A Fresh View of Digital Signal Processing for Software Defined Radios: Part II

fred harris
CUBIC Signal Processing Chair
San Diego State University
fred.harris@sdsu.edu

ABSTRACT

A DSP modem is often designed as a set of processing blocks that replace the corresponding blocks of an analog prototype. Such a design is sub-optimal, inheriting legacy compromises made in the analog design while discarding important design options unique to the DSP domain. In part I of this two part paper, we used multirate processing to transform a digital down converter from an emulation of the standard analog architecture to a DSP based solution that reversed the order of frequency selection, filtering, and resampling. We continue this tack of embedding traditional processing tasks into multirate DSP solutions that perform multiple simultaneous processing tasks.

KEY WORDS

Software Radio Algorithms, Modern DSP Architectures, Resampling Channelizer

INTRODUCTION

Digital signal processing (DSP) is often taught by emphasizing the similarity between the operations performed in the continuous analog and the discrete sampled data domains. Convolution, spectral analysis, transfer functions, and pole-zero patterns are common to both domains. Consequently it is a natural progression for a designer to transfer an existing analog design to a DSP based design. For instance, if a prototype radio receiver's final IF stage quadrature down converts with a pair of mixers and analog Butterworth low-pass filters, the digital version mimics it and also quadrature down converts with a pair of multipliers and discrete Butterworth filters. The design often works, but is likely a poor use of the new technology. We note the obvious: that when DSP replaces, block-by-block, the elements of an analog prototype the resulting system is an emulation of the analog solution, rather than the hoped for enhanced replacement of that radio. Also note that the Butterworth filter may have been the best analog filter in the analog processing chain but may not be the best filter at the corresponding position in the digital processing chain. Perhaps a linear phase FIR filter may offer improved performance over the Butterworth filter! We must keep in mind those digital designs that mimic analog prototypes inherit legacy solutions embedded in the analog prototype solution that may contain compromises valid for an analog solution, but not necessary for a DSP based solution. More importantly, by limiting the design perspectives to an analog viewpoint we discard important DSP based solutions for which there are not analog counterparts.

As hardware capabilities permitted, the DSP insertion point in a receiver signal flow path has steadily migrated toward the antenna in the radio signal-processing path. A consequence of this migration is that DSP now performs nearly the entire back end processing tasks in modern receivers. Digital processing replaces the analog quadrature down conversion, the base-band filtering, and the sample rate selection for post base-band processing. In part I of this paper we converted the standard analog down-conversion process to a multirate digital process that reversed the order of the prototype processing steps. In the multirate variant, spectral translation occurs by aliasing the desired spectral region to baseband via a resampling operation rather than by a conventional complex heterodyne. Since the resampling operation occurs prior to bandwidth reduction, the resampling operation aliases numerous spectral bands to baseband. These frequency bands are centered at kfs/M, integer multiples of the output sample rate fs/M. The M-path filter supplied by the input commutator aligns the M separate output time series to a common time reference. The multiple aliases in each of the M-paths exhibit unique phase profiles across the M-paths, and the phase rotators following the M-path filters phase align the desired alias to extract it from the set of aliases. The phase aligned signal in each path coherently summed, while the remaining aliases, with phases distributed on the M-roots of unity are cancelled in the same sum by destructive cancellation.

The M-path resampling filter can cooperate and participate in other signal processing and signal conditioning tasks required in the radio receiver. We examine a number of these processing tasks and demonstrate a number of architectures that employ the multi-function multi-rate filter. These tasks include spectral translation, bandwidth limiting, and resampling for single and multiple channels, matched filtering, time offset discriminator and timing recovery.

THE CORE OF MODERN DIGITAL RADIO ARCHITECTURES

The core operation performed in a DSP based radio receiver is the down conversion and filtering of the desired narrowband channel present at the output of the IF filter. Figure 1 is a simple block diagram of the digital version of the analog prototype. The replacement of the complex down converter, filter pair, and M-to-1 down sampler with the multirate resampling related alias translation is shown in figure 2. Doppler offsets and local carrier offsets are removed from the complex baseband signal

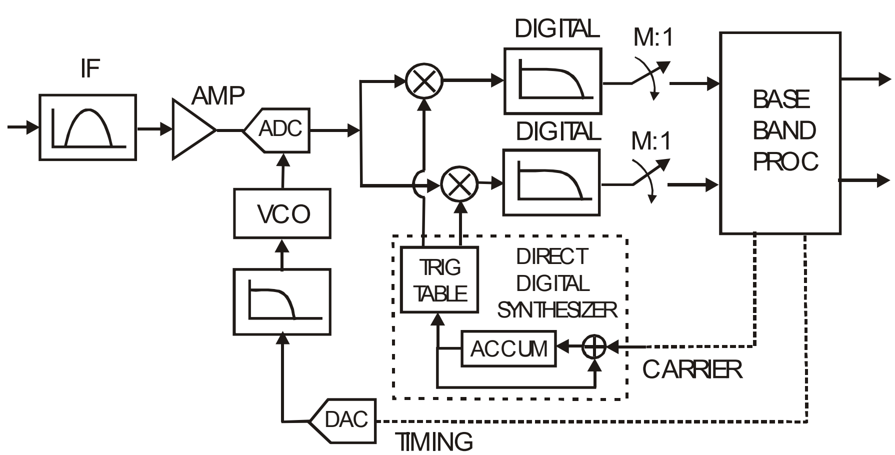

Figure 1. Digital Down Converter: Mimicking Analog Prototype Down Converter

at the output of the multirate filter. There are significant advantages to performing phase locking on down converted signal rather than by controlling the down conversion process as shown in figure 1. The primary advantage is that the transport delay of the baseband low pass filter in no longer in the control loop, hence the loop can be operated with faster response time.

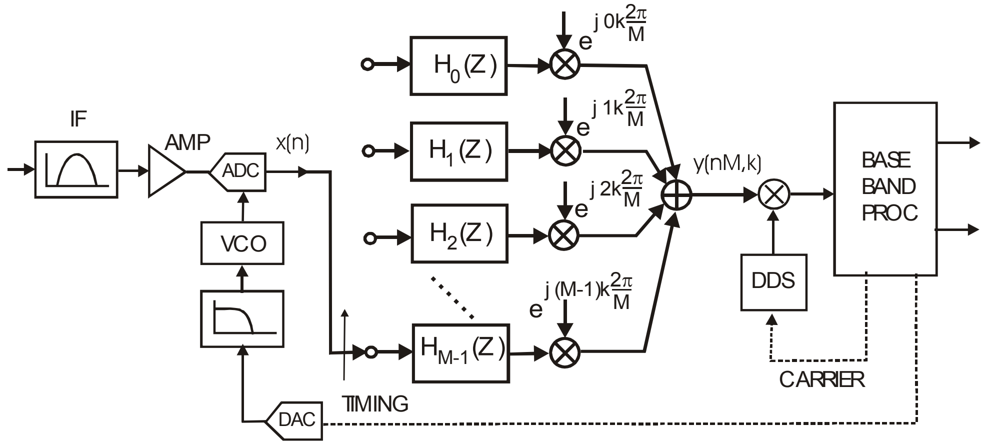

Figure 2. Digital Down Converter: Resampling Alias Based Down Converter

A number of additional advantages accrue to the M-path partitioned down converter. The first is that the stages operate at the reduced output rate rather than the higher input sample rate. The second is that the selection of center frequency is performed after the filter rather than before the filter. Consequently, the data entering the M-filter stages is real and does not become complex till the data leaves the filter and interacts with the complex phase rotators. A third advantage is that since the phase rotators can extract any of the M-frequency bands that alias to baseband as a result of the resampling, it can extract all of the bands. As shown in part I of this paper, the phase coherent sum formed as the phase rotated stage outputs, is identical to the discrete Fourier transform (DFT) of the path outputs. Thus when a DFT or its computationally efficient cousin the FFT processes the path outputs, the down converter will present outputs from the full set of aliased frequencies. The combination of the M-stage filter and the FFT implements a very efficient channelizer of the spectrum delivered by the IF filter. Two other functions we can have the M-path resampling filter perform are arbitrary resampling between input and output sample rates for single or multiple channels, and timing recovery for single channel down converters. Bare in mind that the M-path filter can also be used as a strictly baseband filter performing the filtering and resampling operation following a conventional complex heterodyne to baseband as shown in figure 1. In this mode, the filter can still be used in the timing recovery task with the advantage of this mode being a reduction in hardware required to accomplish the filtering task.

DIGITAL DOWN CONVERTER EXAMPLE

We now introduce a practical example of a digital down converter to more clearly demonstrate the advantages of the M-path processing option. We consider processing a signal containing 50 FDM channels QAM modulated with 50 % excess bandwidth square-root Nyquist spectra at symbol rate of 128 kHz and separated by 192 kHz center frequencies. The spectrum of this signal set is shown in figure 3. We will examine various options related to demodulating a single channel from this signal set as well as a number of options related to demodulating the full set of channels. The two-sided bandwidth of this composite signal is 50*192 kHz or 9.60 MHz. The sample rate must be greater than the signal bandwidth to accommodate the transition bandwidth of the IF filter. In anticipation of using a 64-point FFT to process multiple-channels the sample rate is selected to be 64*192 KHz or 12.288 MHz. In order to use real sampling of the IF center frequency, the input sample rate must be doubled to 2*12.288 MHz or 24.576 MHz. .

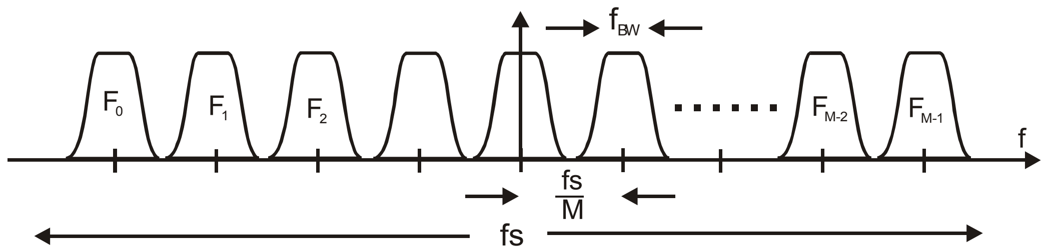

Figure 3. Spectrum of 50-Channel FDM Signal Set

It is desirable to place the center frequency of the signal bandwidth at the quarter-sample rate. We can accomplish this by choosing the IF center frequency to be K±1/4 times the desired sample rate for some K. We choose K to be 1 and find one choice of the IF center frequency to be (5/4)*24.576 MHz or 30.720 MHz. We perform IF sampling, and by sampling the signal centered at the IF frequency (5/4) fs, at a sample rate of fs, the center frequency aliases to (1/4) fs. The frequencies involved in this IF sampling operation are shown in figure 4. Notice that the IF sampling is performed by a single ADC by which we avoid the problem of balancing gain and phase in the two paths of an analog quadrature down converter,

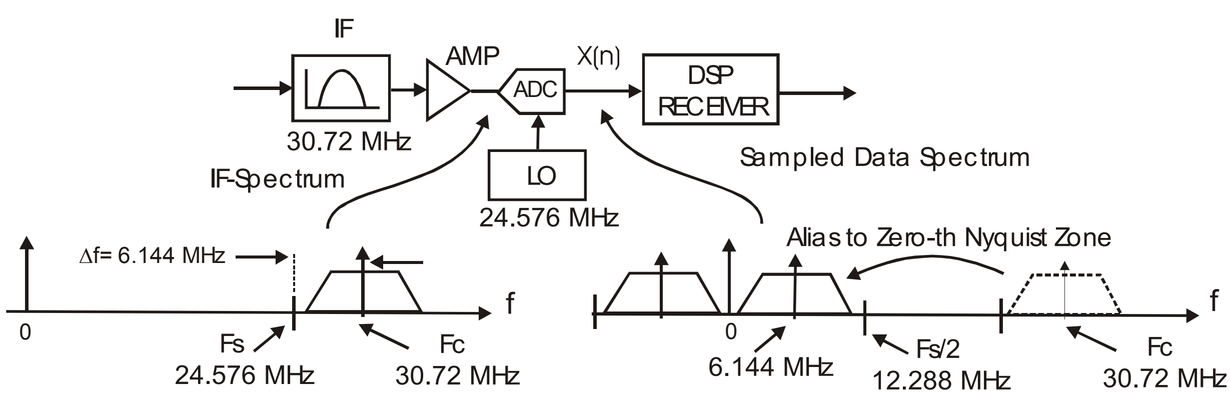

Figure 4. Spectra at Input and Output of ADC

We can now apply a number of different processing options to the sampled IF data to extract, at twice the symbol rate, samples of the complex envelope of a single channel selected from the 50 FDM channels. The first option is a straight digital quadrature down conversion by a pair of multipliers, filtering to the required bandwidth, and down sampling to the required sample rate. This option is shown in figure 5. The length of the FIR filter h(n), required to obtain a 96 kHz transition bandwidth and –72-dB out-of-band attenuation is 960 taps. The output of this filter is down sampled 96:1 to convert input sample rate of 24.576 MHz to output sample rate of 256 kHz. Note that the filter performs 960 ops/output but generates 1 output per 96 inputs for a workload of 10 ops/input. We hold in reserve the option to apply the noble identity to the two filters and perform the down sampling prior to the filter to more easily see the 96-path filter requiring 10 ops/path.

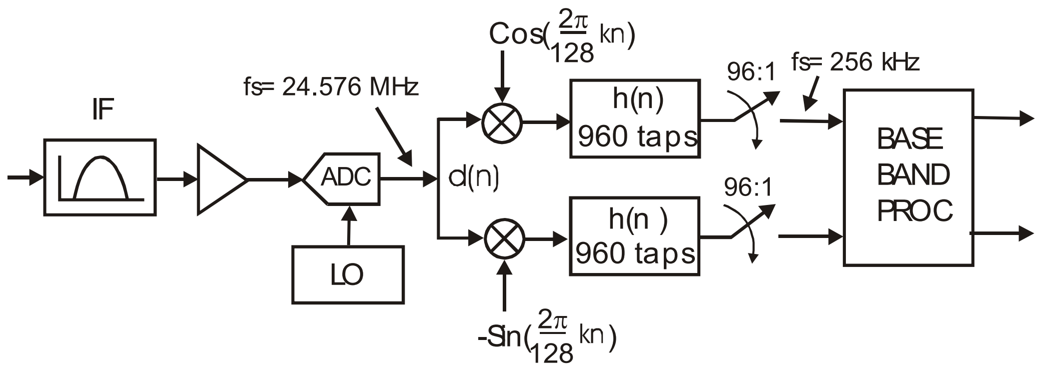

Figure 5. Conventional Complex Heterodyne of Sampled IF Data

An alternate architecture applies a Hilbert transform to the output of the IF sampled data to suppress the negative frequency spectral region and in doing so convert the real input data to complex output data. The rejection of the negative frequency reduces the bandwidth by a factor of two hence permits the reduction in output sample rate by the same factor. The Hilbert transform filter is a half band filter with alternate samples equal to zero, which permits the 2-to-1 resamplers to be pulled through the filter and be replaced with an input commutator. The two filters following the complex heterodyne are each half their former length due to the 2-to-1 reduction in sample rate and operate here at half the previous rate. Here again the filters operate at 10 ops/input but with an input rate reduced by a factor 2. The noble identity can of course also be applied here. This option is shown in figure 6.

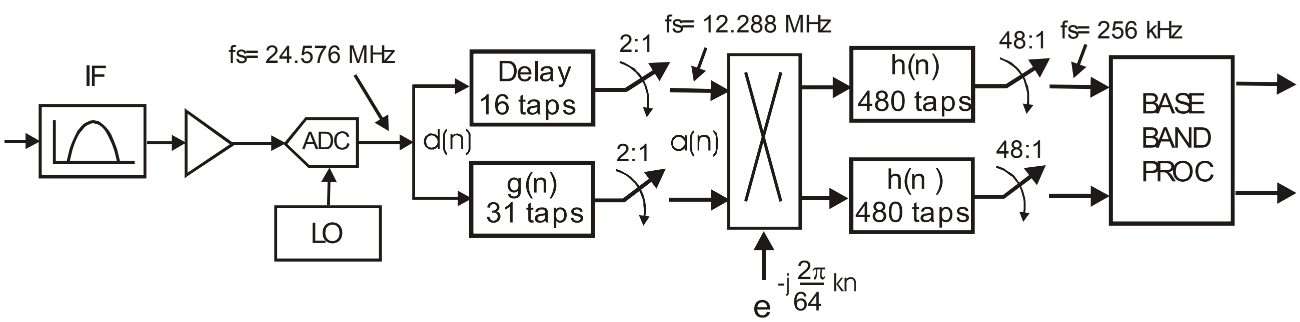

Figure 6. Hilbert Transform Preprocess and Resample of IF Data Prior to Complex Heterodyne

Returning to the architecture shown in figure 5, we can reorder the cascade of complex heterodyne and filter to reflect the equivalency theorem. Here we move the heterodyne into and through the filter, but also continue to move the down converter through the down sampler. By selecting the center frequency of the channels to be multiples of the output sample rate, the channels alias to DC under the resampling operation. The problem here is that the center frequencies, k 192 kHz, which corresponds to digital frequency of $k*2\pi/128$ radians per sample, are not multiples of the output sample rate, 256 kHz. Hence when we down sample by 96 the selected center frequency is not at DC but rather at $k*2\pi*96/128$ or $k*2\pi*3/4$ a residual rotation that must be removed by an output heterodyne. For this example, the rotation is simply powers of "j", a trivial operator. The configuration of this option is shown in figure 7. Note here, the input heterodyne is applied to the filter coefficients as an off line frequency selection rather than to the input data as an on line operation and that the output heterodyne applied on line occurs at the output rate and is, for this example, trivial.

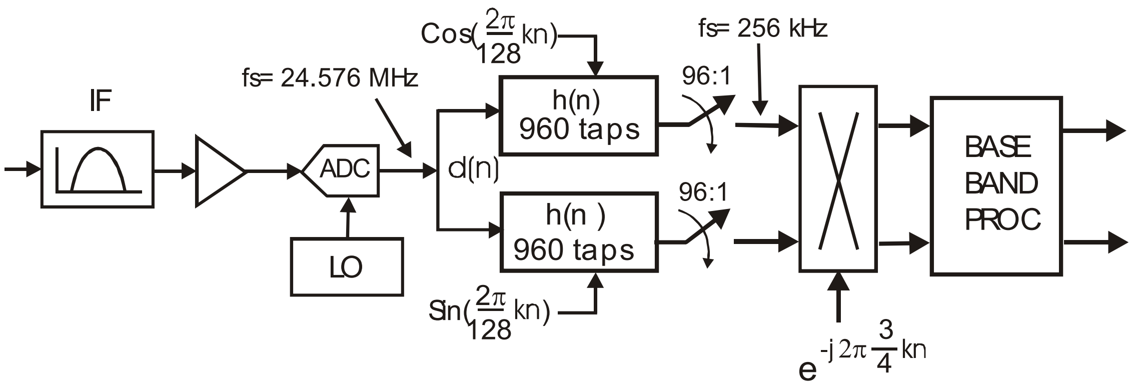

Figure 7. Digital Down Converter as Down Sampled Band Centered Complex Filter

We now invoke the noble identity and pull the 96-to1 down samplers through the complex band-pass filter. This structure is shown in figure 8 where the M-path partition of the cosine and sine heterodyned weights are indicated as $h_r(n)$ and $g_r(n)$ respectively. The M-path filters contain 96 ten tap sub filters, each progressively accessed with successive inputs. Since only one filter path is accessed per input point, we can replace the 96 filters with one (dual geometry) filter stage and a pointer that accesses from memory the successive stage weights with consecutive inputs. This version of the architecture is shown in figure 9. One output is generated per 96 input points with a workload of 10 ops/input on each of the in-phase and quadrature filter paths.

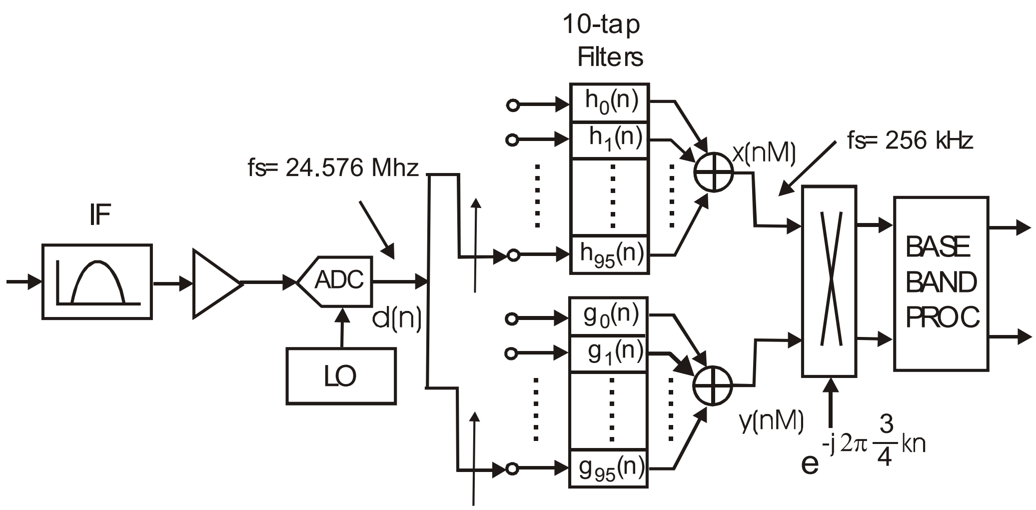

Figure 8. Digital Down Converter as M-Path Partition of Band Centered Complex Filter

Note that in the various arrangements of the filter presented in figures 7, 8 and 9 the weights are complex, and that the data presented to it is real and sampled at 24.576 MHz. We also note that the Hilbert transform pre-processing shown in figure 6 converts the input data to complex data and reduces the sample rate by half to 12.288 MHz. At this reduced rate the filter is half its previous length and operates at half its previous rate, both desirable features. We might be tempted to combine the Hilbert transform preprocess with the heterodyned filtered sets to take advantage of both reduced data rate and translated filter weights. The problem with this notion is that now both the data and the filter weights are complex, and the convolution of complex data with complex weights entails the operation of four filters rather than two filters for the real data with complex filter weights, or complex data with real filter weights. We now modify the resampling band-pass filter so that it presents real weights when seen from its input terminals but presents complex weights when seen from its output terminals. This option will permit the desired use of complex preprocessed input data with the complex band-pass filter with the two-filter convolution rather than the four-filter form.

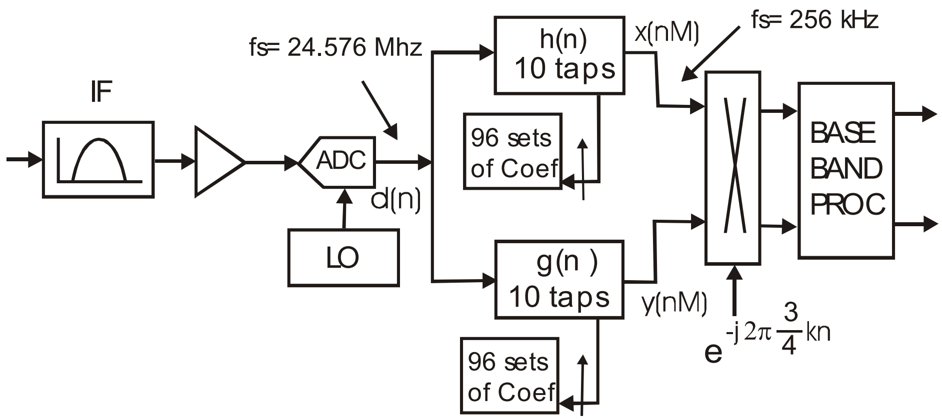

Figure 9. Digital Down Converter Single Path Processing with Commutated Weights from M-Path Partitioned Band Centered Complex Filter

To accomplish this task we first consider a slight variation in how we operate the band-pass resampling filter, discover the benefits of the alternate format, and then return to the original operating scheme. Let us review the current operating conditions of the Hilbert Transform pre-processor. The input sample rate to the filter is 12.288 MHz with desired center frequencies at multiples of 192 kHz so that the center frequencies reside at multiples of fs/64. We require the filter set to enable resampling to 2-samples per symbol or 256 kHz, which represents a resampling ratio of 12288-to-256 or 48-to-1. We know that center frequencies located at multiples of the output sample rate alias to baseband when down sampled. If we were to down sample by 64 rather than by 48 the band-pass signal and filter would both alias to baseband. When the band-pass filter aliases to baseband, the complex heterodyne coefficients default to unity and drop out of the polyphase partition. The complex phase rotators attached to the delays in the partition still survive, and are positioned as scalars applied at the output of each polyphase leg of the M-path filter as shown in figure 2. We can use this filter with real input weights and complex outputs to efficiently translate to baseband, filter, and resample any of the input FDM channels contained in the complex data stream formed by the Hilbert transform pre-processor.

We still have this little problem that the output sample rate is equal to the channel spacing (192 kHz), which is standard for maximally decimated filter banks, rather than the desired 2-samples per symbol rate of 256 kHz. Conventional receivers would, at this point, simply pass the down converted, down sampled data set to a polyphase interpolator to up-sample by 4/3 to obtain the desired 256 kHz sample rate as shown in figure 11. It is common to embed the interpolator in the matched filter to reduce the number of separate filters in the chain. Following this line of reasoning, the matched filter can be folded into the down sampling filter, and had we more time we would in fact do this. What we now do is partition the polyphase filter in the maximally decimated form as if we were going to perform 64-to-1 down sampling as shown in figure 10 and then alter the way we use the partition so that we obtain the 48-to-1 down sampling from the 64-path filter.

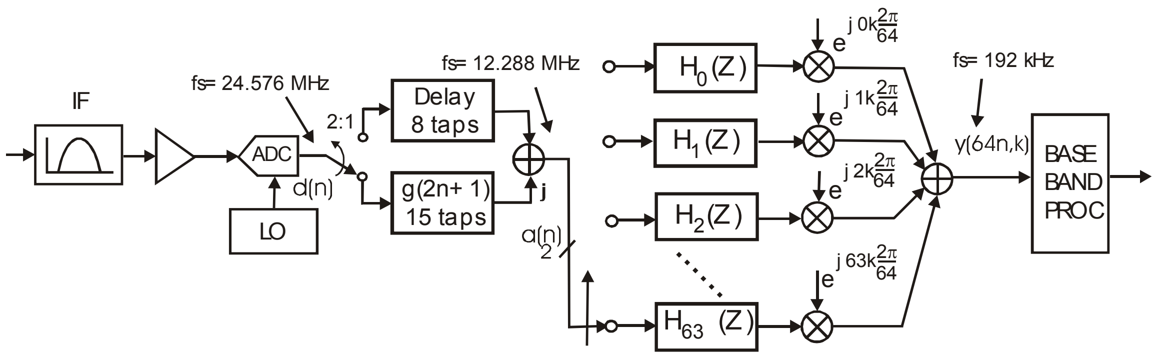

Figure 10. Digital Down Converter: Hilbert Transform Processor and Maximally Decimated Filter

The convolution performed by the non partitioned, filter h(n) translated to the k-th center frequency is shown in eq.1. Here the filter length is increased from 480 to 512 taps to be a multiple of 64. Equation 1 shows the filter partition into 64 8-tap filters and the resulting double sum describes the M-path filter. In eq. 2 we resample the output, taking every 64-th output to see the maximally decimated filter structure shown in figures 2 and 10. We have subscripted the data and coefficient set with the index r to explicitly show that the offset index “r” reflects memory partition of filter and input data buffer.

$$
\begin{aligned}
y(n,k) &= \sum_{m=0}^{511} d(n-m)h(m)e^{j\frac{2\pi}{64}mk} \\
&= \sum_{m'=0}^{7}\sum_{r=0}^{63} x(n-r-64m')h(r+64m')e^{j\frac{2\pi}{64}(r+64m')k} \\
&= \sum_{r=0}^{63} e^{j\frac{2\pi}{64}rk} \sum_{m'=0}^{7} x(n-r-64m')h(r+64m')
\end{aligned} \tag{1}
$$

$$
\begin{aligned}
y(64n,k) &= \sum_{r=0}^{63} e^{j\frac{2\pi}{64}rk} \sum_{m'=0}^{7} x(64n-r-64m')h(r+64m') \\
&= \sum_{r=0}^{63} e^{j\frac{2\pi}{64}rk} \sum_{m'=0}^{7} x(64(n-m')-r)h(r+64m') \\
&= \sum_{r=0}^{63} e^{j\frac{2\pi}{64}rk} \sum_{m'=0}^{7} x_r(64(n-m'))h_r(64m')
\end{aligned} \tag{2}
$$

In eq. 3 we resample the output 48-to-1 to obtain the desired non-maximally decimated filter structure. We note two main differences in the non-maximally decimated filter. First the r-th data set now interacts with cyclic shifts of the coefficient set -16n+r, and second there is a residual heterodyne by powers of “-j”. This is the same shift we observed in figures 7,8, and 9. Notice this heterodyne can be absorbed in the phase rotators as a 16-point offset of the output index r’.

We account for these changes by modifying the standard form of the M-stage filter by including a circular input buffer to roll input data to the offset filter coefficient sets, a circular output buffer to roll output data to absorb the residual output heterodyne, and a state machine to direct the sequence of circular input and output data rolls. A simplified model description of the commutator and filter

$$
\begin{aligned}
y(48n,k) &= \sum_{r=0}^{63} e^{j\frac{2\pi}{64}rk} \sum_{m'=0}^{7} x(48n-r-64m')h(r+64m') \\
&\text{Let r = (-16n + r')MOD(64)} \\
&= \sum_{r'=0}^{63} e^{j\frac{2\pi}{64}(-16n+r')k} \sum_{m'=0}^{7} x(48n-64m'+(16n-r')Mod(64))h((-16n+r')Mod(64)+64m') \\
&= e^{-j\frac{2\pi}{4}nk} \sum_{r'=0}^{63} e^{j\frac{2\pi}{64}r'k} \sum_{m'=0}^{7} x(64(n-m')-r')h(64m'-16n+r') \\
&= e^{-j\frac{2\pi}{4}nk} \sum_{r'=0}^{63} e^{j\frac{2\pi}{64}r'k} \sum_{m'=0}^{7} x_{r'}(64(n-m')-r')h_{(-16n+r')}(64m')
\end{aligned} \tag{3}
$$

bank operating in the two modes, maximally decimated with post processing interpolators and non-maximally decimated, is shown in figure 11. Notice that in the left segment, the commutator spans the full 64 input ports to deliver 64 inputs required to form an output at 1/64-th of input rate while in the right segment the commutator spans 48 input ports to deliver 48 inputs required to form an output at 1/48-th of input rate. Figure 12 shows the data flow of the circular input buffer as successive blocks of 48-points are delivered. We see that the data in the input buffer serpentine advances through the 64-point columns, vacating 48 registers in preparation for the next block of 48 input samples. This serpentine advance is easily accommodated by a circular shift of the top 16 rows to the bottom 16 rows. Also shown here is the circular shifting of the output buffer to absorb the residual heterodyne as a phase shift in the phase rotators.

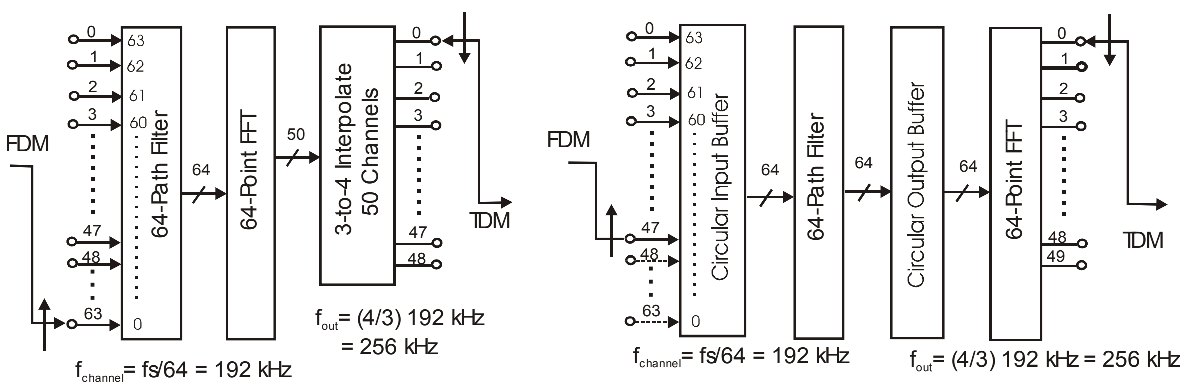

Figure 11. Filter Banks: Maximally Decimated with Interpolators and Non-Maximally Decimated

With the phase rotators applied after the filter, we have the opportunity to apply the phase rotators for all the channels as an FFT. For this mode of operation, the single set of filter weights demodulates, matched filters, and interpolates to 2-sample per symbol, all the channels in the received FDM signal. A final note is that there are 48 different starting phases of the input data stream that can be used to initiate the data load to the 48-stage polyphase filter. Shifting the starting phase under control of the timing recovery loop permits the polyphase filter to perform the time shifts required for timing acquisition. Time offset granularity in the timing loop is related to the number of stages in the polyphase filter bank. One response to this connection is to implement a polyphase bank with a very large number of weights, say 256, and course skip weight sets to obtain the desired 64 sets, but permit fine offset of weight sets to obtain fine grain timing control. A number of consumer modems use this option to obtain matched filtering, arbitrary resampling, and timing recovery in a single process.

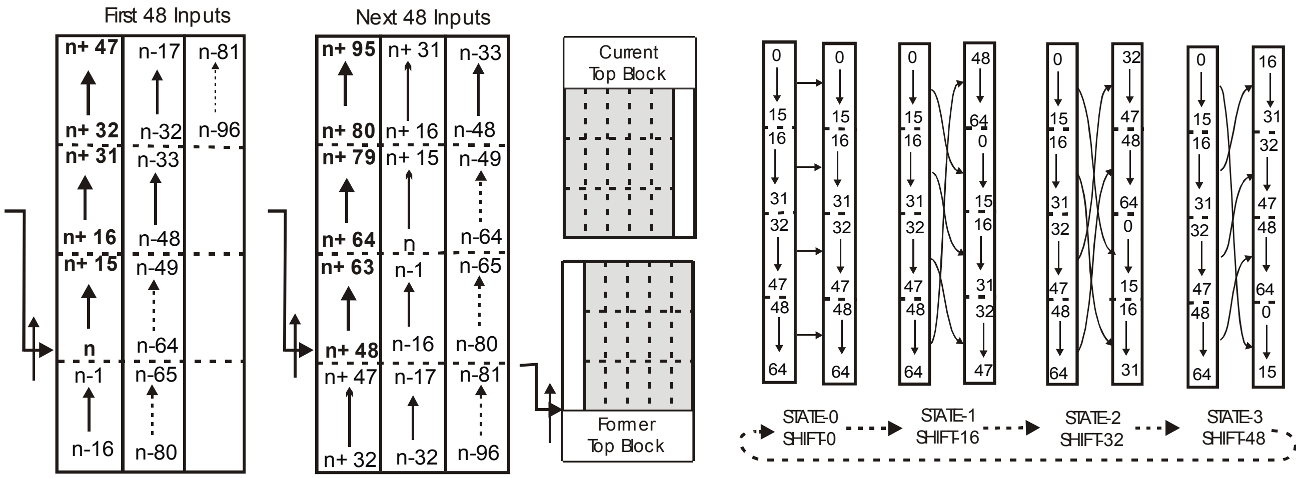

Figure 12. Data Roll of Successive 48-Point Input Blocks to 64x8 Input Buffer and Data Roll of 64-Output Paths Prior to Phase Rotators

Figure 13 presents a segment of time series and the spectrum of a composite signal containing 37 of the nominal 50 square-root Nyquist shaped channels. The center channel is intentionally vacant to simplify the task of locating it in the next figure. Figure 14 is the spectra of 48 separate time series extracted from the resampling channelizer. Here the channelizer performs the matched filtering, the spectral translation, and the non-maximally decimated resampling to 2-samples per symbol.

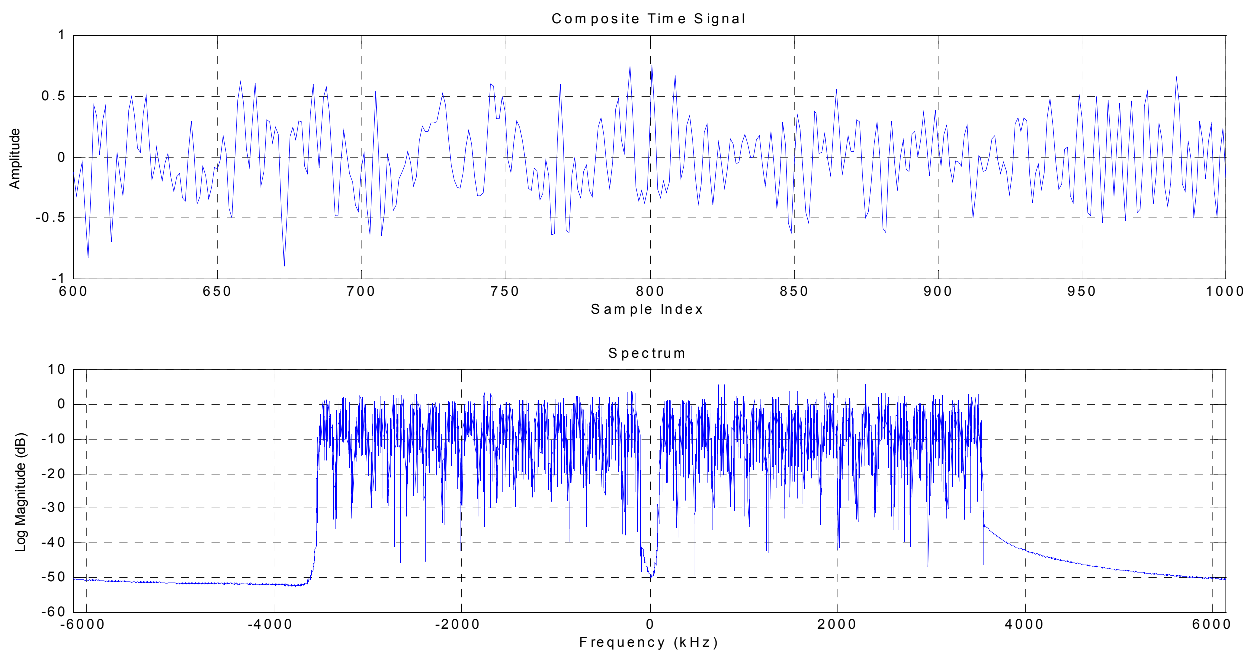

Figure 13 Time Series and Spectrum of FDM Signal with 37 128 kHz channels on 192 kHz Centers

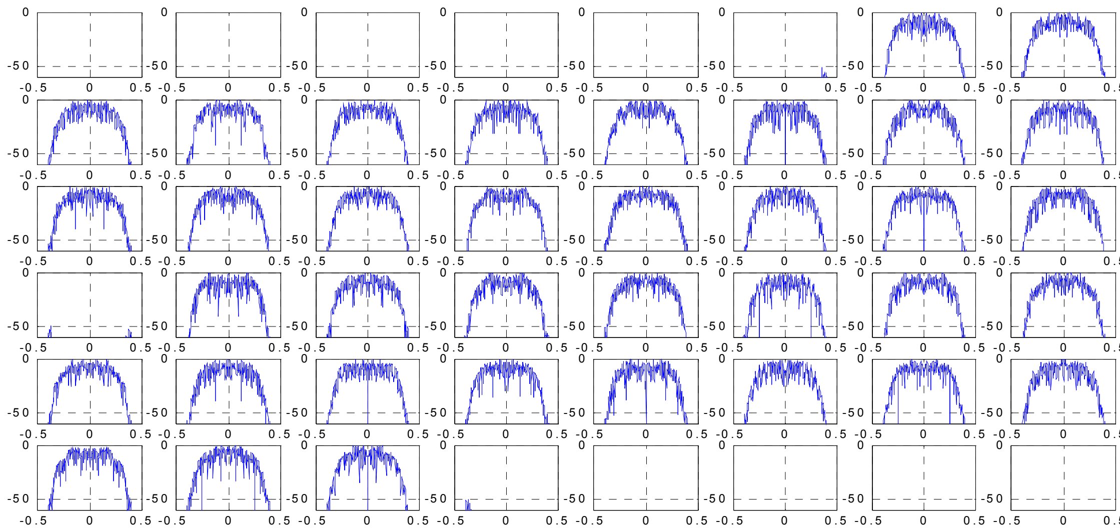

Figure 14. Spectra of Time Series Resampled to 2-Samples per Symbol in Channelized Filter Bank

CONCLUSIONS

In this paper, the second half of a pair of papers, we have further altered the digital down converter to permit IF sampling, Hilbert Transform half-band filtering with 2-to-1 down sampling, and alias based down conversion by down sampling with an M-path filter. By adding circular input and output buffers and a state machine to the M-stage filter we were able to embed a useful resampling option to form 2-samples per symbol. This option permits any arbitrary input rate can be resampled to 2-samples per symbol. We noted that the down sampling filter could also be the matched filter and that the commutator pointer can be advanced or retarded in the re-sampling operation to move the output sample time to the desired time location at the matched filter output. One last comment! The processing options presented here reduce the computational burden in a receiver, with comparable savings in a transmitter, at the expense of more complex and perhaps more interesting control mechanisms. Treat the processing task as a puzzle and enjoy the process of learning and applying clever solutions to old problems.

REFERENCES

fred harris, **"**On Multichannel receivers with Arbitrary, Uncoupled Selection of Channel Bandwidth, Channel Spacing, and Channel Sample Rate**"**, 2002 International Symposium on Advanced Radio Technologies (ISART), Mar. 4-6, 2002, Boulder Colorado, Paper posted on: `http://www.its.bldrdoc.gov/meetings/art/art02/slides02/har//har_slides.ppt`

Michael Rice and fred harris, "Multirate Digital filters for Symbol Timing Synchronization in Software Defined Radios" IEEE Journal on Selected Areas in Communications, Signal Synchronization in Digital Transmission Systems, December 2001, Vol. 19, No.12, pp. 2346-2357.

fred harris and Chis Dick, 'On Structure and Implementation of Algorithms for Carrier and Symbol Synchronization in Software Defined Radios", URSIPCO-2000 (European Association for Signal Processing), Tampere, Finland, 5-8 September 2000.