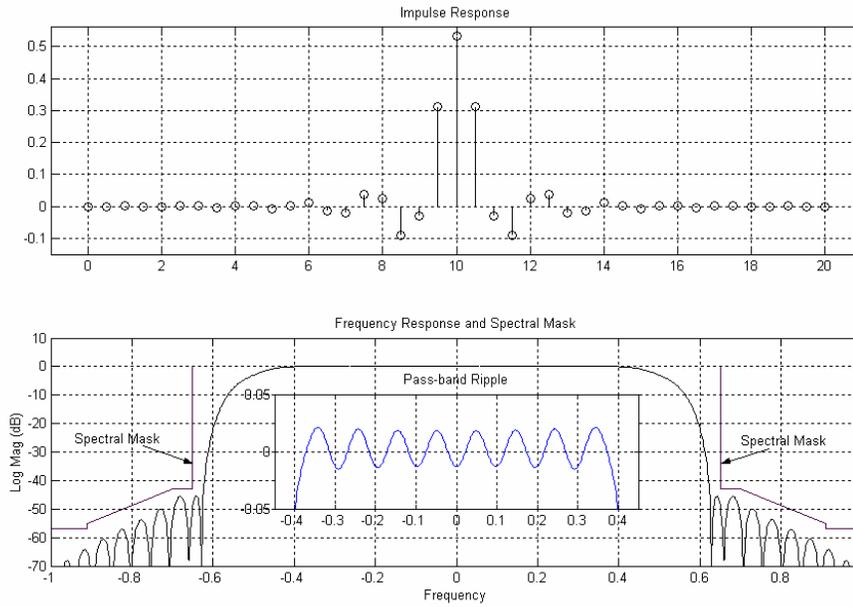


Figure 13.41 Impulse Response and Spectra of Shaping Filter

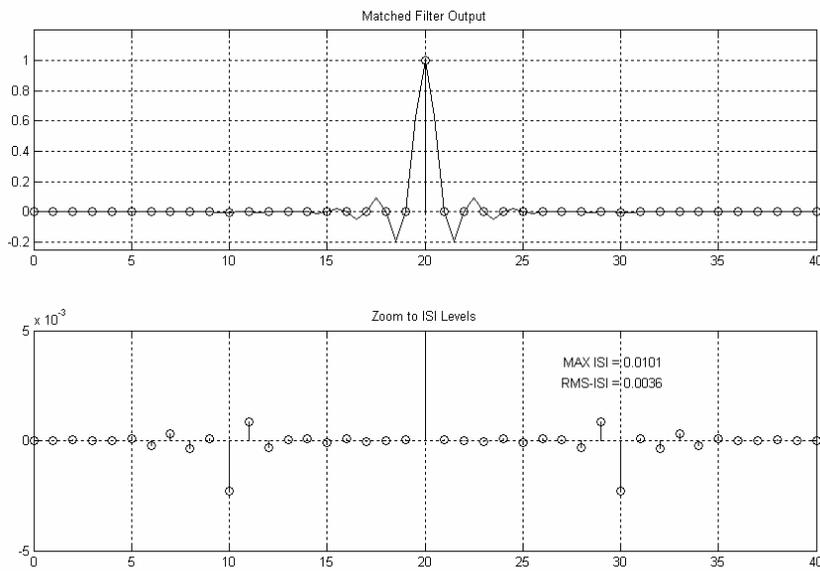


Figure 13.42 Matched Filter Response and Detail Showing ISI levels

Figure 13.42 presents output samples of the matched filter taken at two samples per symbol with circles marking the one-sample-per-symbol time marks aligned with the peak output. The subplot accompanying the matched filter is a zoom to the low ISI levels where we see, as expected, the pre and post echoes at ± 10 symbols from the peak position. The maximum ISI level that the matched filter can exhibit, computed as the sum of the absolute values, was found to be 0.0101, or approximately 1% of the peak output level. The RMS value of the ISI cloud around any constellation point was found to be 0.0036, approximately 1/3 of the peak ISI.

The first contender that comes to mind for use as the 1-to-8 up sampler is an 8-path polyphase FIR filter. A prototype FIR filter of length 72 taps satisfied the filtering requirements dictated by the transition bandwidth and out-of-band attenuation levels. This filter is partitioned into an 8-path polyphase structure that is fed by the shaping filter. The polyphase filter is likely implemented in the form shown in Figure 7.13 rather than in the expanded form suggested by Figure 13.43. We see here that the polyphase form of the filter requires 9 multiplies and adds per interpolated output sample point. We use this workload as the reference against which we compare the other interpolator options.

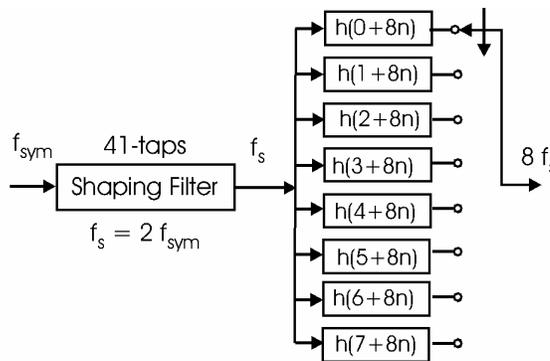


Figure 13.43 Block Diagram of Signal Flow for Shaping Filter and 1-to-8 Polyphase FIR Filter Interpolator

Figure 13.44 shows the phase response of the 8-paths of the polyphase partition of this prototype filter along with its log magnitude frequency response. The phase response imparts little value here except that it does show the constant phase offsets present in the different Nyquist zones that can be used in a polyphase channelizers. The reason we show the phase here is that we will shortly compare this phase profile with corresponding profiles of recursive counterparts of the 8-path filter. Figure 13.45 shows the spectrum of the interpolating filter overlaid on the periodic extension of the shaping filter's spectrum and the spectrum of the composite impulse response of the shaping filter and the 1-to-8 interpolating filter. Also shown in Figure 13.43 is a zoom to the interpolator's in-band ripple. This ripple

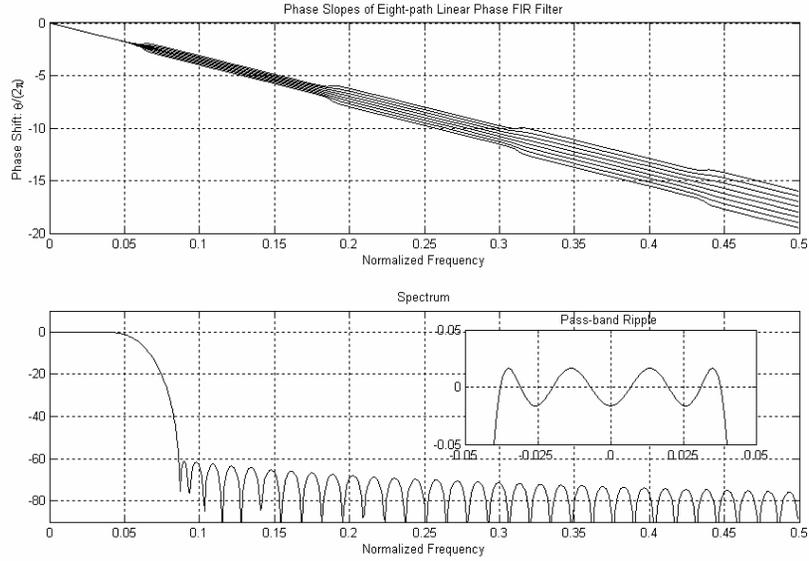


Figure 13.44 Path Phase Responses of 8-Paths Polyphase Filter and Frequency Response of Prototype Filter

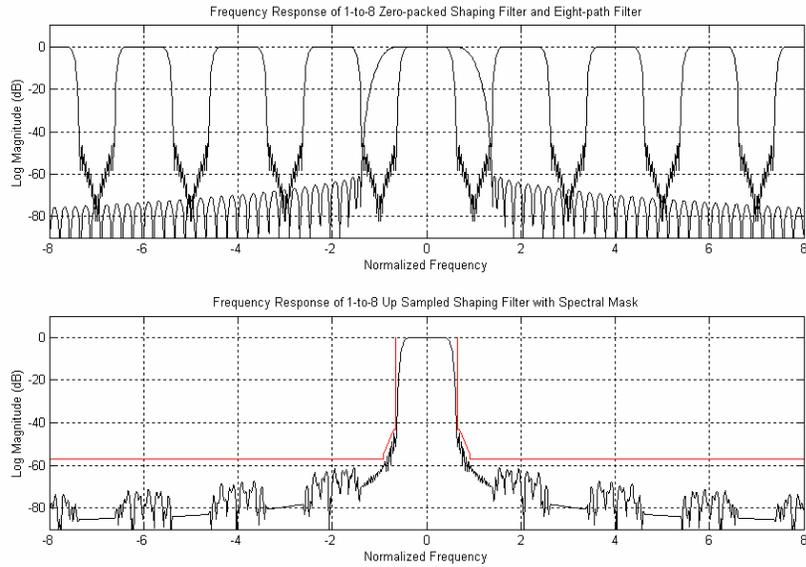


Figure 13.45 Spectra of Up-sampled Shaping Filter with Spectra of Interpolating Filter and Spectra of Interpolated Shaping Filter Response

is seen to be approximately 4-cycles per signal bandwidth and is approximately the same amplitude of the shaping filter ripple. We would expect an increase in ISI level due to the composite filtering action of the two filters. In fact, the ISI exhibited by the time series formed by the 8-path interpolator increased from peak levels of 1% to 1.6% and from RMS levels of 0.36% to 0.4 %.

The next contender for use as the 1-to-8 up sampler is a cascade of 3-stages of half-band FIR filters . The half-band filters offer two primary advantages over a single polyphase filter. The first is that by ratcheting the sample rate up in increments of 2 the successive filters in the cascade, while operating at higher sample rates, do so with shorter filters due to the increased separation between the spectral replicates at the higher sample rates. The second advantage is the nearly 50% reduction in workload due to the zero-valued weights in the half-band filter. A set of prototype FIR filters of lengths 21, 13, and 9 taps respectively satisfied the filtering requirements dictated by the successive transition bandwidths. These filters are partitioned into 2-path polyphase structure in which the upper path of each defaults to a simple delay line. The filter cascade is shown in Figure 13.46, where we see the delays in the upper arm and the nontrivial weights in the lower arm. The lower arm is tagged with its workload per output sample. Note that the first stage is used once per input to output 2 samples with a stage workload of 10 operations. These samples in turn are presented to the next stage, which must operate once per input it sees and thus operates twice to output 4 samples for a stage workload of 12 operations. Finally, the last stage sees 4-inputs and thus operates 4-times to output 8-data samples with a stage workload of 16 operations. The total workload for the three stages is $[10 + 12 + 16]$ or 38 operations, which when amortized over the 8 output samples leads to 4.75 operations per output. This workload is approximately half that of the 8-stage polyphase filter. We note that the half-band filters are too short to exhibit ripple in the signal bandwidth they are interpolating. Thus a second benefit of the half-band filter is that it contributes an insignificant increase to the composite ISI, with the peak ISI changing from 1.0% to 1.1% and the RMS ISI not changing within the measurement resolution of 0.01%. Figures 13.47, 13.48, and 13.49 present the spectra of the successive 1-to-2 up-sampled input data with the overlaid half-band filter responses as well as the spectra of the up-sampled and filtered time responses. These spectra are presented to illustrate the ratcheting process that enables the higher speed processing tasks to be implemented with simpler, shorter filters.

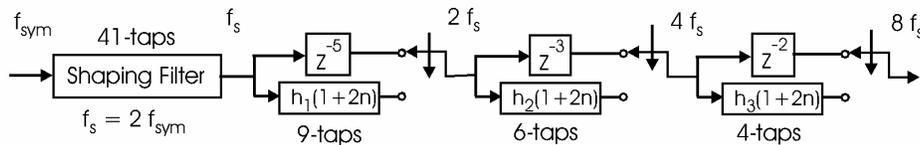


Figure 13.46 Block Diagram of Signal Flow for Shaping Filter and Cascade of Three Levels of Half-band 1-to-2 Up-sampling FIR Interpolating Filter

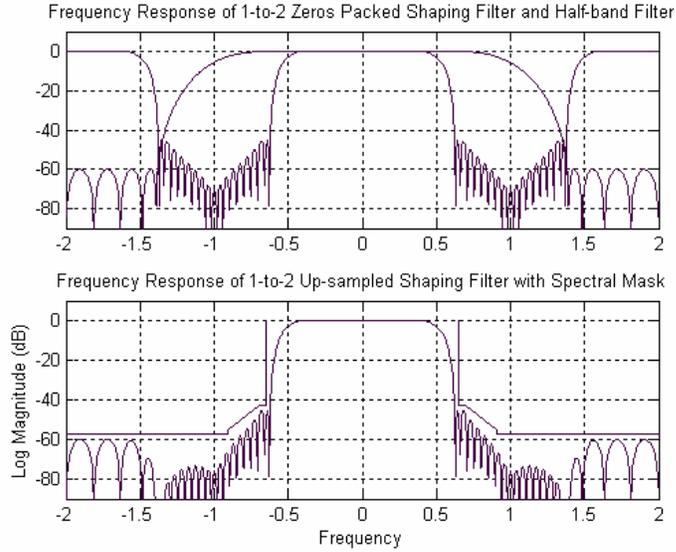


Figure 13.47 Spectra of Up-sampled Shaping Filter with Spectra of First Half-band FIR Interpolating Filter and Spectra of 1-to-2 Interpolated Filter Response

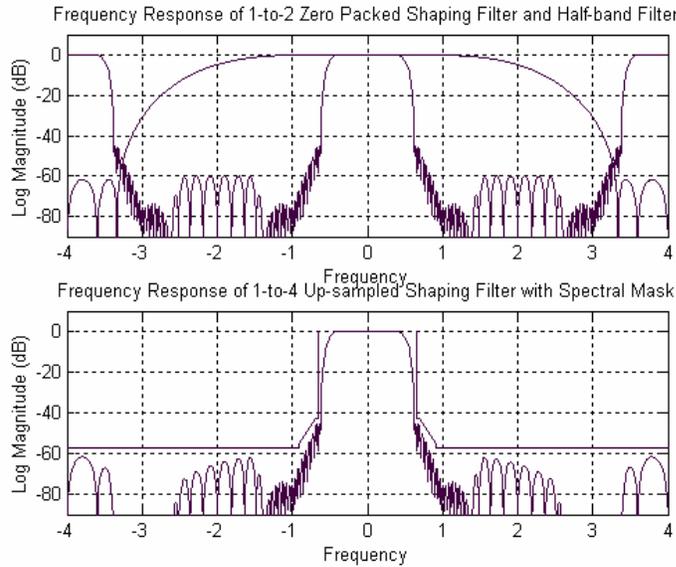


Figure 13.48 Spectra of 1-to-4 Up-sampled Shaping Filter with Spectra of Second Half-band FIR Interpolating Filter and Spectra of 1-to-4 Interpolated Filter Response

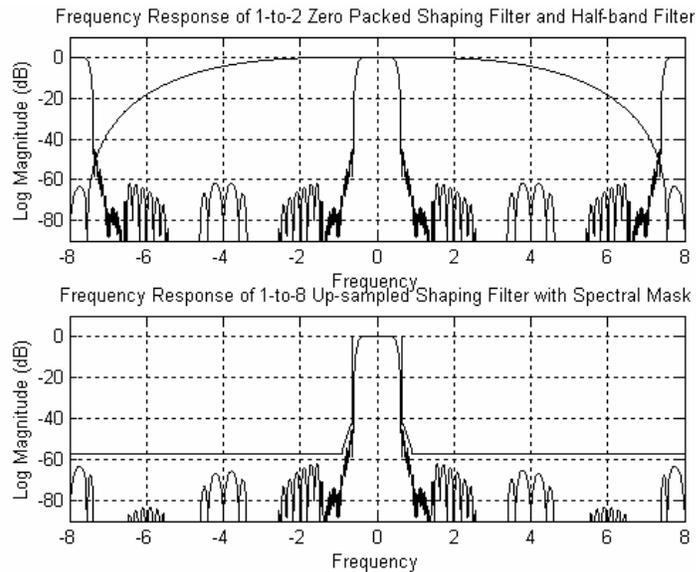


Figure 13.49 Spectra of 1-to-8 Up-sampled Shaping Filter with Spectra of Third Half-band FIR Interpolating Filter and Spectra of 1-to-8 Interpolated Filter Response

At bandwidths, hence sample rates, for which we can use recursive filters for the interpolating function we can bring new contenders to the table. These contenders are the resampling IIR filters described in detail in Chapter 10. There are recursive counterparts to the two nonrecursive options we have just examined. These are an 8-path polyphase filter and a three-stage cascade of half-band filters. Figure 13.50 presents the approximately linear-phase IIR polyphase filter structure. This filter approximates the linear-phase characteristics by setting its top arm to a delay line with linear phase to which the remaining arms approximate with equal-ripple error. The nontrivial arms each have two filters in cascade, one filter requiring a single multiply and two adds and the second filter requiring two multiplies and four adds. The structural detail of these filters is found in Figures 10.11 and 10.35. The workload for this polyphase filter is seen to be less than three operations per output sample. Figure 13.51 shows the group-delay response and the in-band ripple response of the prototype 8-path IIR filter. We first note the remarkable low level of amplitude ripple, the peak ripple being approximately $2 \mu\text{-dB}$ presenting an inconsequential level of ISI. The group-delay ripple, on the other hand, has amplitude of 0.05 and a frequency of approximately 2 cycles per signal bandwidth or 32 cycles per sample rate. We would expect this filter to contribute odd symmetric echoes of amplitude $0.05/32$ at a 2-sample offset from the maximum filter response. Figure 13.52 presents the matched-filter response and a zoom to the ISI levels to show the ISI contributors. The echoes are at the location and at the levels we expected.

Figure 13.53 shows the phase response of the 8-paths of the approximately linear-phase IIR polyphase filter along with its log magnitude frequency response. The phase response shows the phase of the delay-only reference path, and the phase response of the other paths as they approximate the reference slope. Note that at the Nyquist boundaries, the phase difference expands to the next multiple of $2\pi/8$ and that this expansion occurs by inserting additional phase shift. Compare this behavior to the related phase expansion of the 8-path FIR filter in Figure 13.44. For the IIR, the spectral intervals corresponding to the phase transitions between Nyquist zones, the phase terms do not destructively cancel, and the magnitude response exhibits a transition bandwidth between stop band intervals in successive Nyquist zones. We can clearly see the transition bandwidths in the associated log magnitude spectrum. Our first reaction to these spectral bumps is horror; possibly “Good grief!” These transition intervals do not bother us, they can be thought of as don’t care bands matching the spectral intervals in which we know there is no input energy by virtue of the input signal being oversampled. Figure 13.54 shows the spectrum of the IIR interpolating filter overlaid on the periodic extension of the shaping filter’s spectrum. As expected the transition regions of the polyphase IIR filter match the stop bands already present in the input spectrum. The spectrum of the up sampled and filtered composite impulse response is seen to meet the spectral ask requirements of the filtering task.

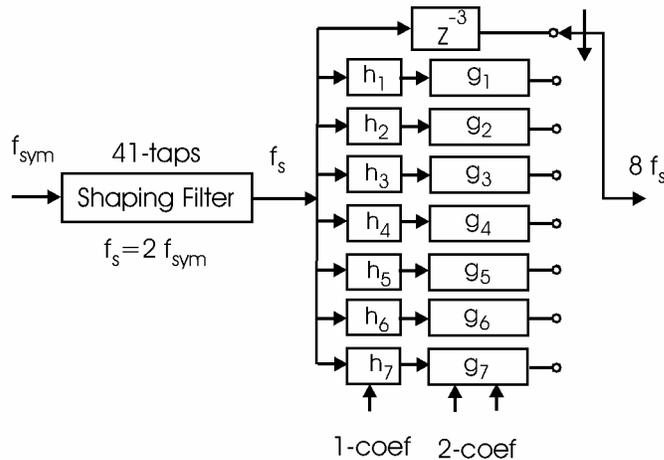


Figure 13.50 Block Diagram of Signal Flow for Shaping Filter and 1-to-8 Polyphase, Approximately Linear, Recursive Filter Interpolator

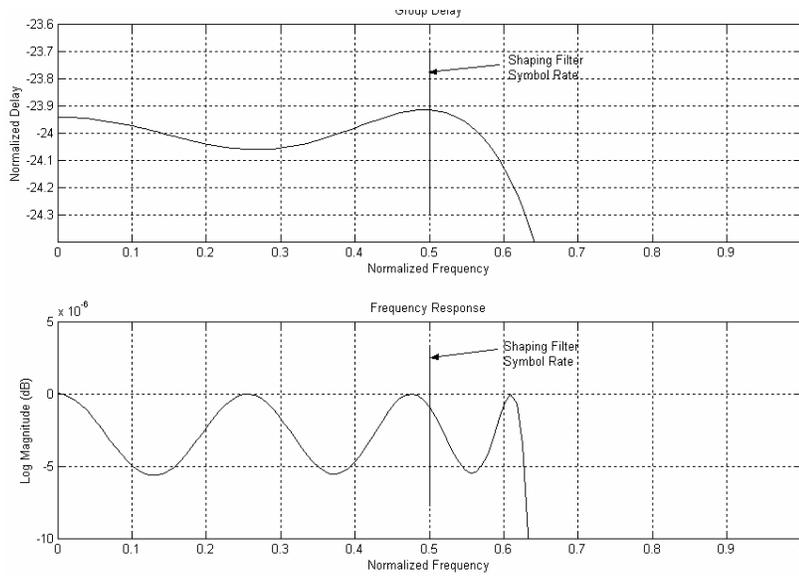


Figure 13.51 Zoom to Group-Delay Ripple and Magnitude Ripple of 1-to-8 Polyphase Recursive, Approximately Linear, Recursive Filter Interpolator

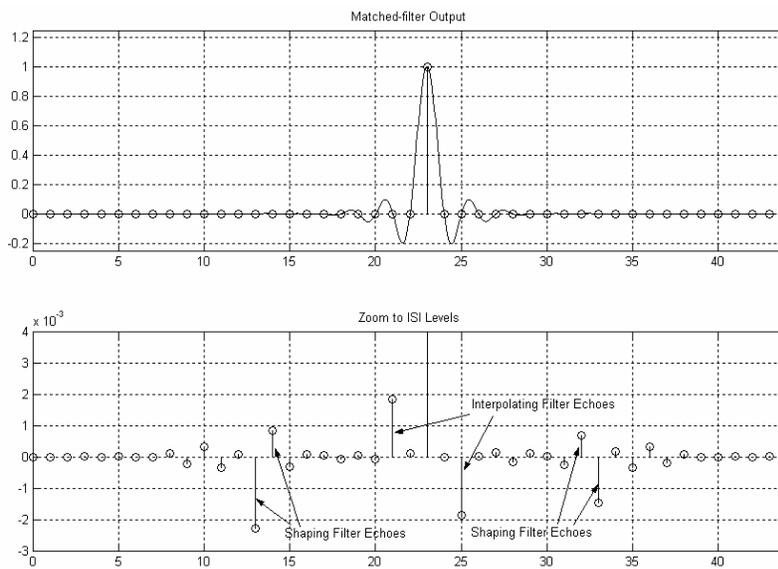


Figure 13.52 Interpolated Matched-filter Response and Detail Showing ISI Levels

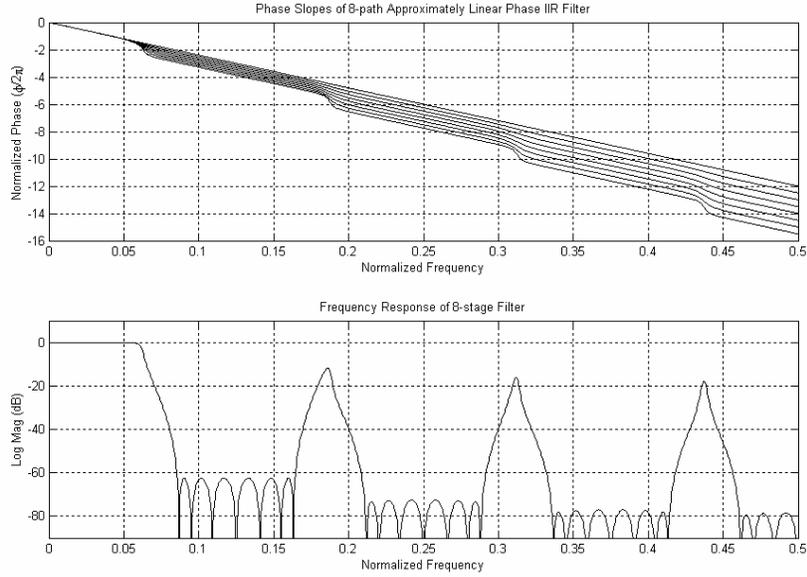


Figure 13.53 Path-phase Responses of 8-path Polyphase IIR Filter and Frequency Response of Prototype Filter

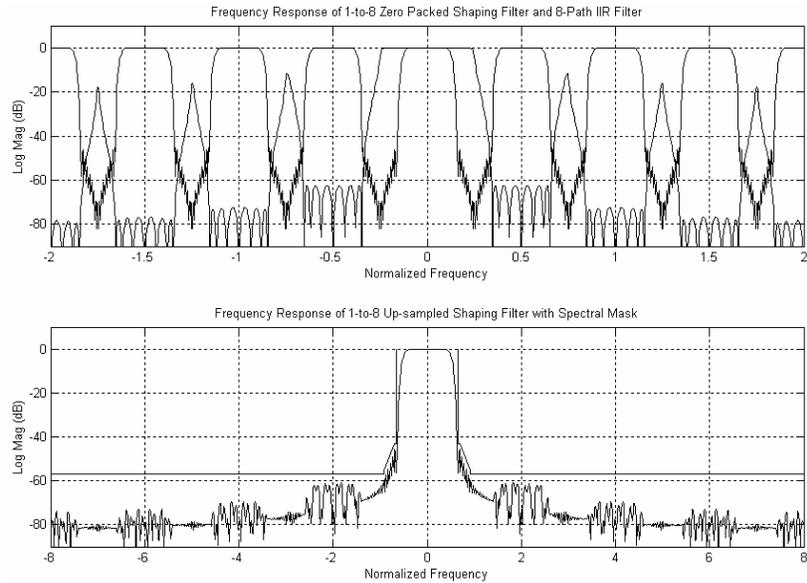


Figure 13.54 Spectra of Up-sampled Shaping Filter with Spectra of IIR Interpolating Filter and Spectra of Interpolated Shaping Filter Response

The final interpolator option we examine in this discussion is the use of the nearly linear-phase recursive half-band filters. These filters have the same property as the FIR half-band filters in that the upper arm of the polyphase half-band is delay only. The IIR version of a 3-stage cascade that meets the filtering requirements of the interpolator task is shown in Figure 13.55. Here we see the delays in the upper arm and the all-pass filter structures in the lower arm. The workload for the filters in the lower arm matches the number of input coefficients that represents a small workload to perform its phase-shift task. In this cascade, the first stage is used once per input to output 2 samples with a stage workload of 4 operations. These samples in turn are presented to the next stage, which must operate once per input it sees and thus operates twice to output 4 samples for a stage workload of 4 operations. Finally, the last stage sees 4-inputs and thus operates 4-times to output 8-data samples with a stage workload of 4 operations. The total workload for the three stages is $[4 + 4 + 4]$ or 12 operations, which when amortized over the 8 output samples leads to 1.5 operations per output. This workload is approximately half that of the 8-stage polyphase IIR filter and one-third of the workload for the FIR half-band cascade. We note that the IIR half-band filters do not exhibit amplitude ripple and that their phase ripple has a long period relative to the bandwidth being processed during the interpolation process. Thus a second benefit of the IIR half-band filter is that it contributes an insignificant increase to the composite ISI, with the peak ISI changing from 1.0% to 1.1% and the RMS ISI not changing within the measurement resolution of 0.01%. Figures 13.56, 13.57, and 13.58 present the spectra of the successive 1-to-2 up-sampled input data with the overlaid half-band IIR filter responses as well as the spectra of the up-sampled and filtered time responses. These spectra are presented to allow comparison between FIR and IIR filters performing the same ratcheting process to permit shorter filters at higher sample rate. Be sure to compare these figures with Figures 13.47, 13.48, and 13.49, keeping in mind the relative workloads expended while performing the interpolation process.

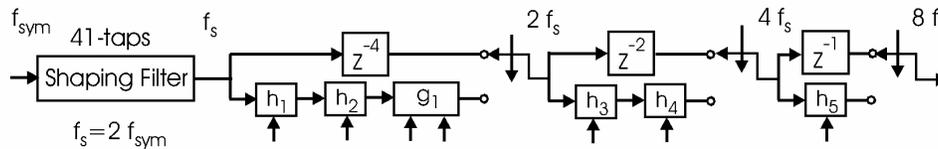


Figure 13.55 Block Diagram of Signal Flow for Shaping Filter and Cascade of Three Levels of Half-band 1-to-2 Up-sampling IIR Interpolating Filter

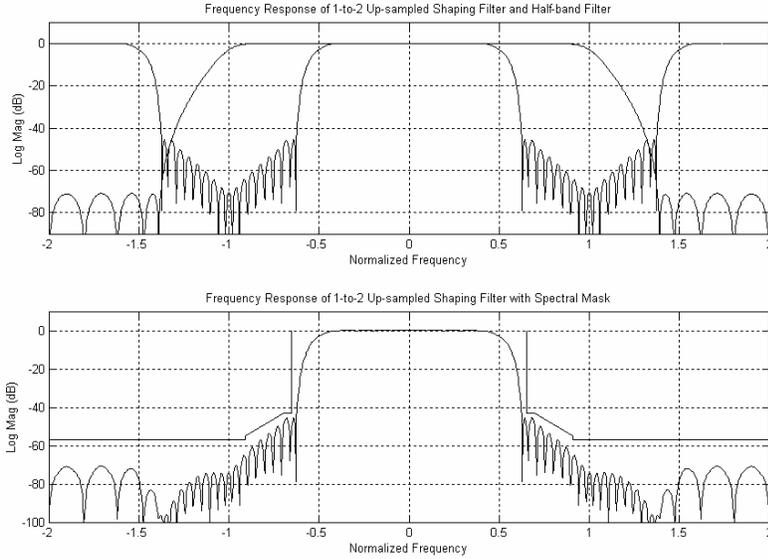


Figure 13.56 Spectra of Up-sampled Shaping Filter with Spectra of First Half-band IIR Interpolating Filter and Spectra of 1-to-2 Interpolated Filter Response

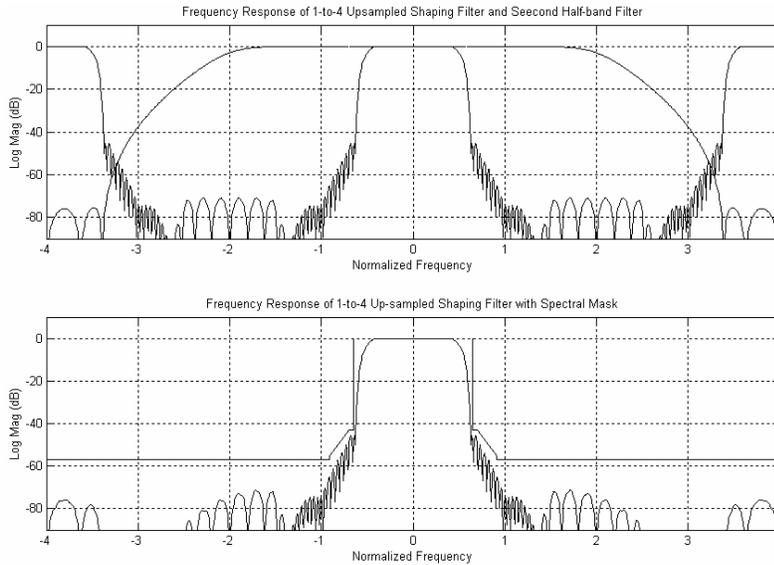


Figure 13.57 Spectra of 1-to-4 Up-sampled Shaping Filter with Spectra of Second Half-band IIR Interpolating Filter and Spectra of 1-to-4 Interpolated Filter Response

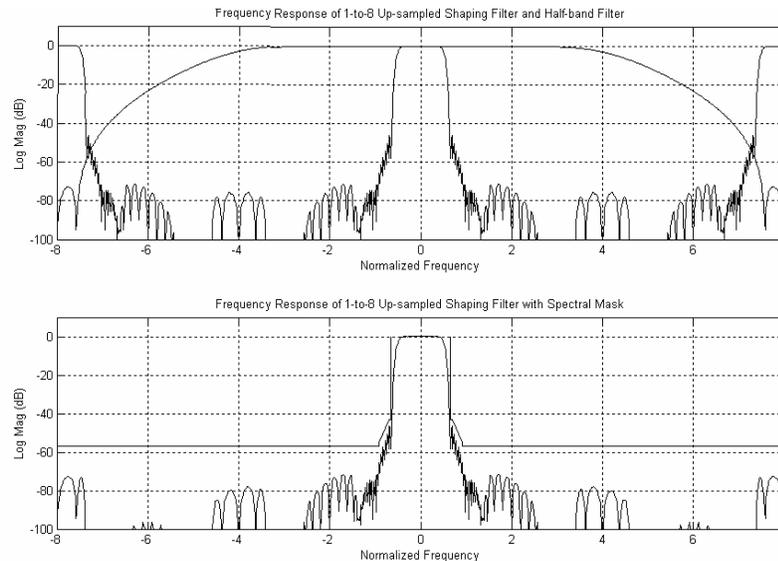


Figure 13.58 Spectra of 1-to-8 Up-sampled Shaping Filter with Spectra of Third Half-band IIR Interpolating Filter and Spectra of 1-to-8 Interpolated Filter Response

A final note in this area is that the 8-path polyphase filter and the 3-stage half-band filter can also be implemented with recursive filters that exhibit nonuniform phase responses. The workload required of these filters is again reduced relative to that of the linear-phase recursive filter set. These filters can be mixed and matched, linear filters at the low sample rate and nonlinear phase filters at the higher sample rate. At the higher rates, the fractional bandwidth is sufficiently small that the nonuniform phase IIR filters are approximately linear-phase filters. Another, tongue in cheek, comment is that group-delay distortion introduced in the shaping filter and interpolator at the modulator will be attributed to channel effects, including analog filters in the signal flow path, and that these small effects will be suppressed by an equalizer in the demodulator. A less onerous comment is that the group-delay distortion terms generated by the low-cost, nonuniform phase recursive interpolating option are known a priori and can be pre-equalized by inserting their conjugate phase in the initial shaping filter.

13.7 SIGMA-DELTA DECIMATING FILTER

Multirate filters find great application in source coding applications. One ubiquitous application is in the filtering and sample-rate change associated with sigma-delta modulators. The sigma-delta modulation process is a source-coding technique that uses memory to represent

samples of data at a given level of fidelity with a smaller number of bits than normally required to achieve that fidelity level when used without memory. The source coding performed by an A-to-D converter is that of representing a sampled data process with a finite set of amplitudes in such a way that the error between the two representations is made acceptably small. Many A-to-D and D-to-A converters perform their conversion process on a sample-to-sample basis without regard to past or future conversion steps. These are often called memoryless converters or, as a friend once observed, converters with amnesia. The memory required for this conversion process resides in the correlation between data samples. If data samples are highly correlated, and if we have the current and recent past samples of the data, there is little uncertainty of the value of the next sample. Predictive encoders operate on this basis. We predict the next sample and use the converter to measure or resolve the error in the prediction process. Similarly, if the data is highly correlated, the error generated by the conversion process is also highly correlated and hence predictable. If we are able to predict the error made by the quantizer, we can subtract the estimate prior to the conversion process. Converters that predict and cancel errors prior to their occurrence are called noise feedback or noise shaping converters. The classic sigma-delta converter is a member of this class of systems.

To assure high levels of correlation between data samples, we significantly oversample the process being encoded. Typical oversample ratios are 16 to 128 times the Nyquist rate of the process. Figure 13.59 presents the block diagram of a simple noise feedback quantizer. In this figure the quantizer is modeled as an additive noise source. The difference between input and output of the quantizer is the quantizer error. We measure this error as the difference between the quantizer input and output ports and present the error to the predicting filter with Z-transform $P(Z)$. This filter, based on previous samples of the quantizer error, predicts a value for the current error and subtracts it from the input signal prior to delivering it to the quantizer. The simplest predicting filter uses the previous error as an estimate of the current error and is in fact merely a one-sample delay register, that is $P(Z) = Z^{-1}$. This substitution for the simple predicting filter is shown in Figure 13.60. In this model we see clearly that the noise is being fed back to the input of the system. The block diagram shown in Figure 13.60 is often redrawn as a two-loop feedback system. The first loop starts at the input goes to the quantizer, but not through it, and returns to the input through the delay line. This loop describes a digital integrator. The second loop starts at the input, goes through the quantizer to the output port, and returns to the input through a delay and negative sign. This alternate description of the noise feedback process is shown in Figure 13.61. Finally, the explicit feedback around the delay line in the first loop forming the digital integrator is suppressed by replacing the loop by its transfer function as shown in Figure 13.62. This is the most common representation of a sigma-delta modulator, one or more integrators and a quantizer in a unity gain feedback loop.

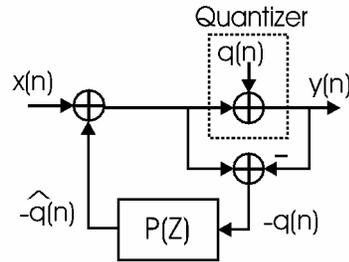


Figure 13.59 Block Diagram of a Noise Feedback Quantizer with Predicting Filter $P(Z)$

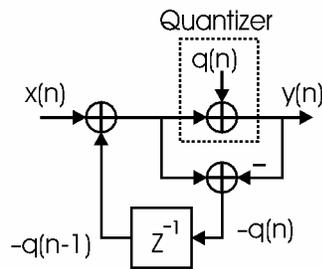


Figure 13.60 Block Diagram of a Noise Feedback Quantizer with Delay Line as Predicting Filter

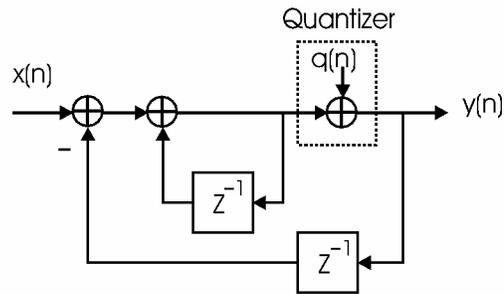


Figure 13.61 Alternate Block Diagram of a Noise Feedback Quantizer Showing Digital Integrator in Feedback Loop

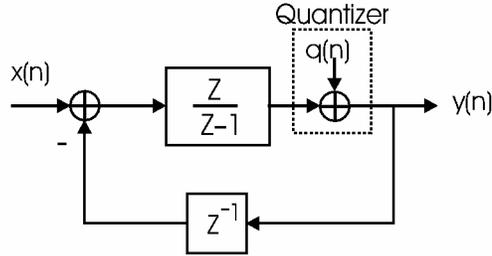


Figure 13.62 Block Diagram of a Noise Feedback Quantizer with Block Diagram of Digital Integrator Replaced by Its Transfer Function

If we model the sigma-delta modulator as a two-input, one-output system we can form the output as the sum of two inputs through their corresponding transfer functions. This is shown in (13.13). When we replace the predicting filter with a delay line Z^{-1} we obtain the relationship shown in (13.14) where we see that the noise transfer function (NTF) is a simple differentiator, $[1 - Z^{-1}]$.

$$Y(Z) = X(Z) + Q(Z)[1 - P(Z)] \quad (13.13)$$

$$Y(Z) = X(Z) + Q(Z)(1 - Z^{-1}) = X(Z) + Q(Z) \left[\frac{Z-1}{Z} \right] \quad (13.14)$$

The single zero of the NTF resides at $Z = 1$, the zero frequency position for sampled data spectra. A stylized power spectrum of the output signal is shown in Figure 13.63. Here we see that the noise has been shaped by the NTF, which has placed a double zero at DC. The double zero suppresses noise in the vicinity of DC but permits noise away from the DC area. Since the input signal has been oversampled by, say, a factor of 64, the spectrum of the input is confined to a small neighborhood around DC, the bottom $\pm 0.8\%$ of the sample rate. Thus the sigma-delta loop has arranged to keep noise away from the low pass spectral region that contains the input signal and place it in a spectral region that contains no input signal. Since the zero of the NTF suppresses the noise, the level of noise injected by the quantization process is not important since the product of any finite noise power spectral density and the zero of the NTF is always zero. Consequently, the quantizer is often implemented as a one-bit decision device. While a number of sigma-delta systems are implemented with as many as four bits performing the quantization process, a one-bit quantizer with multiple integrators in the loop is the most common structure. A low pass filter can reject the out-of-band quantization noise, and the sample rate of the filtered data can be reduced in concert with the bandwidth reduction. This certainly sounds like a task for multi-rate filters.

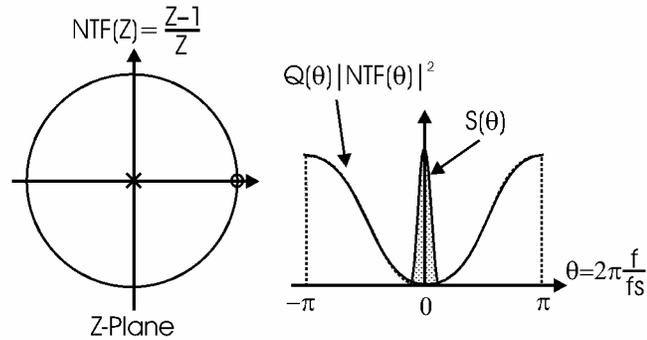


Figure 13.63 Roots of NTF and Power Spectrum of Composite Output Signal of a Noise Feedback Quantizer

13.7.1 Sigma-delta Filter

The filtering task we address here is that of reducing the bandwidth and sample rate of an input data stream formed by a one-bit, two-loop sigma-delta converter operating at 64 times the signal's Nyquist rate. For this example, the signal's two-sided bandwidth is 30 kHz, and we require an output sample rate of 60 kHz. Additional post processing of the oversampled output series is not of interest to us here. The sigma-delta converter input data rate is 3.840 MHz. The dynamic range of the two-loop modulator is 90-dB, so the filtering to be accomplished is equivalent to a single low pass filter with a two-sided bandwidth of 30 kHz, a transition bandwidth of 15 kHz, and a dynamic range of 90-dB with 0.1-dB in-band ripple. The 90-dB is defined by the performance of the 2-loop sigma-delta, which improves quantizing SNR at the rate of 15-dB per doubling of sample rate. The filter following the modulator will of course reduce the sample rate in proportion to the bandwidth reduction. Figure 13.64 is a block diagram of the sigma-delta loop, and Figure 13.65 presents a plot of the input and output time series formed by the modulator as well as the spectrum and a zoomed baseband detail of that spectrum of the output time series.

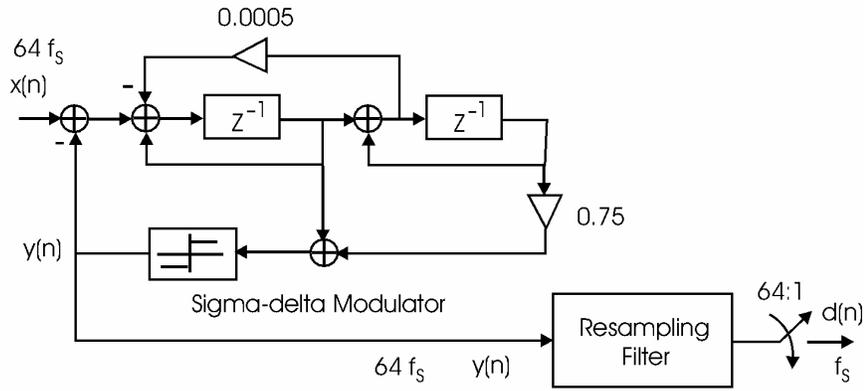


Figure 13.64 Block Diagram of Two-loop, One-bit Sigma-delta Modulator and its Companion Resampling Filter

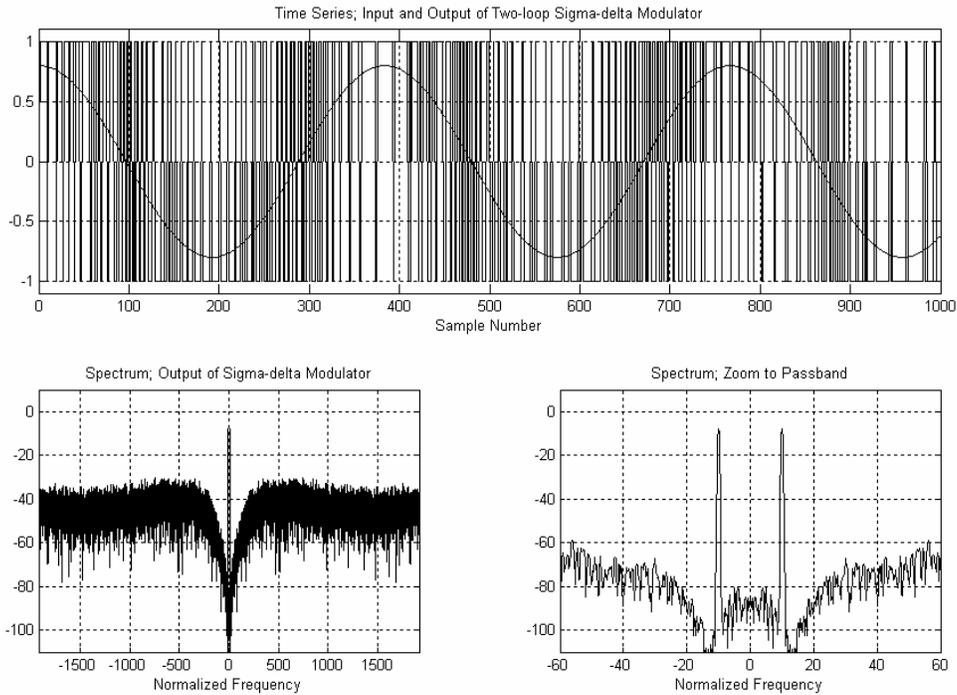


Figure 13.65 Time Series of Input and Output of Sigma-delta Modulator and Spectrum with Zoomed Detail of Output Spectrum from Modulator

The traditional approach to the filtering and resampling task following the sigma-delta modulator is a two-step process comprising a 16-to-1 CIC filter followed by a 4-path poly-phase filter or a pair of half-band filters. This approach is similar to the filtering used in digital down converters. A 4th-order CIC is required to meet the 90-dB suppression requirement. The CIC is operated as a Hogenauer filter, simultaneously performing filtering and 16-to-1 down sampling. Figure 13.66 presents the time series formed at the output of the CIC as well as the spectrum of the output and with an overlaid frequency response of the 4-stage CIC filter. Note that the input spectrum contains multiple equal amplitude, non-harmonically related sinusoids. This signal is used to probe the frequency response of the process. Figure 13.67 presents the same information as did Figure 13.66 except here the time series and the spectra are formed after the time series has been down sampled 16-to-1. Here is where we would observe undesired aliasing into the baseband spectral region. None is seen!

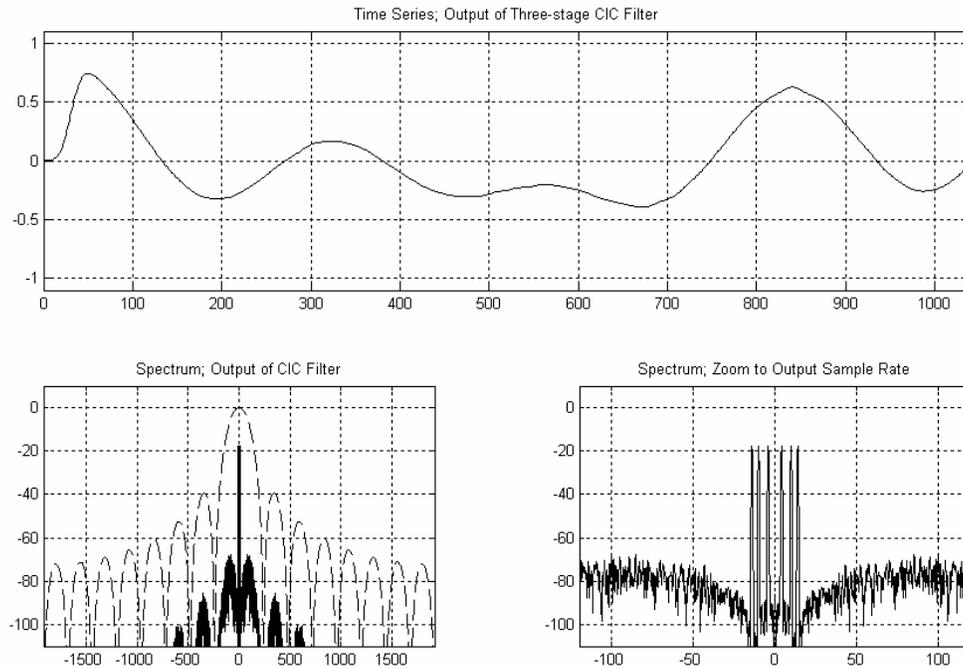


Figure 13.66 Time Series and Spectrum of CIC-filtered Output of Sigma-delta Modulator with Overlaid Filter Response and Zoomed Detail of Output Spectrum

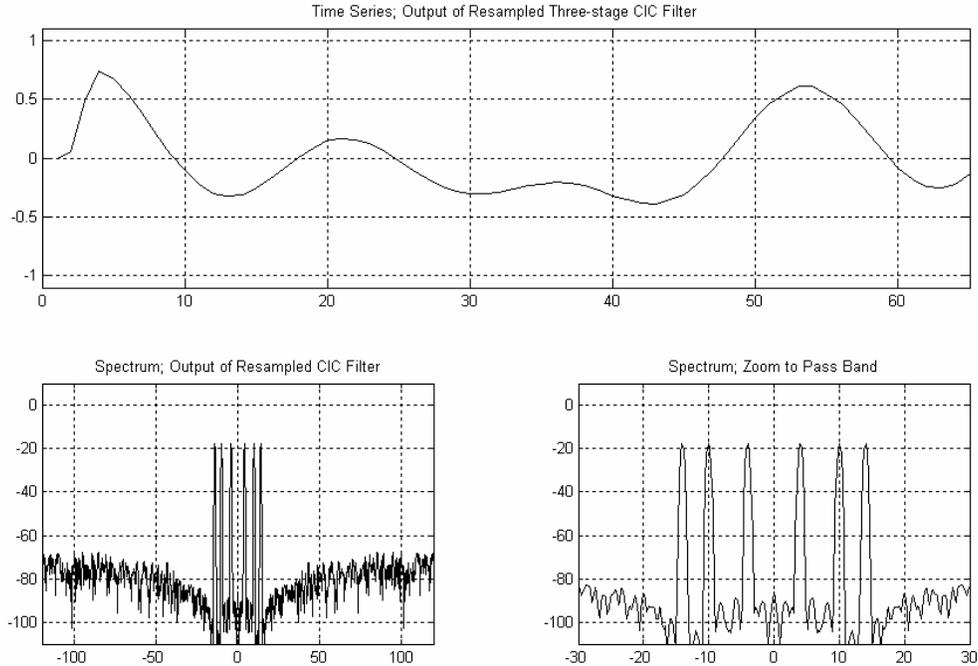


Figure 13.67 Time Series and Spectrum of CIC-filtered and Down-sampled Output of Sigma-delta Modulator and Zoomed Detail of Output Spectrum

The 4-to-1 down-sampling filter following the CIC performs spectral housekeeping, which entails spectral suppression of out-of-band quantizing noise and in-band spectral equalization if required, as well as the sample rate change. The amount of spectral suppression required of this filter is surprisingly small. The required level of additional attenuation can be estimated from the subplot in the lower-left side of Figure 13.67. Here we see that we only require another 20-dB or so to achieve the desired 90-dB attenuation levels. Figure 13.68 presents the time response and spectrum of the output from the final 4-to-1 down sampling filter. Also overlaid on the output spectrum is the frequency response of this 24-tap filter. We also see the spectrum of the final down-sampled time series and note that the spectral terms are 90-dB below full-scale response. What we have illustrated here is that the combination of noise shaping with the 2-loop sigma-delta modulator along with the spectral suppression performed by the cascade of two resampling filters has been able to convert a highly oversampled, hence highly correlated, signal with a 1-bit noise shaping converter to 15-bits of uncorrelated data samples.

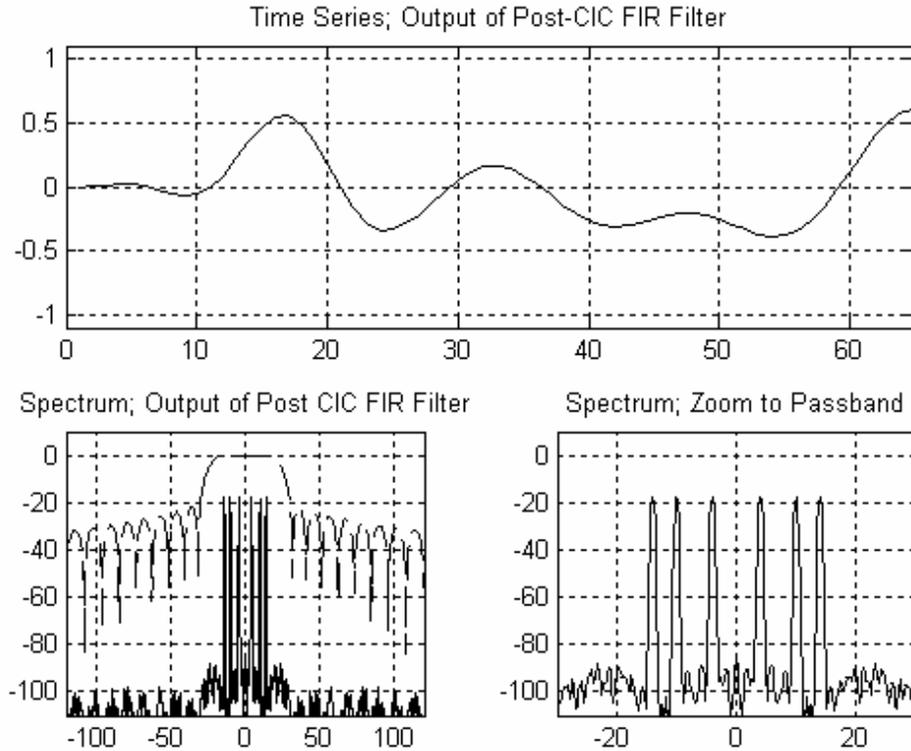


Figure 13.68 Time Series and Spectrum of Second FIR Filtered and Down-sampled Output from CIC Filter with Filter Overlay and Zoomed Detail of Output Spectrum

It may have occurred to you that the filtering performed by the cascade of the CIC and the 4-path polyphase filter did not take advantage of the fact that the data formed by the sigma-delta modulator is bipolar and of fixed amplitude, in fact only the sign of the modulated process. Processing such data in a FIR filter can be accomplished without the need for multiplications. This awareness was not obvious because the CIC filter that processed the modulator output had no multiplies. We depart here from the standard solution and ask about a polyphase FIR filter that can perform the same 16-to-1 bandwidth and sample-rate reduction as the CIC filter. A polyphase FIR filter that accomplishes the same spectral suppression requires four-coefficients per polyphase arm. The data delivered to the polyphase filter from the sigma-delta modulator is a 1-bit sample, the sign-bit, that conceptually is delivered to successive filter paths by the input commutator. Thus the content of each path register is a +1 or -1 in each of the four register positions. The output from any given path is simply the weighted sum of the filter weights where the weighting terms are ± 1 . The polyphase filter requires four-sums for each input sample, but of course the four-stage CIC filter

also requires four-sums for each input sample. That's interesting! The 16-path polyphase filter requires a total of 64 sums to form one output sample at the output rate. The four-stage CIC requires the same number of sums to perform the arithmetic in the four-overflowing accumulators. The output of the final accumulator is then passed to the four-derivatives in the CIC chain to perform four more sums at the output rate. In terms of hardware resources, the four accumulators in the CIC can be mapped to the 4 accumulators in the dual form of the polyphase filter presented in Figures 5.11 and 5.12. This accumulator variation is shown in Figure 13.69. The advantage of using the multiply-free form of the polyphase filter is that we can control the in-band spectral characteristics of the filter, an option not available to the CIC implementation. Another consideration applicable to FPGA implementation is that the content of each polyphase filter path is a four-bit binary word.

There are only 16 possible outputs that any given stage can deliver to its output port. These outputs can be precomputed and stored in a 16-element array unique to each path that is addressable from the four-bit word stored in the input register for that path. The sum of the outputs from the 16 paths is the down -ampled and filtered time series. The 16-path filter then only requires 16 table accesses and 16 sums rather than the 64 adds required by the equivalent CIC. Figure 13.70 shows the look-up table implementation of the 16-path polyphase filter operating with binary inputs from the sigma-delta modulator.

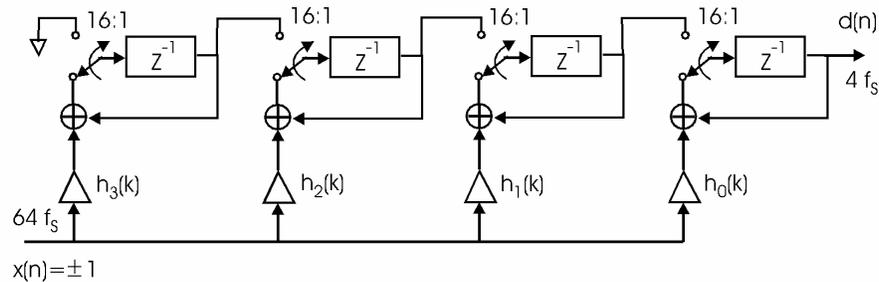


Figure 13.69 Dual Form Filter Showing Accumulators in 4-tap Polyphase Filter

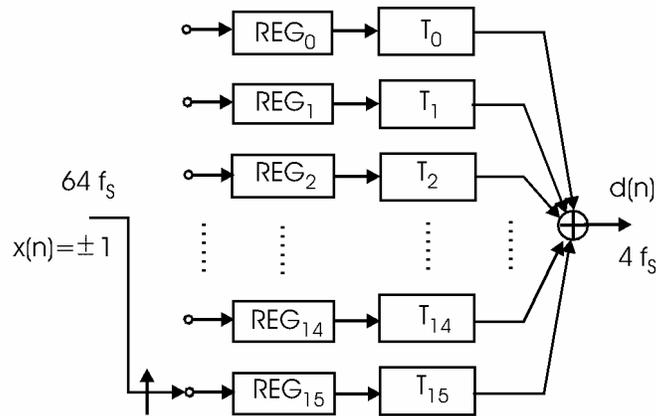


Figure 13.70 Polyphase Filter Implemented with Short Look-up Tables Addressed from Binary Input Data Delivered from Sigma-delta Modulator

Figures 13.71 and 13.72 present the output time series and spectra formed by the polyphase filter performing the first filtering task previously performed by the CIC filter. The two figures correspond to data taken prior to and after the 16-to-1 down sampling operation. Also shown in Figure 13.71 is an overlaid spectral description of the 4-tap 16-arm polyphase filter. Note that the overall filtering effect is the same as that performed by the CIC and presented in Figures 13.66 and 13.67. A housecleaning filter, following the polyphase filter, finishes the filtering task and reduces the sample rate but does not have to correct the sinc/x spectral tilt. Figure 13.73 presents the time response and spectrum of the output from this final 4-to-1 down-sampling filter. Also overlaid on the output spectrum is the frequency response of this 24-tap filter. We also see the spectrum of the final down-sampled time series and note that the spectral terms are 90-dB below full-scale response.

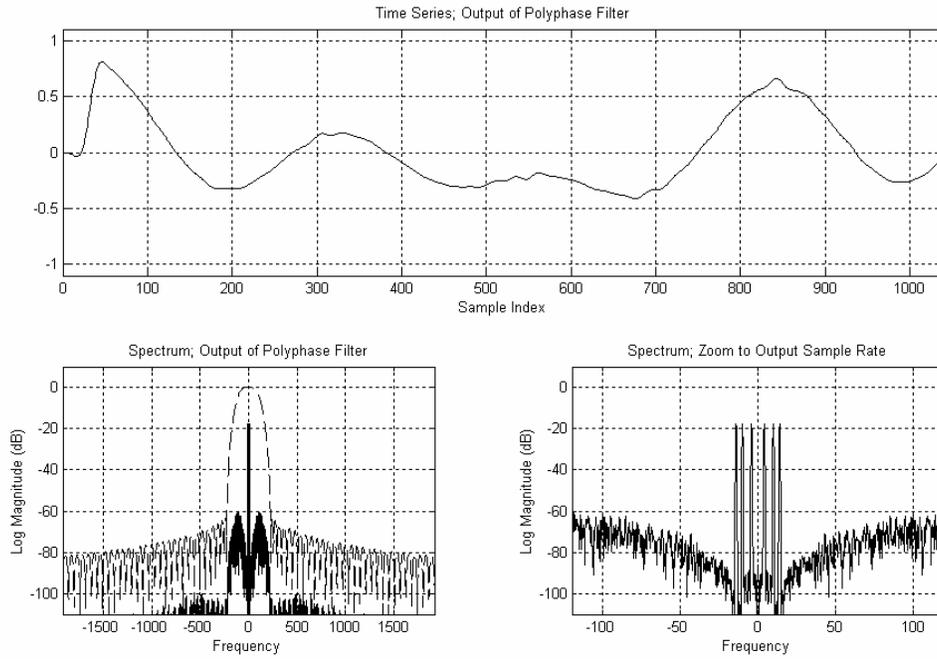


Figure 13.71 Time Series and Spectrum of Polyphase-filtered Output of Sigma-delta Modulator with Overlaid Filter Response and Zoomed Detail of Output Spectrum

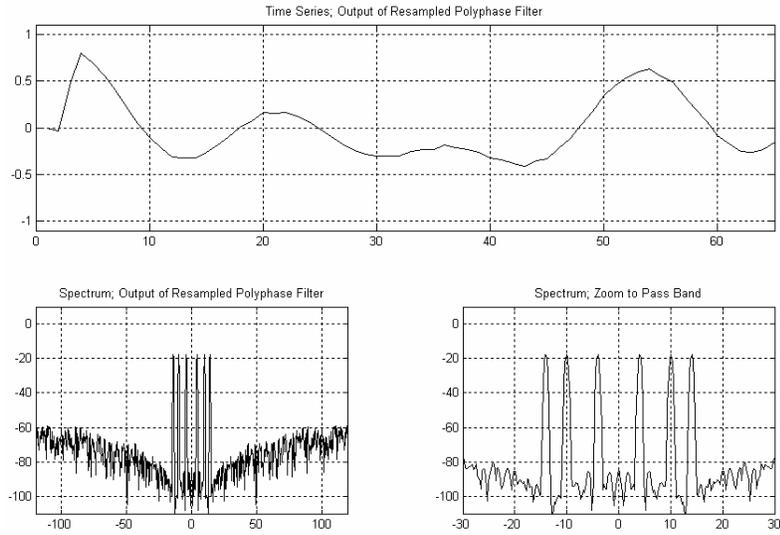


Figure 13.72 Time Series and Spectrum of Polyphase-filtered and Down-sampled Output of Sigma-delta Modulator and Zoomed Detail of Output Spectrum

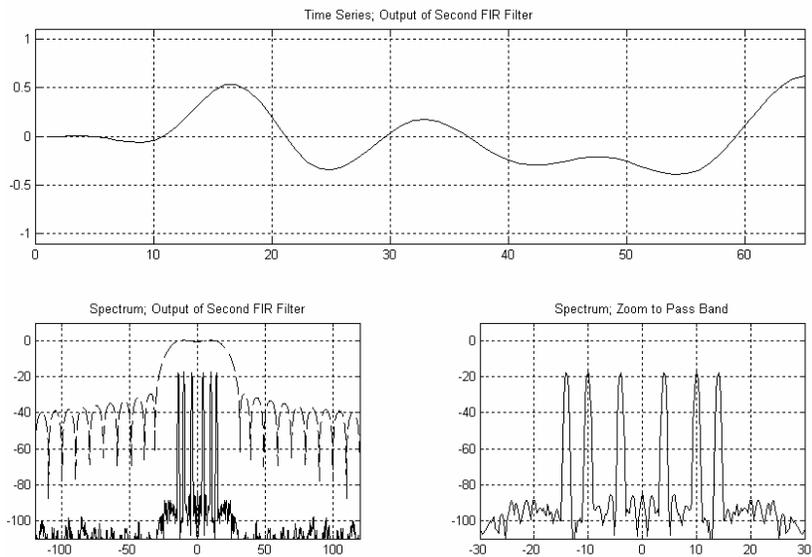


Figure 13.73 Time Series and Spectrum of Final 4-to-1 Down-sampled Filter and Detail of Output Spectrum

13.8 FM RECEIVER AND DEMODULATOR

Multirate signal processing has had a significant influence at the physical or hardware layer of modern communication systems. In particular, multirate signal processing is found at the core of communication systems that couple the Software Defined Radio (SDR) and Software Communications Architecture (SCA) to reconfigure system resources for operation over a wide range of modulation formats and waveforms. In this section we demonstrate one particularly efficient multirate signal-processing solution to the task of implementing a radio configured to select and extract a single FM channel from the commercial FM band, to down convert and demodulate that channel, and to perform stereo demodulation and separation of the resulting baseband signal. Sprinkled through this discussion are descriptions of the necessary background material required to understand the various processing requirements.

In the United States, the commercial FM band spans the frequency interval from 88 MHz to 108 MHz with the different FM stations allocated carriers separated by 200 kHz spacing starting at 88.1 MHz. European FM stations are separated by 100 kHz intervals. The frequency modulation index of the FM signal is 75 kHz, which from Carson's rule leads to a nominal two-sided bandwidth of approximately 180 kHz. The compatible stereo signal, commonly carried by FM stations, is a pair of 15 kHz bandwidth audio signals transmitted as the sum ($L + R$) and difference ($L - R$) of the desired audio component signals denoted L and R for left and right respectively. The $L + R$ signal resides at baseband while the $L - R$ signal is double-sideband suppressed carrier (DSB-SC) modulated to a 38 kHz AM subcarrier. A 19-kHz pilot signal, inserted between the $L + R$ baseband signal and the $L - R$ subcarrier signal, is extracted by the receiver and frequency-doubled to obtain a phase-coherent reference signal required for the DSB-SC down conversion. The separated signals are then summed and differenced to form $2L$ and $2R$, the desired stereo components. Figure 13.74 presents a block diagram of a stereo FM receiver as well as the spectral representation of the input and output spectra of the receiver. Also shown in this figure is a second subcarrier, the subsidiary carrier authorization (SCA) signal, carried by many stations.

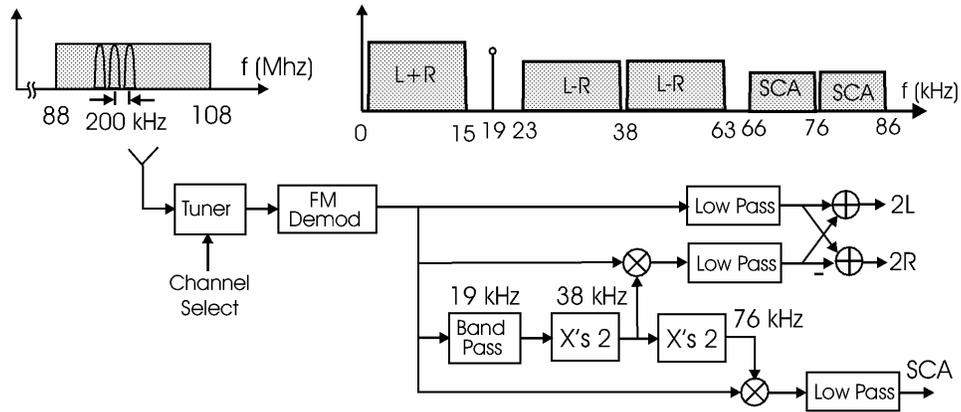


Figure 13.74 Input and Output Spectra of Stereo FM Receiver and Block Diagram of Conventional FM Receiver and Stereo Demodulator

13.8.1 FM Band Channelizer

The tuner section of the FM receiver can be implemented as shown in Figure 13.75. The analog section of the receiver contains an antenna, a band-pass filter with bandwidth matching the 88-to-108 MHz FM band, a gain controllable RF amplifier, and a 10-bit ADC converter operating at an 80-MHz sample rate. The ADC operating at the 80-MHz sample rate performs IF sampling and aliases the FM band centered at 98 MHz to 18 MHz, close to the quarter-sample rate of the converter.

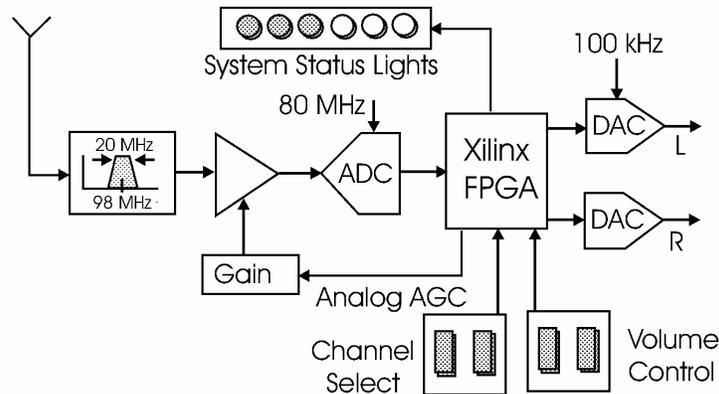


Figure 13.75 Block Diagram of DSP-based FM Channelizer

The digital section of the receiver, residing in the Field Programmable Gate Array (FPGA), contains a polyphase filter that restricts the signal bandwidth and reduces the sample rate 200-to-1 to 400-kHz. The 400-kHz rate was selected to permit wide transition bandwidth in the filter. The spectral response of the baseband prototype filter designed for the 200-to-1 resampling filter is shown in Figure 13.76. A 1200-tap FIR filter operating at 80-MHz sample rate satisfies the spectral specifications indicated in this figure. When implemented as a 200-path polyphase filter, the length of each path is a reasonable 6-taps, which requires six operations per input sample. The total workload per input sample is increased when we include the complex phase rotators required to extract the desired signal. We now examine a few options that accomplish the channel selection along with the bandwidth and sample rate reduction afforded by the polyphase filter partition.

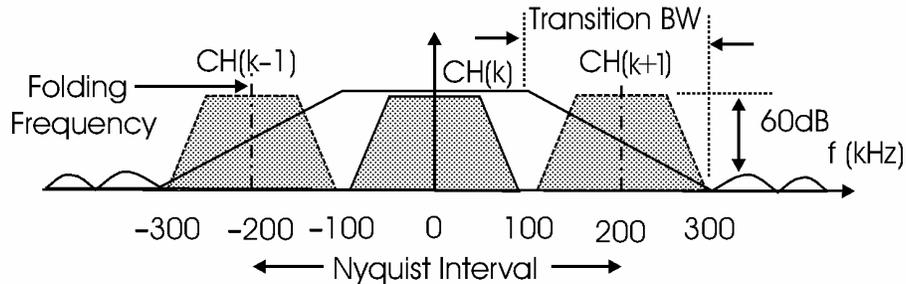


Figure 13.76 Spectral Characteristics of Prototype Low-pass Filter in Polyphase Receiver

The first approach to extract the desired channel from the sampled data stream is a standard complex heterodyne and low pass filter pair with the filters implemented in a polyphase structure that performs simultaneous bandwidth reduction and sample rate reduction. This option is shown in Figure 13.77. Here we see that the workload for the real or imaginary output is 7-multiplies per input sample, 1-multiply for the heterodyne and 6-multiplies for the polyphase filter. The total workload for this implementation is 14-multiplies per input sample point.

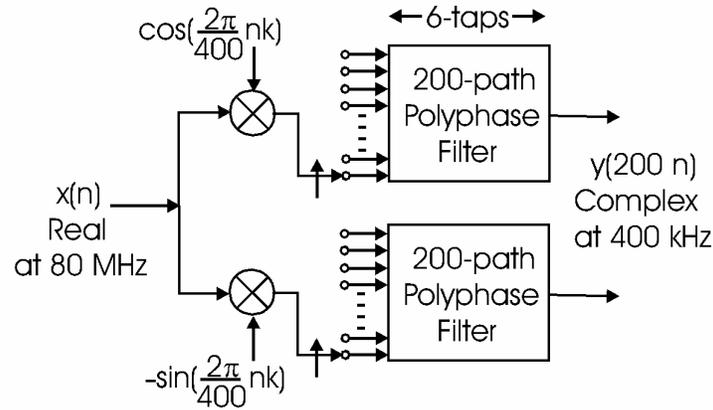


Figure 13.77 Standard Heterodyne, Filter, and Down-sample Architecture Channel Selector

We can apply the equivalency theorem to this channel-selection task to obtain a desired reduction in processing load. Rather than downconvert the selected channel to baseband by an on-line heterodyne, we can upconvert the prototype low pass filter to the selected channel as an off-line heterodyne. The savings we incur is that the on-line post-filtering downconversion, if necessary, occurs at the reduced output rate rather than at the high input rate. The band-pass version of the channel selection process is shown in Figure 13.78. In this system, a signal centered at 8.0 MHz, the image of 88.0 MHz prior to IF sampling, aliases to baseband in the 200-to-1 down sampling to 400 kHz. Thus a signal centered at 8.1 MHz, the first FM channel, aliases to 100 kHz, the quarter-sample rate at the 400 kHz output rate. The required complex heterodyne following the down sampling is a simple translation from the quarter sample rate to baseband. This operation is performed without actual multiplication since the values of the cosine and sine can be restricted to ± 1 and 0. The workload for this form of the channelizers is 6-multiplies per input sample for each of the real and imaginary output port, thus the total workload is 12-multiplies per input sample. By embedding the heterodyne in the filter, we have saved the workload required by the heterodyne process.

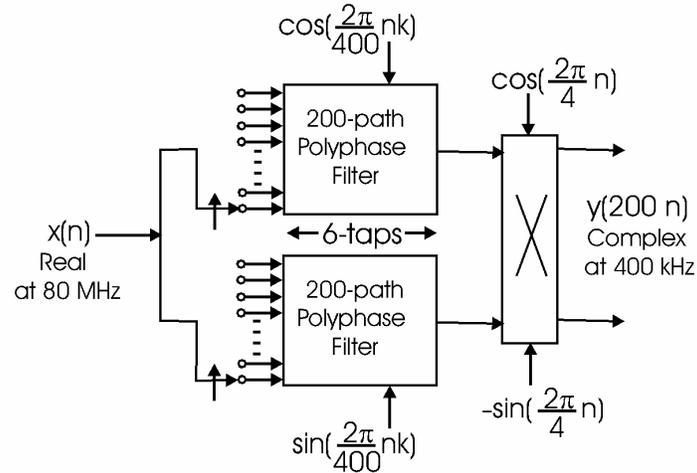


Figure 13.78 Heterodyned Filter, Down-sample, and Down-convert Architecture
Version of Channel Selector

A lesson learned early in multirate filtering is that it is wiser to make the data complex as we leave a processing algorithm than it is to make it complex as we enter the algorithm. In the standard channel-centered polyphase filter structure, the data is made complex by the output phase rotators that unwrap the desired alias from the multiple aliases that have been baseband by the aliasing caused by the resampling process. In the standard polyphase filter structure all multiples of the output sample rate alias to baseband. In the system we are examining, these frequencies are all multiples of 400 kHz. As noted in the previous paragraph, the original 88 MHz, aliased to 8.0 MHz due to the IF sampling, is a multiple of 400 kHz and folds to zero frequency in the standard polyphase structure. The first FM channel is centered 100 kHz above 88.0 MHz, and this frequency will fold to the quarter sample rate at the output sample rate of the polyphase filter. In fact, all the desired channels will fold to ± 100 kHz. We now examine a variant of the standard polyphase filter structure that will fold and translate the FM spectral centers to zero frequency. Equation (13.15) presents the Z-transform of the frequency-translated version of the prototype filter impulse response while (13.16) presents the 1-to-M polyphase partition of the same.

$$H(Z) = \sum_{n=0}^{N-1} h(n) e^{j\frac{2\pi}{M}nk} Z^{-n} \quad (13.15)$$

$$\begin{aligned}
H(Z) &= \sum_{r=0}^{M-1} \sum_{n=0}^{\frac{N}{M}-1} h(r+nM) e^{j\frac{2\pi}{M}(r+nM)k} Z^{-(r+nM)} \\
&= \sum_{r=0}^{M-1} e^{j\frac{2\pi}{M}rk} Z^{-r} \sum_{n=0}^{\frac{N}{M}-1} h(r+nM) e^{j\frac{2\pi}{M}nMk} Z^{-nM}
\end{aligned} \tag{13.16}$$

When the frequency index k is an integer, $2\pi nk$ is congruent to 2π , and the selected frequency bin, bin k , aliases to zero in the polyphase partition. A variant of this relationship is to be seen by replacing k with $k + s/4$, for $s = 0, 1, 2$, or 3 . This is shown in (13.17) where we see that the inner sum representing the operation of the polyphase stages still has a phase shift that varies with the time index n . For the example developed here, the residual phase-shift term is trivial, simply being powers of j . In operation, when the path coefficients are loaded into the path filters, the coefficients are rotated by the path rotations $\exp(j 0.5 \pi n)$ for $s = 1$ or $\exp(-j 0.5 \pi n)$ for $s = 3$. This pre-rotation of the weights results in the successful down conversion, by the resampling operation, of the frequency components of the FM channels offset ± 100 kHz from the multiples of 400 kHz. Note that the offset is also embedded in the phase rotators on each polyphase arm that are applied in the outer summation of (13.17).

$$\begin{aligned}
H(Z) &= \sum_{r=0}^{M-1} e^{j\frac{2\pi}{M}r(k+s/4)} Z^{-r} \sum_{n=0}^{\frac{N}{M}-1} h(r+nM) e^{j\frac{2\pi}{M}nM(k+s/4)} Z^{-nM} \\
&= \sum_{r=0}^{M-1} e^{j\frac{2\pi}{M}r(k+s/4)} Z^{-r} \sum_{n=0}^{\frac{N}{M}-1} h(r+nM) e^{j\frac{2\pi}{4}ns} Z^{-nM}
\end{aligned} \tag{13.17}$$

embedding the j phase rotator in the path weights has a slight impact on the structure of the polyphase filter arms and the subsequent phase rotator. While no actual complex products are involved in the polyphase arms, the data samples formed by the polyphase arms are now complex rather than real. This means that the formerly complex scalar phase rotators applied at the stage output now requires a full complex product. The structure of the modified polyphase filter is shown in Figure 13.79. The workload for this version of the channel select-process is seen to be 6-multiplies per input sample for the polyphase filter and 4-multiplies for the output complex rotator for a total workload of 10-multiplies per input sample. This is the most efficient of the three variants of the channel selection process we have examined.

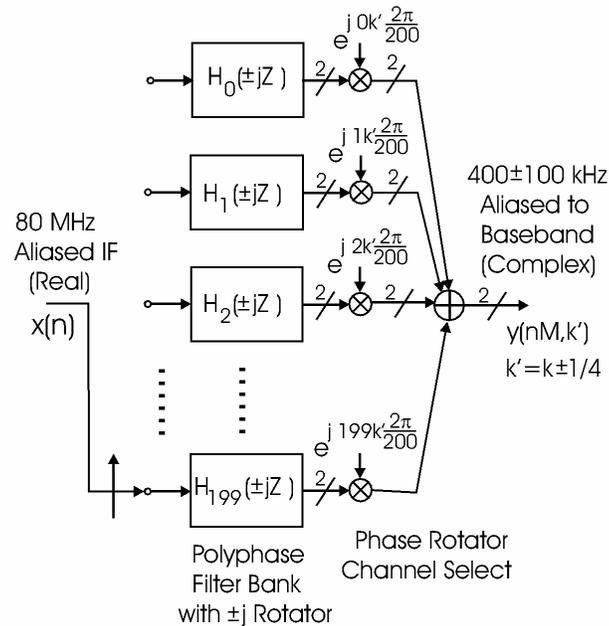


Figure 13.79 Signal Flow Diagram of Modified Polyphase Filter to Permit Frequencies Offset by Quarter of Output Sample Rate to Alias to Baseband

13.8.2 FM Demodulator

For completeness, we now present a short description of the digital FM demodulator. An FM demodulator performs the task of extracting the modulation signal from the FM modulated waveform. Since frequency is the time derivative of a sinusoid's phase angle, an FM demodulator must form the derivative of the received signal's phase. In analog systems this process is accomplished by a circuit called a discriminator that applies a sequence of operators to the carrier-centered FM signal. These operators are a hard-limiter that removes incidental AM from the signal, a derivative circuit that converts the FM signal to an AM signal, and a diode detector that responds to the resulting amplitude variations. A balanced version of this system is used to linearize the derivative process while canceling its DC component.

In the digital receiver, the channel-selection process forms a complex baseband signal with the $I(n)$ and $Q(n)$ components defining the complex envelope of the FM signal. The I-Q components define the modulated phase angle of the original FM signal, and the digital FM demodulator must form the derivative of that phase angle. One implementation forms the angle $\theta(n)$ of the ordered pair by an arctangent or by a CORDIC routine and then forms its derivative with a FIR filter. In another implementation, the derivative of the arctangent is formed in a single step by the relationship shown in equation (13.18) and illustrated in Figure 13.80 without the denominator scaling. The scaling term serves the same function as the

hard limiter in an analog discriminator, that of removing amplitude modulation from the input signals.

$$\dot{\theta}(n) = \frac{d}{dn} \arctan\left(\frac{I(n)}{Q(n)}\right) = \frac{I(n)\dot{Q}(n) - \dot{I}(n)Q(n)}{I^2(n) + Q^2(n)} \quad (13.18)$$

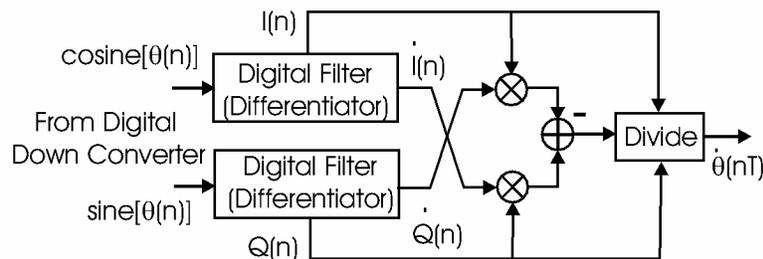


Figure 13.80 Signal-flow Diagram of Digital FM Discriminator

13.8.3 Stereo Decoding

Stereo decoding is a straightforward task that closely follows the signal flow suggested in Figure 13.74. The interesting part of this problem is the processing to extract the pilot signal and then frequency double it for use in the demodulation of the DSB-SC modulated L – R signal. We first examine the length of the filter required to isolate the pilot signal and then present a clever multirate implementation of the pilot extraction and frequency-doubling task. The brute-force filter designed to isolate and extract the pilot signal would have to satisfy the specifications indicated in Figure 13.81. A filter with approximately 273 taps is required to satisfy these specifications. Since the filter bandwidth is a very small fraction of the sample rate, we know that we can use multirate signal processing to implement a more efficient option. We now examine one such option based on down sampling, filtering, and up sampling with a delightful little twist unique to this application.

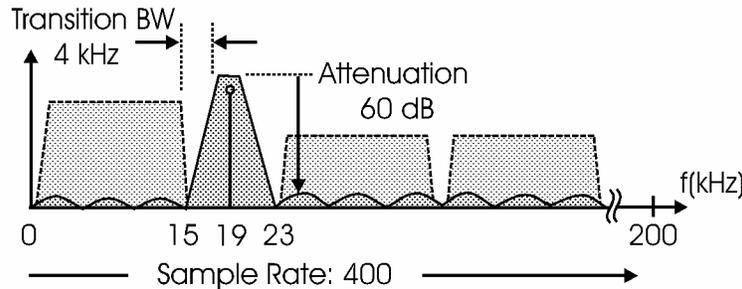


Figure 13.81 Filter Specifications for Pilot Extraction Filter

The structure of an alternate pilot extraction filter is shown in Figure 13.82 and the spectra of the signal at various points in the processing scheme can be seen in Figure 13.83. In this structure we perform a 20-to-1 downsample with a 20-stage polyphase filter with phase rotators that extracts the first Nyquist zone centered at 20 kHz. The prototype low-pass filter used in this partition has a large transition bandwidth equal to 19 kHz, which requires a filter with 60 taps. The output of the phase-rotated polyphase filter contains the alias of the 19-kHz pilot now located at -1.0 kHz with a sample rate of 20 kHz. The filter following the polyphase filter limits the bandwidth around the pilot signal, and it is designed as a low pass filter with two-sided bandwidth of 2 kHz and transition bandwidth of 2.0 kHz. This filter requires 30 taps. The complex signal output by the low pass filter contains only the aliased pilot signal. We now double the frequency of this signal from -1 kHz to -2 kHz by squaring the complex samples. We now up sample the double frequency aliased pilot by a factor of 1-to-20 in a second polyphase filter. The phase rotators at the input to the second polyphase are selected to output a real signal in the second Nyquist zone centered at 40 kHz. This aliases the -2 -kHz baseband signal to 38 kHz, the carrier frequency required for the DSB-SC demodulation of the L – R signal component. The delight found in this process is that the frequency doubling is performed at baseband between the down-sampling and up-sampling processes and that we have selected different Nyquist zones in the down-sampling and up-sampling process.

The workload required for this technique of pilot extraction and doubling can be determined as follows. The input polyphase filter requires 3-multiplies and adds for each path filter and 2-multiplies and adds for the path phase rotators for a workload of 5-multiplies per input sample. Similarly, the output polyphase filter, the dual of the input filter, also requires 5-multiplies and adds per output sample. The filtering performed at the reduced 20-kHz sample rate requires 30-multiplies and adds for each of the real and imaginary filter legs and 4-multiplies and 2-adds in the complex product that doubles the frequency of the extracted pilot. The total of 64-multiplies is amortized over 20 input or output samples so that the filtering load is 3.2-multiplies per input. The total workload, the sum of these terms, is seen to be approximately 13.2-multiplies and adds per input-output pair. This is a significant improvement relative to the 273-multiplies and adds required of the brute-force, nonmultirate filter implementation.

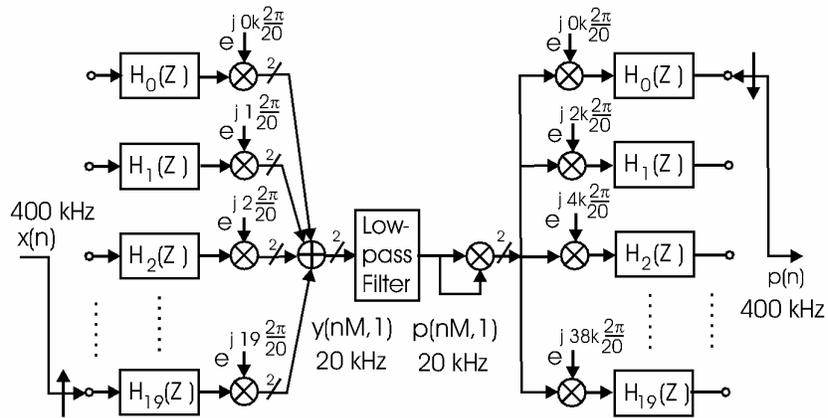


Figure 13.82 Pilot Extraction and Doubling by Polyphase Down-sample, Filter, Frequency Doubling, and Polyphase Up-sample

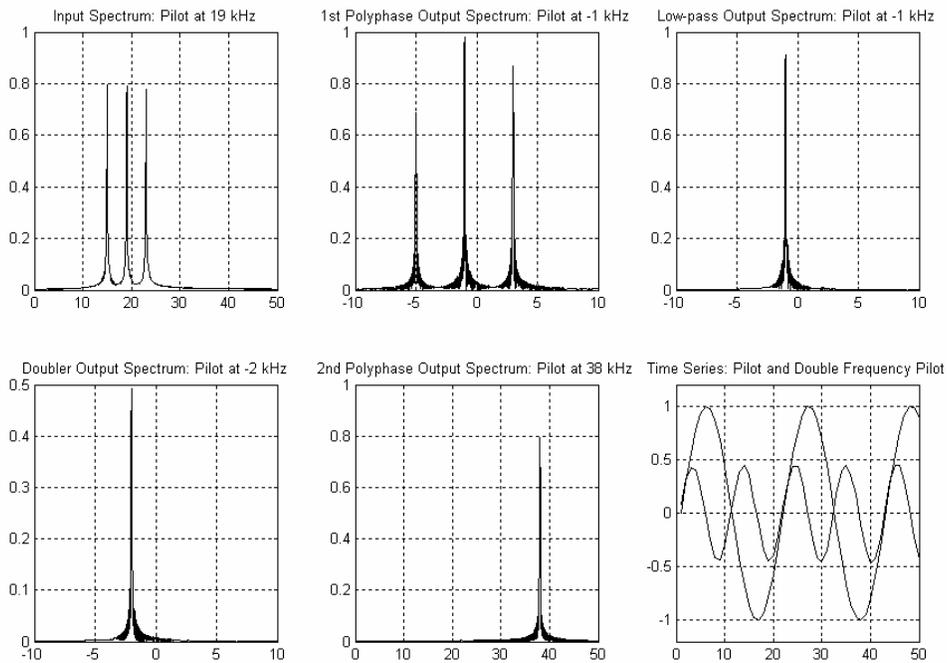


Figure 13.83 Spectrum of Input and Output of Polyphase Down Sampler, at Output of Low-pass Filter, at Output of Squaring Circuit, and at Output of Poly-

phase Up-sampling Filter. Also Time Series of Pilot Component of Input Signal and Double Frequency Pilot at Output of Process

References

- Aziz, Pervez, Henrik Sorenson, and Jan Van Der Spiegel, "An Overview of Sigma Delta Converters: How a 1-bit ADC Achieves more than 16-bit Resolution," *IEEE Signal Processing Magazine*, Vol. 13, No. 1, pp. 61–84, Jan. 1996.
- Candy, James, and Gabor Temes, *Oversampling Delta-sigma Data Converters: Theory, Design, and Simulation*, Piscataway, NJ, IEEE Press, 1992.
- Crochiere, Ronald, and Lawrence Rabiner, *Multirate Signal Processing*, Englewood Cliffs, NJ, Prentice-Hall, Inc., 1983.
- Dick, Chris., fred harris and Michael Rice, "Synchronization in Software Radios: Carrier and Timing Recovery Using FPGAs," IEEE Symposium on Field-Programmable Custom Computing Machines, Napa Valley, CA, April 16–19, 2000.
- Fliege, Norbert, *Multirate Digital Signal Processing: Multirate Systems, Filter Banks, Wavelets*, West Sussex, John Wiley & Sons, Ltd., 1994.
- harris, fred, "Carrier Synchronization Techniques for DSP-based Modems," *Communication Systems Design Magazine*, (Trade Journal), pp. 32–34, 36, 38–39, 42, July 2000.
- harris, fred, "A Fresh View of Digital Signal Processing for Software Defined Radios, Part I and Part II," International Telemetry Conference (ITC), San Diego, CA, Oct. 21–24, 2002.
- harris, fred, "Sigma-Delta Converters in Communication Systems," *Encyclopedia of Communications*, John Wiley & Sons, John Proakis, Editor, 2002.
- harris, fred, "Teaching MODEM Concepts and Design Procedure with MATLAB Simulations," Signal Processing Education, Workshop, Kent, Texas, Oct. 15–18, 2000.
- harris, fred, and Chris Dick, "On Structure and Implementation of Algorithms for Carrier and Symbol Synchronization in Software Defined Radios," URSIPCO-2000, European Association for Signal Processing, Session on Efficient Algorithms for Hardware Implementations of DSP Systems, Tampere, Finland, Sep. 5–8, 2000.
- harris, fred, and Chris Dick, "Performing Simultaneous Arbitrary Spectral Translation and Sample Rate Change in Polyphase Interpolating and Decimating Filters in Transmitters and Receivers," 2002-Software Defined Radio Technical Conference, San Diego, CA, Nov. 11–12, 2002.
- harris, fred, and Michael Rice, "Multirate Digital Filters for Symbol Timing Synchronization in Software Defined Radios," *IEEE Journal on Selected Areas in Communications*, Vol. 19, pp. 2346–2357, Dec. 2001.
- Hentschel, Tim, *Sample Rate Conversion in Software Configurable Radios*, Norwood, MA, Artech House, Inc., 2002.
- Jovanovic-Dolecek, Gordana, *Multirate Systems: Design and Applications*, London, Idea Group, 2002.
- Mitra, Sanjit, *Digital Signal Processing: A Computer-based Approach*, 2nd ed., New York, McGraw-Hill, 2001.

- Mitra, Sanjit, and James Kaiser, *Handbook for Digital Signal Processing*, New York, John Wiley & Sons, 1993.
- Norsworthy, Steven, Richard Schreier, and Gabor Temes, *Delta-sigma Data Converters: Theory, Design, and Simulation*, Piscataway, NJ, IEEE Press, 1997.
- Rice, M. and harris, f., "Polyphase Filter Banks for Symbol Synchronization in Sampled Data Receivers," MILCOM-2002, Anaheim, CA, Oct. 7–10, 2002.
- Rice, M., harris, f., and Chris Dick, "Maximum Likelihood Carrier Phase Synchronization In FPGA-based Software Defined Radios," ICASSP-2001, International Conference on Acoustics, Speech and Signal Processing, Salt Lake City, Utah, May 7–11, 2001
- Vaidyanathan, P. P., *Multirate Systems and Filter Banks*, Englewood Cliffs, NJ, Prentice-Hall, Inc., 1993.

Problems

- 13.1 Equation 13.1 presented the Taylor series for $[\sin(\theta)/\theta]^P$, the main lobe response of the CIC filter. Determine the value of θ for which the droop due to the main lobe curvature is -0.1 -dB and then 0.5 -dB. What signal bandwidth, as a fraction of the sample rate, corresponds to these values of θ ?
- 13.2 A signal with two-sided bandwidth of 20 kHz is centered on an analog 450 kHz intermediate frequency carrier. Suppose we select a sample rate of 360 kHz to perform IF sampling and alias the signal from the center frequency to a new digital-center frequency. Draw a block diagram of this process. Show where the spectrum is located, and describe the remaining processing required to bring the signal to baseband with filtering and resampling to 90-kHz output sample rate.
- 13.3 A signal with two-sided bandwidth of 20 kHz is centered on an analog 450-kHz intermediate frequency carrier. Suppose we select a sample rate of 120 kHz to perform IF sampling and alias the signal from the center frequency to a new digital center frequency. Draw a block diagram of this process. Show where the spectrum is located, and describe the remaining processing required to bring the signal to baseband with filtering and resampling to 40-kHz output sample rate. Be careful of spectral inversion.
- 13.4 The information a receiver needs to determine correct timing information from a received signal is obtained from the output of a matched filter and the derivative matched filter. We now examine the transfer function of the time error detector. Form a SQRT-Nyquist pulse oversampled by 10-to-1. Also form a corresponding matched filter and a derivative matched filter with the same over sampling ratio. Be certain that the derivative filter and the matched filter are time aligned, that is, that the zero crossing of the derivative filter matches the peak of the matched filter. Convolve the impulse response of the shaping filter with impulse response of the matched filter and of the derivative matched filter. Form the dot product of the two responses, and plot the results. This function is the timing error detector function. Determine its slope in units of amplitude per time-offset. If the amplitude of the input signal is doubled, what happens to the gain of the time offset error detector? How does this impact the time-offset error for a QAM signal as opposed to a QPSK signal?

- 13.5 Determine how to form a band-edge filter from a matched filter operating at 4-samples per symbol. Hint: This will require a 4-path polyphase partition, but no resampling, of the matched filter. Where is the band edge located for this process? Can we reduce the sample rate by a factor of 2 as part of separating the two band edges? Why would one try to do this?
- 13.6 Form a simple band-edge filter by operating the matched filter at 2-samples per symbol and extracting the overlap of the low pass and high pass polyphase partition with a Hilbert transform filter. Feed the band-edge filters from the output of a shaping filter also running at 2-samples per symbol and modulated with 16-QAM random data. Introduce selectable frequency offsets to the modulated signal, and form a plot of the average output level from the energy difference of the two band-edge filters as a function of frequency offset. Be sure to test both positive and negative offsets. Determine the gain of the frequency-offset detector. Is this gain a function of modulation constellation density? Is this gain a function of input signal level?
- 13.7 A simple digitally controlled, digital delay line can be formed by a cascade of one or more identical all-pass networks with transfer function $H(Z) = (1 - \alpha Z)/(Z - \alpha)$. This is particularly effective when the signal bandwidth is a small fraction of the sample rate, as might occur when a signal is oversampled by a factor of 4. Under this condition, the all-pass network exhibits approximately linear phase shift in the signal bandwidth. Determine the expression for phase slope of the all-pass network. You will find assistance in Chapter 10. Then determine group delay as a function of the parameter α .
- 13.8 Form a delay line using three stages of the recursive one-tap all-pass filter presented in Chapter 10. Form a SQRT-Nyquist pulse that is oversampled by a factor 10-to-1 and then pass the SQRT-Nyquist pulse through the delay line filter chain with the parameter α set to zero, plot the input and output signal, and verify the delay is 3-samples. Repeat the experiment for a range of α between -0.5 and $+0.5$ in increments of 0.1 . Examine the relationship between the observed delay and the setting for the parameter α .