J Sign Process Syst (2011) 62:157–171
DOI 10.1007/s11265-008-0327-y

Filter Bank Channelizers for Multi-Standard Software Defined Radio Receivers

R. Mahesh · A. P. Vinod · Edmund M-K. Lai · Amos Omondi

Received: 7 October 2008 / Accepted: 25 November 2008 / Published online: 21 January 2009

Abstract The ability to support multiple channels of different communication standards, in the available bandwidth, is of importance in modern software defined radio (SDR) receivers. An SDR receiver typically employs a channelizer to extract multiple narrowband channels from the received wideband signal using digital filter banks. Since the filter bank channelizer is placed immediately after the analog-to-digital converter (ADC), it must operate at the highest sampling rate in the digital front-end of the receiver. Therefore, computationally efficient low complexity architectures are required for the implementation of the channelizer. The compatibility of the filter bank with different communication standards requires dynamic reconfigurability. The design and realization of dynamically reconfigurable, low complexity filter banks for SDR receivers is a challenging task. This paper reviews some of the existing digital filter bank designs and investigates the potential of these filter banks for channelization in multi-standard SDR receivers. We also review two low complexity, reconfigurable filter bank architectures for SDR channelizers based respectively on the frequency response masking technique and a novel coefficient decimation technique, proposed by us recently. These filter bank architectures outperform existing ones in terms of both dynamic reconfigurability and complexity.

Keywords Software defined radio · Channelization · Digital filter banks · Reconfigurability · Low complexity

R. Mahesh (✉) · A. P. Vinod
School of Computer Engineering,
Nanyang Technological University,
Singapore 639798, Singapore
e-mail: rpmahesh@ntu.edu.sg

E. M-K. Lai
School of Engineering and Advanced Technology,
Massey University,
Massey, New Zealand

A. Omondi
School of Computing, University of Teeside,
Middlesbrough, United Kingdom

1 Introduction

The wireless industry has been experiencing an exponential growth with the emergence of new radio access technologies and standards. All these technologies have been optimized to obtain a good trade-off between data rate, range and mobility to suit specific application needs. Lack of harmony in spectrum allocation globally has also resulted in this growth. However, with the increase in trade relationship between different continents, researchers had to look for a common multi-standard wireless communication platform which can support all these radio technologies and standards. This has resulted in the birth of the software defined radio (SDR) concept. SDR can be regarded as an ultimate communications solution which can ideally cover any cellular communication standard in a wide frequency spectrum with any modulation and bandwidth.

The term SDR signifies that the same hardware architecture can be programmed or reconfigured to cope with any radio standard. The major application of SDR will be in mobile communication transceivers, generic cellular base stations and military radio systems. Some of the benefits that will result with the realization of SDR are:

- easier international roaming, improved and more flexible services, increased personalization and choice for subscribers of mobile services.

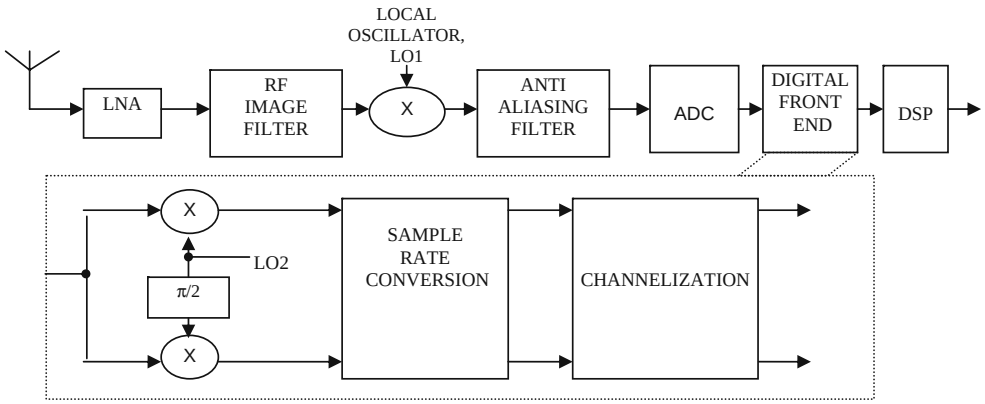


Figure 1 A feasible SDR receiver.

- the potential to rapidly develop and introduce new value-added services and revenue streams with increased flexibility of spectrum management and usage for mobile network operators.
- the promise of increased production flexibility, improved and more rapid production evolution for handset and base station manufacturers.
- the prospect of increased spectrum efficiency and better use of scarce resource for regulators.

The basic idea of SDR is to replace the conventional analog signal processing in radio transceivers by digital signal processing by placing the analog to digital converter (ADC) in receivers (digital to analog converters (DAC) in transmitters) as close to the antenna as possible. Thus SDR should be able to support multiple communication standards by dynamically reconfiguring the same hardware platform. Also, SDR should be able to use the same architecture for any number of channels by simply reconfiguring the digital front-end as compared to a conventional radio transceiver whose complexity grows linearly with the number of received channels. A feasible SDR receiver architecture is shown in Fig. 1.

A radio frequency (RF) image filter is used to remove the image frequencies that can affect the output of the mixer. The mixer is used to down convert the frequency of the input signal from RF to intermediate frequency (IF) so that modern day ADCs can easily digitize the IF signal. This is followed by an anti-aliasing filter to band-limit the input signal to prevent aliasing while sampling the signal in the analog to digital converter (ADC). The portion of the SDR which includes LNA, RF image filter, mixer and anti-aliasing filter is known as analog front-end. Thus the main functions of the analog front-end are to convert the carrier frequency from RF to IF and to bandlimit the input signal to prevent aliasing. The part of the SDR terminal where analog signal processing is replaced by digital signal processing is referred to as the digital front-end (DFE). The main functionalities of DFE are digital down conversion (frequency shifting) and channelization (filtering) as shown in Fig. 2. After, digital down conversion, the desired channel at baseband is isolated by employing lowpass filtering. A basestation receiver is often required to extract multiple channels, which is accomplished using a bank of filters. The order of digital down conversion and filtering can be interchanged. After digital down conversion and filtering, the sampling rate can be reduced. In the DFE, channelizer is the most computationally intensive part, as it comes directly after the ADC and hence needs to operate at a very high sampling rate [1, 2]. The channelizers in SDR receivers must be realized to meet the stringent specifications of low power consumption and high speed [3, 4]. In SDR receivers, channelization is usually done using digital filter banks. Uniform discrete Fourier transform (DFT) filter

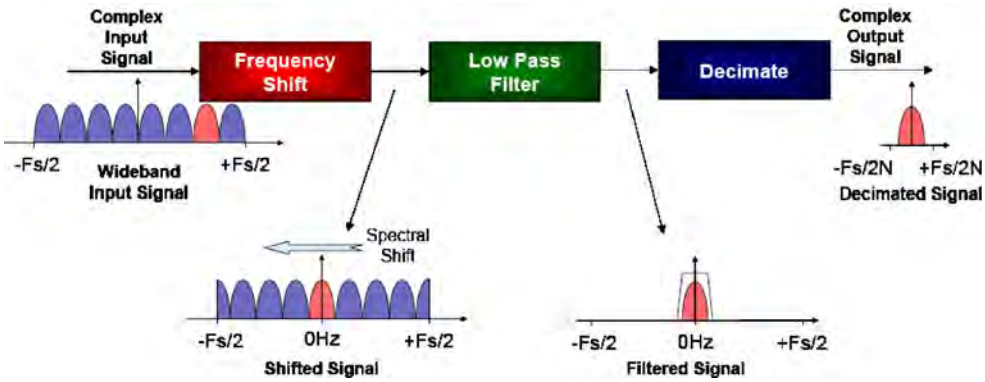


Figure 2 Typical channelization process in the digital front-end.

bank is the most commonly employed filter bank in SDR channelizer [5].

In this paper, we review some of the existing digital filter banks and investigates the potential of these filter banks for channelization in multi-standard SDR receivers. We also present two of our low complexity, reconfigurable filter bank architectures for SDR channelizers. The rest of the paper is organized as follows. In Section 2, we present the mathematical representation of SDR signals, define some important characteristics for SDR filter banks and review various filter bank architectures. In Section 3, we present a reconfigurable frequency response masking filter based channelizer. A novel coefficient decimation filter bank architecture is presented in Section 4. In Section 5, we have compared all the filter bank techniques reviewed in this paper. Section 6 provides conclusions.

2 SDR Signal Formulation and Review of Filter Banks

2.1 Signal Formulation

An SDR can be regarded as a system which should be able to modulate or demodulate any kind of signal, anywhere, on any network. In this context, it is clear that an SDR signal is a composite signal. It is given by the equation [6]:

$$x(t) = \sum_{i=1}^{S} S_i(t) \tag{1}$$

where $S_i(t)$ represents the i^{th} standard signal and S, the number of standards contained in the composite signal. The generic equation of a standard, for channels spread on a plurality of carriers, gives:

$$S_i(t) = \sum_{p=1}^{P_i} r_{i,p}(t) e^{2j\pi f_{i,p} t} \tag{2}$$

where $r_{i,p}(t)$ represents the modulated and filtered signal associated to the carrier p of the standard i, which is at the centre frequency, f_i. Thus finally we have the equation:

$$x(t) = \sum_{i=1}^{S} \sum_{p=1}^{P_i} r_{i,p}(t) e^{2j\pi f_{i,p} t} \tag{3}$$

with $r_{i,p}(t) = fem_i(t) * m_{i,p}(c(t))$, where $m_{i,p}(t)$ represents the modulation relative to the carrier p and $fem_i(t)$ the pulse shaping filter function. In conclusion, a multi-standard SDR signal can be represented as

$$x(t) = \sum_{i=1}^{S} \sum_{p=1}^{P_i} (fem_i(t) * m_{i,p}(c(t))) e^{2j\pi f_{i,p} t} \tag{4}$$

In (4), the terms $fem_i(t)$ and $m_{i,p}(t)$ are standard dependent and varies from one standard to another. Thus $x(t)$ in (4) can be considered as a combination of channels with varying factors such as bandwidth. Thus it is obvious from (4) that, for an SDR system, we require a filter bank which can adapt to any standard dynamically and efficiently.

Based on (4) and the discussion so far, we define the following essential requirements for an SDR receiver:

1. The receiver must be capable of extracting channels corresponding to different communication standards which are independent of each other. This means specifically that the receiver must be capable of extracting non-uniform bandwidth channels as bandwidths of different standards are different.
2. The receiver must be capable of extracting narrowband channels from the wideband input signal.
3. In an ultimate SDR, reconfigurability of the receiver must be accomplished by reconfiguring the same filter bank for a new communication standard with minimal reconfiguration overhead, instead of employing separate filter banks for each standard. In the sequel, we call this requirement as *ultimate reconfigurability.*

2.2 Filter Banks for SDR Channelizers

In this section, we review existing digital filter banks and discuss their suitability for channelization in SDR receivers.

A. Per-Channel (PC) Approach

The PC approach is based on a parallel arrangement of many one-channel channelizers. Each one-channel channelizer performs the channelization process shown in Fig. 2. The basic architecture of the PC approach is shown in Fig. 3.

In Fig. 3, the order of channelization is filtering ($H_0(z)$ to $H_n(z)$), digital down conversion (DDC), sample rate conversion (SRC) and finally baseband processing (BBP). The filter, $H_0(z)$, is a low pass filter and all other filters,

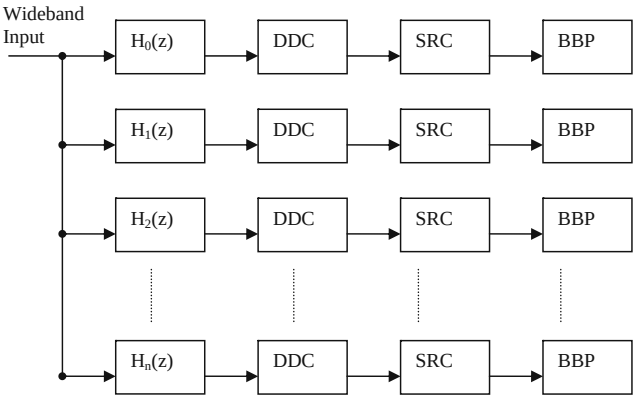


Figure 3 Per-Channel approach based channelization.

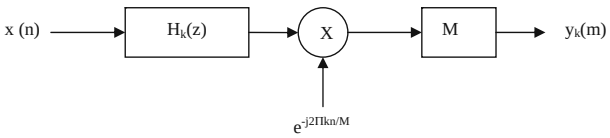
$H_1(z)$ to $H_n(z)$), are bandpass filters. It is possible to do digital down conversion followed by filtering and consequently, all the filters will be low pass filters (all filters are $H_0(z)$). It is also possible to further reduce the complexity of the PC approach by employing polyphase decomposition of each of the filters and then moving the SRC to the left of filtering operation (i.e., performing SRC before filtering). By employing polyphase decomposition, the speed of filtering operation can be relaxed.

The PC approach is a straightforward approach and hence relatively simple. But the main drawback is that, the number of branches of filtering-DDC-SRC is directly proportional to the number of received channels. Hence the PC approach is not efficient when the number of received channels is large. Furthermore, if the channels are of uniform bandwidth, a filter bank approach would be a cost-effective solution than the PC approach. In conclusion, although all the three requirements in an SDR receiver listed in Section 2.1 can be met using the PC approach, its hardware cost is very high, which has led to the development of DFT filter banks.

B. DFT Filter Banks

DFT filter bank is a uniformly modulated filter bank, which has been developed as an efficient substitute for PC approach when the number of channels need to be extracted is more, and the channels are of uniform bandwidth (for example many single-standard communication channels need to be extracted). The main advantage of DFT filter bank is that, it can efficiently utilize the polyphase decomposition of filters. The derivation of DFT filter bank from PC approach can be explained with reference to Fig. 3 [7]. Consider the k^{th} channelization branch as shown in Fig. 4.

Figure 4a shows the whole process of bandpass filtering followed by DDC and SRC (downsampling by M). Note that the only modulator outputs not discarded by the SRC are those with time index $n = mM$. For these outputs, the modulator has the value $e^{-j2\Pi kn/M} = 1$, and thus it can be ignored. The resultant figure eliminating DDC is as shown in Fig. 4b. Now it is possible to expand $H_k(z)$ in terms of M polyphase branches as shown in Fig. 5 and it is possible to



(a) Down conversion to baseband after bandpass filtering.

(b) Modified down conversion to baseband after bandpass filtering.

Figure 4 **a** Down conversion to baseband after bandpass filtering. **b** Modified down conversion to baseband after bandpass filtering.

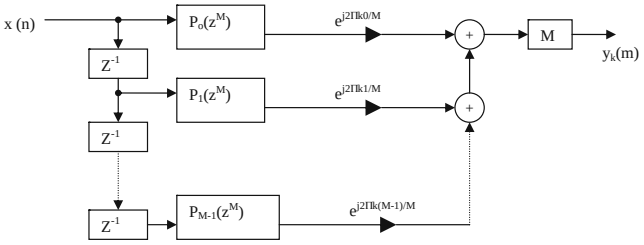


Figure 5 k^{th} filter bank branch containing M polyphase branches.

move the down sampling by M to the left of filtering operation. This can be explained with the help of following equations:

$$H_k(z) = \sum_{m=-\infty}^{\infty} h_k(n)z^{-n} = \sum_{l=0}^{M-1} z^{-l} \sum_{m=-\infty}^{\infty} h_k(mM+l)z^{-mM} \tag{5}$$

where $h_k(mM+l)$ represents the polyphase components of $H_k(z)$. Now expanding h_k in terms of the lowpass filter coefficient, h, i. e. substituting $h_k(mM+l) = h(mM+l)e^{j\frac{2\pi k(mM+l)}{M}}$ (5) becomes,

$$\begin{aligned} H_k(z) &= \sum_{l=0}^{M-1} z^{-l} \sum_{m=-\infty}^{\infty} \left(h(mM+l)e^{j\frac{2\pi k(mM+l)}{M}} \right) z^{-mM} \\ H_k(z) &= \sum_{l=0}^{M-1} z^{-l} \sum_{m=-\infty}^{\infty} \left(h(mM+l)e^{j\frac{2\pi kl}{M}} \right) z^{-mM} \\ H_k(z) &= \sum_{l=0}^{M-1} z^{-l} \left(\sum_{m=-\infty}^{\infty} h(mM+l)z^{-mM} \right) e^{j\frac{2\pi kl}{M}} \end{aligned} \tag{6}$$

Now replacing, $\sum_{m=-\infty}^{\infty} h(mM+l)z^{-mM}$ by $P_l(z^M)$, (6) becomes,

$$H_k(z) = \sum_{l=0}^{M-1} z^{-l} P_l(z^M) e^{j\frac{2\pi kl}{M}} \tag{7}$$

The expression (7) is shown in Fig. 5. In Fig. 5, $P_0(z)$ to $P_{M-1}(z)$ represent the polyphase components of a lowpass filter. The modified version of Fig. 5 by making use of the noble identity is shown in Fig. 6. It can be seen that, the dotted portion in Fig. 6 represents the inverse DFT (IDFT) operation and hence can be replaced by IDFT. By employ-

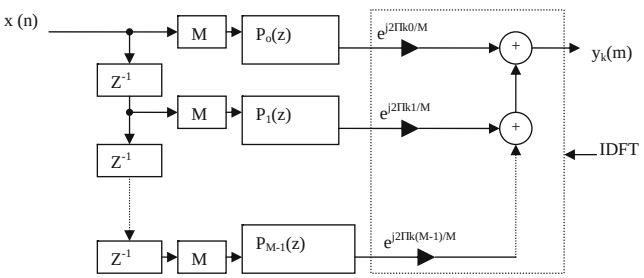


Figure 6 Modified k^{th} filter bank branch containing M polyphase branches.

ing IDFT, we get all the channels (frequency bands) simultaneously in a DFT filter bank as shown in Fig. 7. Efficient implementations of a channelizer using DFT filter banks (DFTFBs) are available in literature [5].

It can be seen from Fig. 7 that DFTFB can be realized by implementing one low-pass filter and a corresponding modulator (IDFT). Thus instead of implementing N separate channel filters as in the case of PC approach, a single lowpass filter followed by DFT (complexity of IDFT is equivalent to that of DFT) is only required. However, DFTFBs have following limitations for multi-standard receiver applications [5]:

1. DFTFBs cannot extract channels with different bandwidths. This is because DFTFBs are modulated filter banks with equal bandwidth of all bandpass filters. Therefore, for multi-standard receivers, distinct DFTFBs are required for each standard. Hence the complexity of a DFTFB increases linearly with the number of received standards.
2. Due to fixed channel stacking, the channels must be properly located for selecting them with the DFTFB. The channel stacking of a particular standard depends on the sample rate and the DFT size. To use the same DFTFB for another standard, the sample rate at the input of the DFTFB must be adapted accordingly. This requires additional sample rate converters (SRCs), which would increase the complexity and cost of DFTFBs.
3. If the channel bandwidth is very small compared to wideband input signal (extremely narrowband channels), the prototype filter must be highly selective resulting in very high-order filter. As the order of the filter increases, the complexity increases linearly. Also the DFT size needs to be increased.

Reconfigurability is another key requirement as we had mentioned earlier. Ideally, the reconfigurability of the filter bank must be accomplished by reconfiguring the same prototype filter in the filter bank to process the signals of the new communication standard with the least possible overhead, instead of employing separate filter banks for each standard. However reconfiguration of DFTFB suffers from following overheads:

1. The prototype filter needs to be reconfigured. Generally DFTFB employs the polyphase decomposition. Hence reconfiguration can involve changing the number of polyphase branches which is a tedious and expensive task.
2. Downsampling factor needs to be changed. If down sampling is to be done after filtering, then we need separate digital down converters. This will add more cost.
3. The DFT needs to be reformulated accordingly which is also expensive.

For example, if we are switching from a 8-channel filter bank to 16-channel filter bank, the number of polyphase branches need to be changed from 8 to 16 (first limitation of DFTFB), the downsampling factor needs to be adjusted from 8 to 16 (second limitation of DFTFB) and the 8-point DFT needs to be expanded to 16-point DFT.

In the case of multiple channel extraction of single-standard signal i. e., extraction of many channels of identical bandwidth, the complexity of PC approach is given by $N.L.f_s$, where N is the number of channels extracted, L is the total length of filters employed in all the branches for PC approach and f_s is the sampling frequency. The complexity of DFTFB is only $L.f_s$ which is N times lower than PC approach. But in the case of SDR, multiple channels of multiple standards need to be extracted (extraction of multiple channels of non-uniform bandwidths). In that case, the complexity of PC approach and DFTFB are N_{CSs} and N_{Ss} respectively, where N_C and N_S are the number of channels and number of standards respectively. Thus the complexity of these channelizers can be reduced further if (1) the length of filter, L, can be reduced and (2) the same filter bank can be reconfigured to the new standard (which will eliminate the dependency on the term N_S from complexity equation). Thus there is a need for developing new filter bank architectures for SDR receivers. In the next section, we discuss the approaches presented in literature to minimize the hardware complexity of filters and filter banks in the channelizer of an SDR.

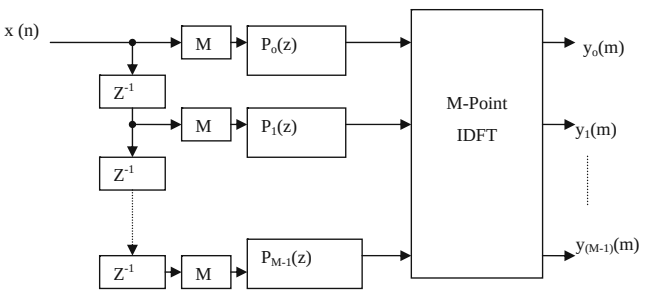


Figure 7 DFT filter bank.

C. Alternative Filter Banks

A Goertzel filter bank (GFB) based on modified Goertzel algorithm was proposed in [5] as a substitute to DFTFB. In GFB, the DFT is replaced by a modified Goertzel algorithm which performs the modulation of the prototype low-pass frequency response to any centre frequency which is not possible using DFT. This will eliminate the limitation of fixed channel stacking associated with DFTFBs. But the GFB is also a type of modulated

filter bank; hence it cannot extract channels with different bandwidths, as in the case of DFTFB. Also, extraction of narrow-band channels using GFB requires a very narrow passband prototype filter, which would in turn result in higher order filter. The GFB approach requires IIR filter for the implementation of Goertzel algorithm and hence has stability constraints while reconfiguring the filter bank from one communication standard to another. Even though a theoretical introduction of GFB as a solution to some of the problems of DFTFB is given in [5], there is no consideration on the actual implementation complexities of GFB.

A channelizer based on a combination of polyphase filter bank and modified DFT (MDFT) modules have been proposed in [8]. The MDFT module performs real signal calculations instead of complex signal calculations and thus reduces computational complexity associated with the DFT operation. This is achieved by taking the real part of the DFT for the complex values. The MDFT module consists of one adder and two K-tap FIR filters, where K represents the number of polyphase branches of the prototype filter. Thus the over-all computational complexity of the filter bank is reduced when compared to conventional DFTFBs. However the channelizer in [8] is less flexible when compared to DFTFB. This is because the coefficients of the FIR filters in the MDFT module are dependent on the polyphase prototype filter. The reconfigurability of same filter bank for a new communication standard is also not achieved by the method in [8].

A multi-standard channelizer that has two stages of DFTFBs and efficient sample rate converters has been proposed in [9]. The front-end DFTFB has fixed number of channels, but the passband supports overlap with each other considerably resulting in easier isolation of channels with center frequencies of successive band-pass filters. The outputs of the front-end DFTFB are then fractionally decimated using SRCs. These decimated outputs are fed to the back-end DFTFB. Since the sample rate is considerably lowered in the back-end, the DFTFB at the back-end need to operate only at low-speed. Due to the reduced speed requirements, the back-end DFT can be repeatedly used to extract variable bandwidth channels. The drawback of the architecture in [9] is that, since the back-end DFTFB is employed for varying bandwidth channels, hardware optimization can be done only for the fixed front-end DFTFB. The back-end needs to be changed according to the new communication standard.

In [10], a channelizer based on modulated perfect reconstruction bank (MPRB) has been proposed. The MPRB consists of an analysis section and a synthesis section. By adding up the subband signals generated by the analysis section, wideband signals can be generated at the synthesis section. Thus the approach in [10] can be used for the channelization of signals of unequal bandwidths. However, the bandwidths of the wideband signals generated by the synthesis section are integer multiples of the bandwidths of the subband signals generated by the analysis section. Thus the approach in [10] is not always appropriate for SDR signals, where the multiple communication standards have bandwidths which are not integer multiples of each other. A new method for the efficient design of the MPRB is also proposed in [10]. Also the approach in [10] consists of a polyphase prototype filter, IDFT analysis section and DFT synthesis section. Thus the computational complexity of the MPRB is double that of DFTFB. Also the implementation complexities associated with MPRB have not been considered in [10].

A pipelined frequency transform (PFT) based on the PC approach has been proposed in [11]. The basic PFT architecture consists of a binary tree of DDCs and SRCs, which splits the input signal frequency into a low and high frequency subbands, and then splits each half-band again until the last tree level extracts the desired channels. The PFT architecture consisting of binary tree of DDCs and SRCs is shown in Fig. 8, where DDCs followed by SRCs are employed for dividing the input signal into a low-pass and high-pass bands with half sampling rate at the output. The main advantage of PFT approach over PC approach is that, the complexity of filtering can be reduced substantially taking advantage of half-band symmetry and reduced sampling rate at each output stage.

A reconfigurable channelizer using tree-structured quadrature mirror filter bank (TQMFB) has been proposed in [12]. The TQMFB consisted of a tree of quadrature mirror filter banks, splitting the frequency band of input signal into high and low frequencies at quadrature frequency in each stage. Thus the TQMF approach in [12] is very similar to the PFT approach in [11]. The desired channel is obtained at an appropriate stage corresponding to the bandwidth of the channel-of-interest. The main drawback of the TQMFB is its delay in obtaining the desired output due to multistage filtering and decimation processes. The channelizers in [11] and [12] suffer from the drawback that they can only extract signals whose channel spacing are related by a factor-of-two. This constraint is imposed by the power-of-two

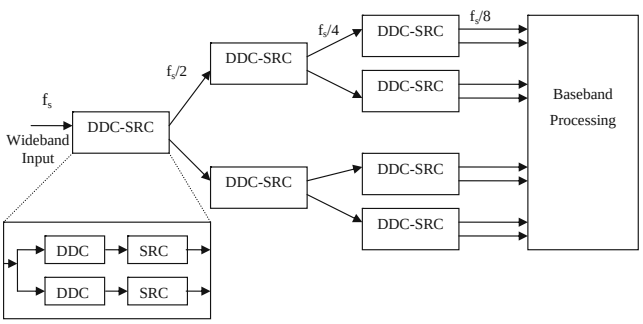


Figure 8 Architecture of PFT approach.

subband stacking adopted in these architectures. Another problem of the methods in [11, 12] is that, as the subband decomposition tree extends, the wordlength of the output increases linearly and finite wordlength multiplication would introduce truncation error, which propagates along the tree. In the PFT approach, the problem with power-of-two subband stacking can be overcome by a tunable PFT (TPFT) architecture [13]. In the TPFT architecture, interleavers are introduced between different stages of PFT, which will enable the usage of intermediate outputs from different stages along the binary tree. These interleavers will help in fine tuning of channelization process and thus add more flexibility to the PFT architecture. Thus in TPFT, two levels of tuning are done, a coarse tuning at the PFT level and a fine tuning using another complex up/down converter assisted by a numerical controlled oscillator. However the implementation complexity of TPFT approach is much more than that of the PFT approach and thus not a very good candidate for wideband channelization.

A qualitative comparison of different channelization approaches is given in Table 1. The PC approach is compared to DFTFBs and PFT approaches based on four parameters. The TPFT approach is not suitable for SDR, whereas the approaches in [5, 8–10] are modifications of DFTFB. In Table 1, the parameter 'computational complexity' represents the number of multiplications associated with each method. Previous works [5, 14, 15] showed that when the number of uniform bandwidth channels to be extracted is more than two, the DFTFBs outperform the PC approach. It is also shown in [14] that an improvement in the filters of the PC approach can make it more efficient up to extraction of 20 channels in some scenarios. The computational complexity of PFT method is less than the PC approach, but not lower than that of DFTFB. The parameter 'silicon cost' shows the actual implementation cost in FPGA. A drawback of this parameter is that it is platform dependent. It is shown in [11] that up to 256 channels, the silicon cost of PFT approach is comparable to DFTFB, but beyond 256 channels, DFTFB outperforms PFT approach. The third parameter is the 'initial design flexibility' which involves a combination of two factors: 1) ability to extract non-uniform bandwidth channels, and 2) the number of channels extracted. When 'initial design flexibility' is considered, the PC approach is obviously the best as all the extracted channels are independent, can have different bandwidths and can be non-uniformly and discontinuously distributed over the input frequency band. Neither PFT nor DFTFB is able to extract independent channels. The PFT has the limitation of power-of-two subband stacking and hence the number of extracted channels will be in powers of two. Even though DFTFB has more flexibility, the most economical implementation of DFT has integer power of two bins and hence the number of extracted channels can be very similar to the PFT approach. On the other hand, the PC approach has no such problems. The parameter 'reconfigurability' represents the adaptation of channelization architecture to satisfy the new requirements with minimum overhead. As discussed in the previous sections, none of the existing approaches satisfy the reconfigurability requirement. It can be noted from Table 1 that the existing channelization approaches do not offer an efficient trade-off between complexity and reconfigurability. In the next two sections, we present two filter banks called frequency response masking (FRM) based filter bank and coefficient decimation (CD) based filter bank, which can offer a good trade-off between reconfigurability and low complexity and satisfy most of the requirements for SDR receivers.

3 Frequency Response Masking Based Filter Banks

Ideally, the reconfigurability of the receiver must be accomplished in such a way that the filter bank architecture serving the current communications standard must be reconfigured with least possible overhead to support a new communication standard while maintaining the parallel operation (simultaneous reception/transmission of multi-standard channels). To realize a filter bank which can be reconfigured to accommodate multiple standards with reduced hardware overhead, we have proposed a frequency response masking (FRM) based reconfigurable filter bank (FRMFB) in [16, 20]. The FRM technique was originally proposed for designing application-specific low complexity sharp transition-band finite impulse response (FIR) filters (fixed-coefficient filters) [17]. We have modified the

Table 1 Comparison of channelization approaches.

Parameter		PC Approach	DFTFB	PFT
Computational complexity	For multiple uniform bandwidth channels	Poor	Excellent	Good
	For multiple non-uniform bandwidth channels	Poor	Poor	Poor
Silicon cost		Poor	Excellent	Good
Initial design flexibility	Independent channels	Yes	No	No
	Number of channels	Selectable	2^N	2^N
Reconfigurability		Poor	Very poor	Poor

original FRM approach in [17] to achieve following advantages: (1) Incorporate reconfigurability at the filter level and architectural level, (2) Improve the speed of filtering operation and (3) Reduce the complexity.

A. Review of FRM Technique

Finite impulse response (FIR) filters are widely employed in wireless communication systems because of its linear phase property and guaranteed stability. Sharp transition-band FIR filters are required in these systems to meet the stringent wireless communication specifications. In conventional FIR filter designs, higher order filters are required to obtain sharp transition-band. The complexity of FIR filters increases with the filter order. Several approaches have been proposed for reducing the complexity of FIR filters. In [17], an FRM technique was employed for the synthesis of sharp transition-band FIR filters with low complexity. The advantage of FRM technique is that, the bandwidths of the filters are not altered and the resulting filter will have many sparse coefficients (because the subfilters have wide transition-band) resulting in less complex filters. The basic idea behind the FRM technique is to compose the overall sharp transition-band filter using several wide transition-band subfilters. We proposed to use the sharp transition-band filter designed using FRM with necessary modifications as filter bank in a CR system [16]. Given a prototype symmetrical impulse response linear phase low-pass filter (known as 'modal filter') $H_a(z)$ of odd length N_a, its complementary filter $H_c(z)$ can be expressed as

$$H_c(z) = z^{-M\frac{(N_a-1)}{2}} - H_a(z) \tag{8}$$

Replacing each delay elements of both filters by M delays, two filters with transfer functions $H_a(z^M)$ and $H_c(z^M)$ are formed. The transition widths of $H_a(z^M)$ and $H_c(z^M)$ are a factor of M narrower than that of $H_a(z)$. In the FRM technique, two filters $H_{Ma}(z)$ and $H_{MC}(z)$, are cascaded to $H_a(z^M)$ and $H_c(z^M)$, respectively as shown in Fig. 9. The transfer function of the entire filter is given by

$$H(z) = H_a(z^M)H_{Ma}(z) + H_c(z^M)H_{Mc}(z) \tag{9}$$

Note that the group delay of the filters $H_{Ma}(z)$ and $H_{Mc}(z)$ must be equal, and $M(N_a-1)$ in the Eq. (8) must be an even number. The design steps for the subfilters in Fig. 9 (i. e the passband and stopband specifications of the subfilters) involve the solution of the expressions [17]:

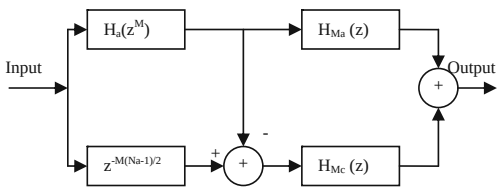


Figure 9 FIR filter architecture based on FRM technique.

$$m = \lfloor f_p M \rfloor \tag{10(a)}$$

$$f_{ap} = f_p M - m \tag{10(b)}$$

$$f_{as} = f_s M - m \tag{10(c)}$$

$$f_{map} = f_p \tag{10(d)}$$

$$f_{mas} = \frac{m + 1 - f_{as}}{M} \tag{10(e)}$$

$$f_{mcp} = \frac{m - f_{ap}}{M} \tag{10(f)}$$

$$f_{mcs} = f_s \tag{10(g)}$$

where m denotes the largest integer less than $\lfloor f_p M \rfloor$, M is the up-sampling rate for H_{ap} and f_s are the passband and stopband edges of the overall filter, f_{ap} and f_{as} are the passband and stopband edges of the modal filter $H_a(z)$, f_{map} and f_{mcp} are the passband edges and f_{mas} and f_{mcs} are the stopband edges of the two masking filters respectively. All the stopband and passband edges mentioned in this paper including expressions (10 (a)–(g)) are normalized to unity. Thus by suitable selection of the passband and stopband edges of the modal and the masking filters, any sharp transition-band FIR filter can be implemented with low complexity [17].

The FRM approach can be illustrated with the help of Fig. 10. Figure 10a represents the frequency response of a low-pass filter $H_a(z)$. The passband and stopband edges of the modal filter are f_{ap} and f_{as} respectively. The complementary filter of the modal filter, $H_c(z)$, is shown in Fig. 10b. Replacing each delay of $H_{ac}(z)$ by M delays, two filters $H_a(z^{MM})$ are obtained, and their frequency responses are shown in Fig. 10c. Two masking filters $H_{Ma}(z)$ and $H_{Mc}(z)$ as shown in Fig. 10d, are used to mask $H_a(z^{MM})$ respectively. If the outputs of $H_{Ma}(z)$ and $H_{Mc}(z)$, are added, as shown in Fig. 9, the frequency response of the resulting filter, $H(z)$, is shown in Fig. 10e. Thus a sharp transition-band FIR filter is obtained using four subfilters. Since these subfilters are having wide transition-band specifications, the overall complexity will be much less than direct or conventional design of sharp transition-band FIR filters.

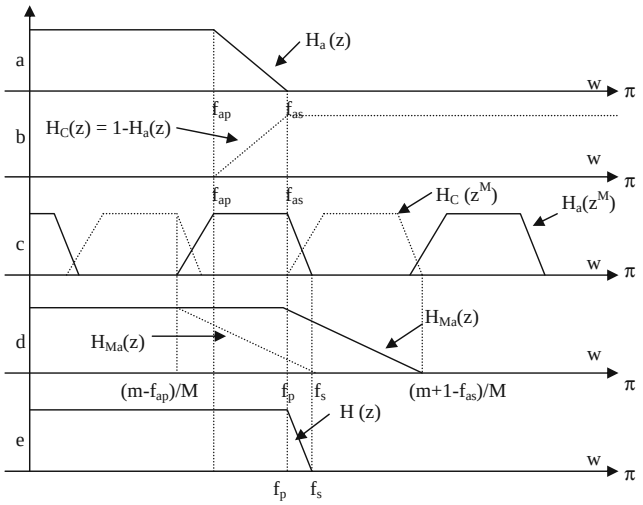


Figure 10 Frequency response illustration of FRM approach.

B. Reconfigurable FRM Filter Bank

In this section, we present a brief overview of the filter bank proposed by us in [16]. The reconfigurability of the proposed filter bank can be illustrated using the expressions given by (11, 12). Though the proposed architecture is capable of handling multiple modes of communication, for ease of explanation, we use a dual-mode operation to illustrate reconfigurability. Let f_{p1} and f_{p2} be the passband frequencies and f_{s1} and f_{s2} be the stopband frequencies of the channel filters corresponding to two modes of operation, m_1 and m_2 respectively. Reconfigurability can be achieved by using the same subfilters in Fig. 9 for both modes (communication standards). The parameters f_{ap} and f_{as} remain unchanged for both the modes and therefore, the same modal filter is employed for both modes. The masking filters can be selected according to the desired specifications which will be explained in detail later. Thus we have,

$$f_{p1}M_1 - \lfloor f_{p1}M_1 \rfloor = f_{p2}M_2 - \lfloor f_{p2}M_2 \rfloor \tag{11}$$

$$f_{s1}M_1 - \lfloor f_{s1}M_1 \rfloor = f_{s2}M_2 - \lfloor f_{s2}M_2 \rfloor \tag{12}$$

where M_1 and M_2 denote the up-sampling factors for the two modes m_1 and m_2 respectively, which can be obtained by solving (11) and (12). Thus by changing the number of delay elements, we can employ the same modal filter to work for both the modes. The dual-mode filter bank can be easily extended to incorporate additional communication modes by choosing an appropriate number of delay elements. For example, if the architecture needs to be modified to include a third communication mode, m_3, the number of delays or the up- sampling factor (M_3) can be

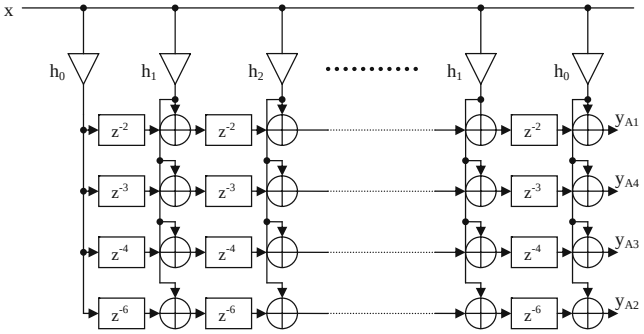


Figure 11 Architecture of modal filter for mode-4 filter bank.

obtained by substituting the filter specification corresponding to the third mode into expressions (11) and (12). The architecture of such a modal filter is shown in Fig. 11. In Fig. 11, mode-4 operation is shown.

The filters operating for all the communication modes share the same coefficient multiplication which results in good savings in area and power. The outputs of modal filter y_{A1} to y_{A4} are multiple frequency bands as shown in Fig. 10c. The complementary outputs corresponding to each of these modal filter outputs can be obtained by using complementary delays as shown in Fig. 9 and the number of delays needed for obtaining the complementary output is given by

$$(N-1)M/2 \tag{13}$$

where N is the length of the modal filter and M is the number of delays. Thus for obtaining four modes simultaneously, four set of complementary delays are required. Similarly for each mode, two masking filters are also required as shown in Fig. 9. A generalized architecture for the proposed filter bank for extracting n channels is shown in Fig. 12.

In Fig. 12, *FMA* and *FMC* are masking filters for the modal filter and complementary delay outputs respectively. Since all the filters (modal filter and masking filters, *FMA* and *FMC*) employed in the proposed filter bank are wide

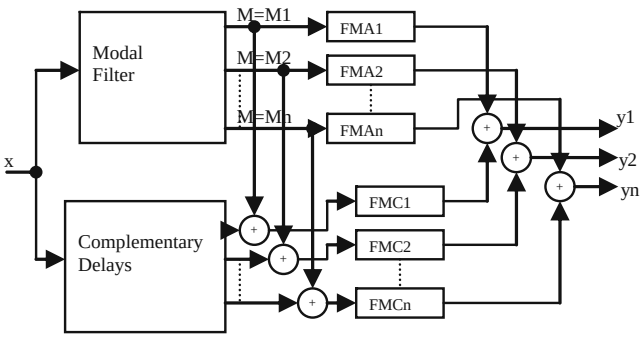


Figure 12 Architecture of proposed mode-n FRMFB.

transition-band filters as shown in Fig. 10, their lengths are short and consequently their complexities are low. The FRMFB offers reconfigurability at two levels—architectural level and filter level. At the architectural level, it is possible to obtain different frequency bands (channels) by changing the number of delays as shown in Fig. 12. By masking out the undesired frequency bands, it is possible to obtain the desired frequency output as shown in Fig. 10. At the filter level, it is possible to reconfigure the filter coefficients to obtain an entirely new frequency specification [18]. The number of coefficients to be reconfigured for the proposed filter bank will be much less compared to the conventional filter bank implementations. This is because all the subfilters (masking filters) in the proposed FRM filter bank are having wide transition band and hence lower order filters. Note that the modal filter remains unchanged for extracting all the channels (fixed-coefficient model filter) and only the masking filters and the complementary delays need to be changed based on the channel to be extracted. Thus by selecting the appropriate output response of modal filter and suitable masking filters, it is possible to obtain any frequency band at the output of the filter bank. If DFTFB were employed for channelization, an extremely higher order filter of around 8 times that of the total order of all the filters in the proposed FRMFB would be required.

Thus by employing reconfigurabilities at architectural and filter levels, FRMFB is capable of adapting to any signal with reduced hardware overhead. For different values of M, the passband widths are different; hence non-uniform channel extraction is possible in the FRMFB. Also FRMFB does not employ any modulator such as DFT and is free from the problem of fixed channel stacking.

4 Coefficient Decimation Based Filter Bank

The disadvantage of FRMFB is that it does not have full flexibility of selection of passband widths and location of centre frequencies of passbands. Recently, we have proposed a novel coefficient decimation (CD) approach for implementing computationally efficient reconfigurable filter banks for SDR receivers [19]. The filter bank based on the CD approach have absolute control over the passband width and passband locations i. e., center frequencies of passbands, when compared to other filter banks in literature. The principle of CD approach is as follows: If the coefficients of an FIR filter are decimated by M, i.e., every M^{th} coefficient is kept unchanged and all others are replaced by zeros, we get a frequency response similar to images created during upsampling (Our definition of coefficient decimation is that unused coefficients are replaced by zeros as opposed to the conventional notion of discarding unused samples in the decimation of a signal). This can be explained theoretically as follows:

Let $h(n)$ be the original set of coefficients. If we replace all the coefficients other than every M^{th} coefficient by zeros,

$$h'(n) = h(n)c_M(n) \tag{14}$$

where,

$$c_M(n) = 1 \; for \; n = mM;\; m = 0, 1, 2 \text{ etc.} = 0 \text{ otherwise} \tag{15}$$

The function $c_M(n)$ is periodic with period M, and hence the Fourier series expansion is given by

$$c_M(n) = \frac{1}{M}\sum_{k=0}^{M-1} C(k)e^{\frac{j2\pi kn}{M}} \tag{16}$$

where $C(k)$ are complex-valued Fourier series coefficients defined by

$$C(k) = \sum_{n=0}^{M-1} c_M(n)e^{\frac{-j2\pi kn}{M}} \tag{17}$$

Substituting (15) into (17) it follows that $C(k)=1$ for all k. Hence,

$$c_M(n) = \frac{1}{M}\sum_{k=0}^{M-1} e^{\frac{j2\pi kn}{M}} \tag{18}$$

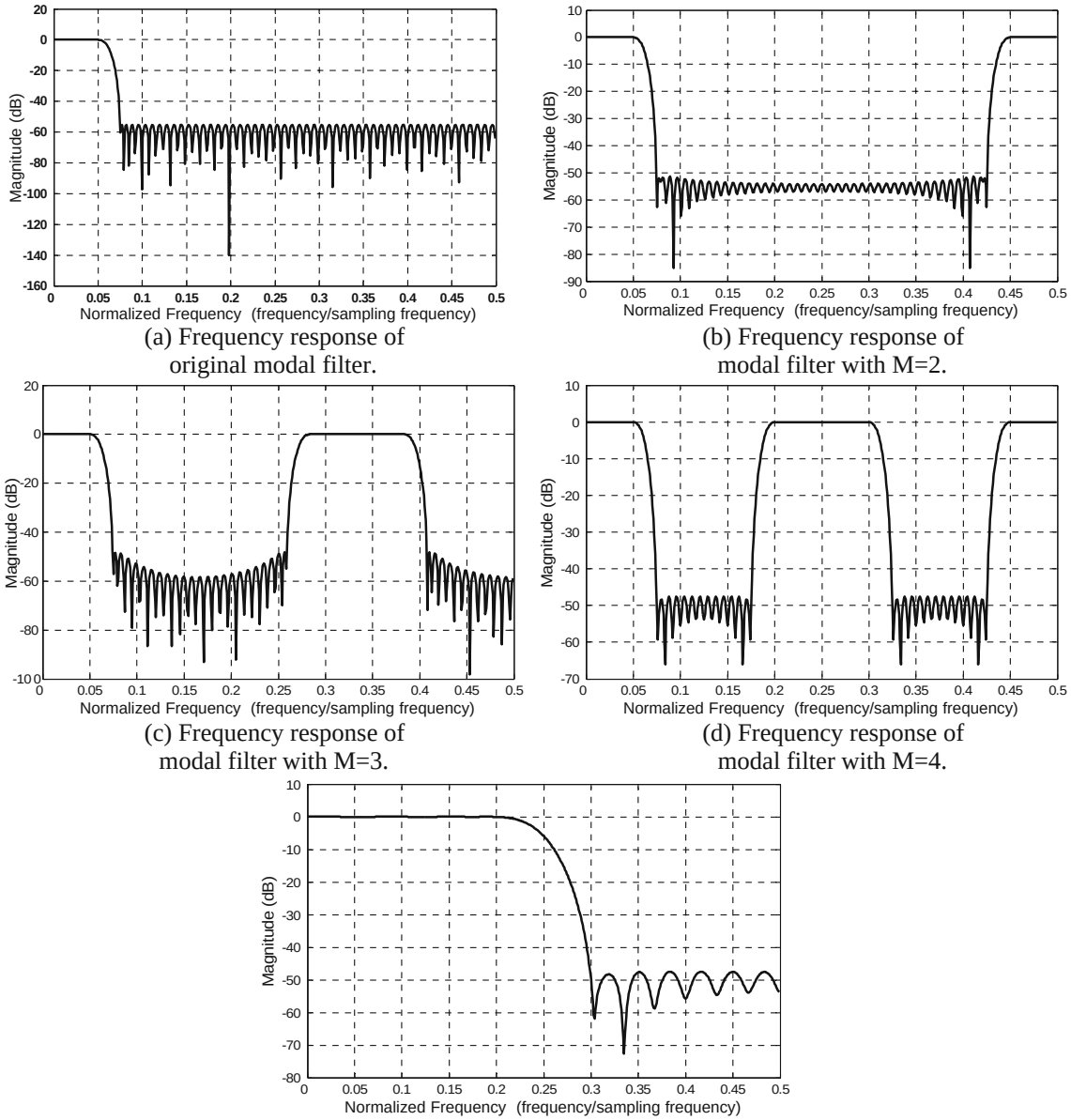
Now the Fourier transform of the modified coefficients, $h(n)$,

$$\begin{aligned} H'(e^{jw}) &= \sum_{n=-\infty}^{\infty} h'(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty} h(n)c_M(n)e^{-j\omega n} \\ &= \sum_{n=-\infty}^{\infty} h(n)\left(\frac{1}{M}\sum_{k=0}^{M-1} e^{\frac{j2\pi kn}{M}}\right)e^{-j\omega n} \end{aligned} \tag{19}$$

Finally by interchanging the sums in (19)

$$\begin{aligned} H'(e^{jw}) &= \tfrac{1}{M}\sum_{k=0}^{M-1}\sum_{n=-\infty}^{\infty} h(n)e^{-jn\left(\omega-\frac{2\pi k}{M}\right)} \\ &= \tfrac{1}{M}\sum_{k=0}^{M-1} H\left(e^{j\left(\omega-\frac{2\pi k}{M}\right)}\right) \end{aligned} \tag{20}$$

It can be noted from (20) that, the frequency response is scaled by M and the replicas of the frequency spectrum are introduced at integer multiples of $2\pi/M$. Thus in order to recover the original signal, the output of the filter needs to be scaled by M. For each value of M, different multiband responses are obtained. If each of these multiband responses are masked using suitable masking filters, different low-pass, band-pass and high-pass channels (bands) can be obtained. Reconfigurability can be easily achieved by changing the value of M with a given set of filter coefficients. The proposed methodology can be illustrated with the help of Fig. 13. In Fig. 13a, represents



(a) Frequency response of original modal filter.

(b) Frequency response of modal filter with M=2.

(c) Frequency response of modal filter with M=3.

(d) Frequency response of modal filter with M=4.

(e) Frequency response of decimated modal filter with M=4.

Figure 13 **a** Frequency response of original modal filter. **b** Frequency response of modal filter M=2. **c** Frequency response of modal filter M=3. **d** Frequency response of modal filter M=4. **e** Frequency response of decimated modal filter with M=4.

the original modal filter with normalized (with respect to sampling frequency) passband and stopband specifications of f_p=0.05 and f_s=0.075. Let the passband and stopband ripple specifications be 0.1 dB and −55 dB respectively. Fig. 13b represents frequency response for M=2, i.e., the case when every second coefficients in the original filter coefficient set are kept unchanged and remaining coefficients are replaced by zeros. Note that the frequency response is obtained by scaling the coefficients by M=2. In the proposed reconfigurable FIR filter implementation, this can be achieved by scaling the output of the filter by M=2. This is possible because $(M \times h) \otimes x = M \times (h \otimes x)$, where x is the input and h represent the filter coefficients and $\otimes$ represents the convolution operation. As seen from (20) and Fig. 13b, for M=2, the frequency responses are obtained at $2\pi k/2 = \pi k$, for k=0 and 1. Similarly, Fig. 13c and d represent the case M=3 and M=4 respectively. Figure 13e is obtained as a special case of M=4. If every fourth coefficients are grouped together, a decimated frequency response compared to original frequency response of Fig. 13a is obtained with M=4. It can be seen from Fig. 13b to d that the stopband attenuation reduces as M increases. But it should be noted that the transition width remains unaltered for all values of M. Therefore, based on

the desired stopband attenuation of the final filter, the original modal filter should be designed with larger stopband attenuation keeping in account of the deterioration of the final filter's stopband response.

The proposed CD approach [19] can also be extended to develop a filter bank to extract multiple frequency bands simultaneously. This can be illustrated using Fig. 13. If the frequency responses in Fig. 13a to d can be obtained simultaneously, then by subtracting the outputs of Fig. 13b and Fig. 13c from Fig. 13a and the output of Fig. 13d from Fig. 13b, different frequency bands (channels) located at integer multiples of $2\pi/M$ can be extracted. A generalized architecture for the proposed FB is shown in Fig. 14. The frequency responses in Fig. 13 a to d are obtained using the filter bank structure in Fig. 14 as outputs y_1 to y_4 respectively. Also, the responses in Fig. 13a to c are obtained as outputs y_{21}, y_{31} and y_{42} in the architecture shown in Fig. 14. Thus CDFB is capable of extracting channels corresponding to the frequency responses in Figs. 13a to d and 15a to c simultaneously without the need of any extra filters or modulation operations. Note that neither the passband width nor the transition band width is altered while obtaining all the above frequency responses. CDFB is a low complexity alternative to DFTFB and is more flexible and easily reconfigurable than the DFT filter bank. Furthermore, the CDFB is able to receive channels of multiple standards simultaneously, where as separate filter banks would be required for simultaneous reception of multi-standard channels in a DFT filter bank based receiver.

5 Complexity Comparison

In this section, we present a quantitative comparison of different filter banks reviewed in this paper. Table 2 shows the comparison of the multiplication rate of the PC approach, DFTFB, GFB [5], MPRB [10], our FRMFB [16] and CDFB [19]. Multiplication complexity of a channelizer is defined as the total number of multiplications for extracting N_I number of channels (of same communication standard) simultaneously. The multiplications involved in a channelizer can be grouped into three categories: Multiplications associated with (1) Channel filtering; (2) Digital down conversion and (3) Modulation of filters (this is not applicable for PC approach, FRMFB and CDFB).

In Table 1, L represents the number of non-zero coefficients of the prototype filter for the PC approach, DFTFB and GFB, l represents the additional number of non-zero coefficients of the modal filter (because of over

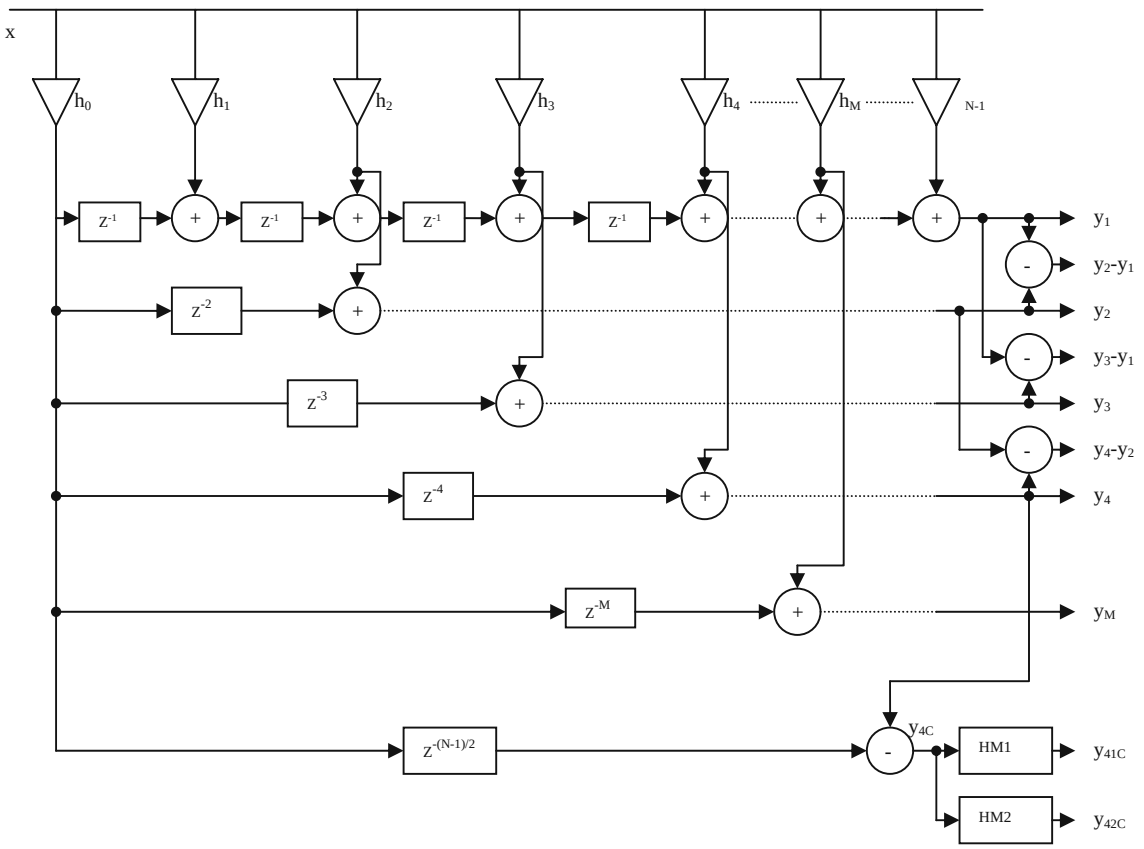


Figure 14 Architecture of the coefficient decimated filter bank.

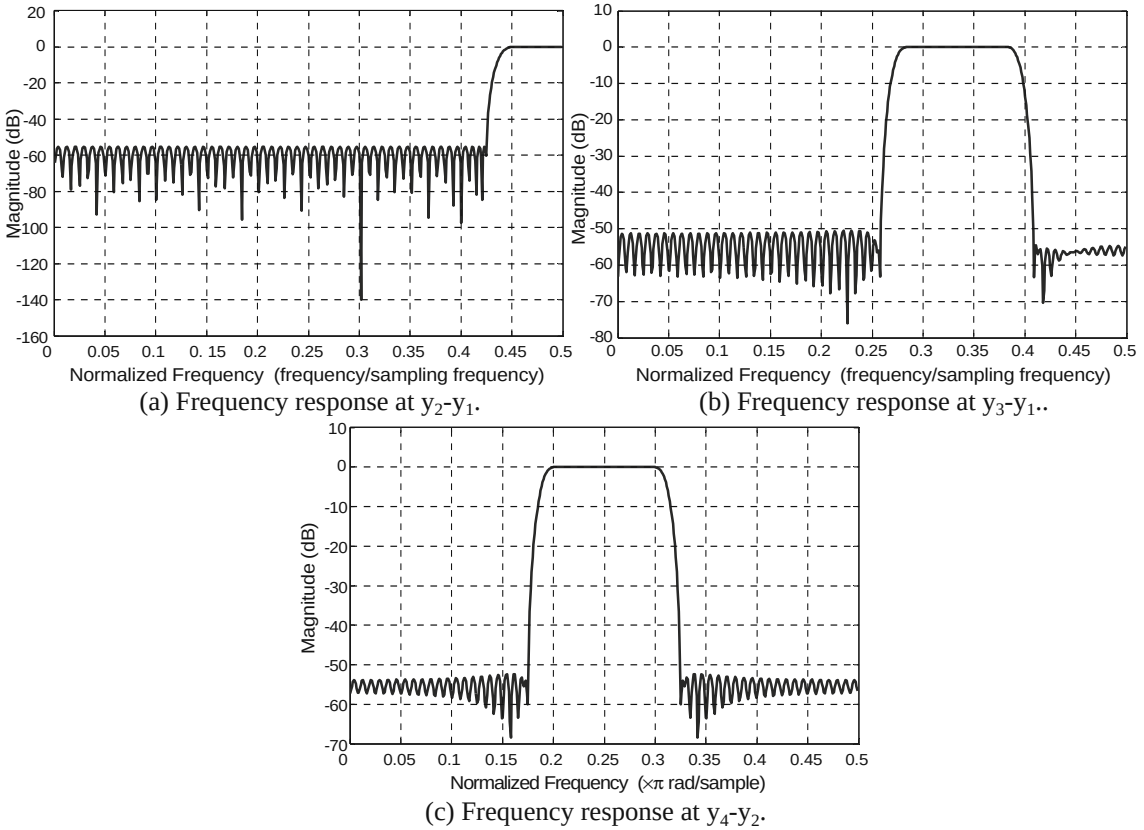


Figure 15 **a** Frequency response at y_2-y_1. **b** Frequency response at y_3-y_1. **c** Frequency response at y_4-y_2.

design) and masking filters in the proposed CDFB, *lm* represents the total number of non-zero coefficients for the modal filter and masking filters in the FRMFB (We have considered only non-zero coefficients as they will only result in multiplication complexity), and F_s represents the sampling frequency.

The multiplication complexities for PC approach, DFTFB and GFB are taken directly from [5]. From Table 2, it is clear that the complexity of PC approach is directly proportional to the number of channels, N_I. Thus higher the number of channels, the PC approach is not hardware efficient. It can be seen that, the complexity of filtering (multiplication) operation is same for the proposed DFTFB and GFB and slightly higher for CDFB (because of overdesigning and masking filters). The MPRB [10] consists of an analysis DFTFB and a synthesis DFTFB and hence the complexity is exactly twice that of DFTFB. As the FRMFB and CDFB do not require any DFT, there are no modulation complexity associated with FRMFB and CDFB. However separate (N_I-1) digital down converters are required in the FRMFB and CDFB for converting all the channels except the low-pass channel to baseband. We have not considered FFT for the implementation of IDFT in DFTFB and MPRB, as FFT is appropriate only if the number of channels to be extracted is a power-of-two. From Table 2, it can be seen that the over-all complexity of the proposed CDFB is lower than that of the PC approach, MPRB and GFB. Also, the proposed CDFB is less complex compared to DFTFB, when the number of channels, N_I, increases. The FRMFB has the least filter length because of the wide transition-band subfilters compared to other filter banks and hence has the least over-all complexity.

Table 2 Multiplication rate of channelizers.

	PCApproach	DFTFB	GFB [5]	MPRB [10]	FRMFB [16]	CDFB [19]
Filter	$N_I \bullet L$	L	L	$2L$	lm	$L+l$
DDC	N_I-1	–	–	–	N_I-1	N_I-1
Modulation of filters	–	N_I^2	$N_I \bullet L$	$2N_I^2$	–	–
Sum	$N_I \bullet (L+1)-1$	$L+N_I^2$	$L \bullet (1+N_I)$	$2(L+N_I^2)$	$lm+N_I-1$	$L+l+N_I-1$

However, the design of FRMFB is a tedious task as it follows separate design procedure for each bandpass channel. For the extraction of channels of different standards (non-uniform bandwidth channels), only PC approach and FRMFB can be employed efficiently. Hence for non-uniform bandwidth channel extraction, FRMFB is a very good substitute for PC approach because of former's inherent low complexity and easy reconfigurability. Similarly for the extraction of channels of uniform bandwidth, CDFB is an excellent substitute for DFTFB, GFB and MPRB.

6 Conclusions

In this paper, we have studied different filter banks for channelizers in multi-standard software defined radio (SDR) receivers. Low complexity and reconfigurability are the two key requirements of filter banks in SDR receivers. From our studies it can be concluded that, none of the existing filter banks such as per-channel approach, discrete Fourier transform (DFT) based filter bank and its modifications satisfy the stringent requirements of SDR receivers. We have presented two of our contributions which can satisfy the above requirements and can be employed as substitutes for the traditional approaches. To be more precise, the filter bank based on frequency response masking can be used as an efficient substitute for the per-channel approach and the filter bank based on the coefficient decimation approach can be used as a low complexity alternative to DFT filter banks.

References

1. Mitola, J. (2000). *Software radio architecture*, John Wiley.
2. Hentschel, T., Henker, M., & Fettweis, G. (1999). "The digital front-end of software radio terminals," IEEE Person. Commun. Magazine, pp. 40–46.
3. Vinod, A. P., Premkumar, A. B., & Lai, E. M.-K. (2003). "An optimal Entropy coding scheme for efficient implementation of pulse shaping FIR filters in digital receivers," in *Proc. of IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 229–232, Bangkok, Thailand.
4. Vinod, A. P., & Lai, E. M-K. (2006). "Low Power and High-Speed implementation of FIR filters for software defined radio receivers," *IEEE Transactions on Wireless Communications,* pp. 1669–1675, no. 5, vol. 7.
5. Hentschel, T. (2002). "Channelization for software defined base-stations.". *Annales des Telecommunications*, *57*(5-6), 386–420 ISSN 0003-4347.
6. Zabre, S., Palicot, J., Louet, Y., Moy, C., & Lereau, C. (2006). "Carrier per Carrier Analysis of SDR Signals Power Ratio", SDR Forum Technical Conference, Orlando, US.
7. Phil Schniter: http://cnx.org/content/m10424/latest/.
8. Kim, C., Shin, Y., Im, S., & Lee, W. (2000). "SDR-based digital channelizer/de-channelizer for multiple CDMA signals," in Proceedings of IEEE Vehicular technology Conference, vol. 6, pp. 2862–2869.
9. Fung, C. Y., & Chan, S. C. (2002). "A multistage filter bank-based channelizer and its multiplier less realization", in Proceedings of IEEE International Symposium on Circuits and Systems, vol.3, pp. 429–432.
10. Abu-Al-Saud, W. A., & Stuber, G. L. (2004). Efficient wideband channelizer for software radio systems using modulated PR filter banks. *IEEE transactions on signal processing*, *52*(10), 2807–2820. doi:10.1109/TSP.2004.834242.
11. Lillington, J. (2003). "Comparison of wideband channelization architectures,"http://www.techonline.com/community/techgroup/dsp/tech paper/33204.
12. Vinod, A. P., Lai, E. M.-K., Premkumar, A. B., & Lau, C. T. (2003). "A reconfigurable multi-standard channelizer using QMF trees for software radio receivers," in proceedings of 14th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication, pp. 119–123.
13. Lillington, J. (2002). "TPFT–Tunable pipelined frequency transform," RF Engines Ltd, Technical Report. http://www.rfel.com/whipapdat.asp.
14. Zangi, K., & Koilpillai, R. (1999). Software radio issues in cellular base stations. *IEEE journal on selected areas in communications*, *17*(4), 561–573. doi:10.1109/49.761036.
15. Zangi, K., & Koilpillai, R. (1998). "Efficient filter bank channelizers for software radio receivers," in Proceedings of International Conference on Communications, vol. 3, pp. 1566–1570.
16. Mahesh, R., & Vinod, A. P. (2008). Christophe Moy and Jacques Palicot, "A low complexity reconfigurable filter bank architecture for spectrum sensing in cognitive radios," Proceedings of 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications, Singapore.
17. Lim, Y. C. (1986). Frequency-response masking approach for the synthesis of sharp linear phase digital filters. *IEEE transactions on circuits and systems*, *33*, 357–364. doi:10.1109/TCS.1986.1085988.
18. Delahaye, J. P., Leray, P., Moy, C., & Palicot, J. (2005). "Managing Dynamic Partial Reconfiguration on Actual Heterogeneous Platform", SDR Forum Technical Conference'05, Anaheim (USA).
19. Mahesh, R., & Vinod, A. P. (2008). "Coefficient decimation approach for realizing reconfigurable finite impulse response filters," in *Proceedings of IEEE International Symposium on Circuits and Systems,* pp. 81–84, Seattle, USA.
20. Mahesh, R., & Vinod, A. P. (2008). "Reconfigurable frequency response masking filters for software radio channelization.". *IEEE Transactions on Circuits and Systems-II*, *55*(3), 274–278.