This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2015.2469072, IEEE Transactions on Circuits and Systems II: Express Briefs

Design and Realization of Variable Digital Filters for Software Defined Radio Channelizers using Improved Coefficient Decimation Method

Abhishek Ambede*, *Student Member, IEEE*, Shanker Shreejith, *Student Member, IEEE*, A. P. Vinod, *Senior Member, IEEE*, and Suhaib A. Fahmy, *Senior Member, IEEE*

***Abstract*—Variable digital filters (VDFs) are used in software defined radio handsets for extraction of individual radio channels corresponding to multiple wireless communication standards. In this paper, we propose a VDF based on the improved coefficient decimation method (ICDM). The proposed VDF provides variable lowpass, highpass, bandpass, bandstop and multi-band frequency responses on-the-fly, using the same set of prototype filter coefficients. We present non-pipelined as well as pipelined implementation architectures for the proposed VDF, along with FPGA implementation results for multiple VDF designs. Analysis of the implementation results shows that the pipelined implementations achieve average reductions of 27.66%, 49.17% and 25.59% in the number of occupied slices, dynamic power and energy consumption respectively, when compared with corresponding non-pipelined implementations. Also, the proposed pipelined implementation architecture provides high operating frequencies that are independent of the prototype filter order across different VDF designs. An average maximum frequency of 157.89 MHz is obtained.**

***Index Terms*—Variable digital filter, improved coefficient decimation method, software defined radio.**

I. Introduction

Software defined radio (SDR) has been proposed and widely researched as a solution to seamlessly support the existing and upcoming wireless communication standards [1-3]. The SDR technology is being envisioned as an integral component of the next-generation 5G wireless communication networks that will feature high speed and spectrally efficient data, voice as well as peripheral device communications [4], [5]. SDRs have the ability of software reconfiguration of their transceiver architecture which enables transmission/ reception of signals corresponding to multiple wireless communication standards. This ability of SDRs offers the advantages of low hardware resource utilization and thus reduced size and costs. Field Programmable Gate Arrays (FPGAs) are widely used to realize SDR test beds. Their volatile nature allows run-time adaptability, which coupled with highly parallel architecture makes them best suited for energy efficient advanced baseband processing. Multiple SDR test beds make use of FPGA capabilities for accelerating baseband computation and run-time adaptation like IRIS platform [6] and WARP [7]. In [8, 9], the authors presented high-level synthesis based methods for direct mapping of SDR description in MATLAB/Domain Specific Language (DSL) to system architecture on FPGA. In [10], the authors presented a closely coupled SDR based cognitive radio platform based on the Zynq FPGA with high-speed software-based hardware reconfiguration and simple API controls for managing them.

The most computationally intensive block in the digital front end of SDR receivers is the channelizer, which extracts the desired radio channels (frequency bands) from a wideband input signal [3]. In SDR handsets, a single channel needs to be extracted during a certain time interval, but the channel bandwidth (BW) and location can be different at some other time interval when the handset works on a different wireless communication standard. Variable digital filters (VDFs), i.e., filters whose output frequency responses can be modified by controlling a small set of parameters, are typically used in SDR handsets to perform the channelization operation. In resource constrained SDR handsets, VDFs that provide variable frequency responses while ensuring low complexity of implementation and high speed of operation are desired.

The programmable finite impulse response (FIR) filter approach involves updating all filter coefficients each time a different frequency response is desired [11]. When multiple wireless communication standards are involved with varying signal locations in the input frequency range, a large number of distinct frequency responses are required. This leads to a large memory requirement to store the filter coefficients and causes significant reconfiguration delays. The programmable FIR filter approach is thus impractical for SDR handsets. In [12], a digit based reconfigurable FIR filter architecture was proposed. In this architecture, the tap (coefficient) number and the number of non-zero digits in each tap are arbitrarily assigned and the architecture is independent of the total number of taps involved. But this architecture has high hardware resource utilization thus making it impractical for resource constrained SDR handsets. In [13], the frequency response masking (FRM) technique was proposed to obtain

This paragraph of the first footnote will contain the date on which you submitted your paper for review.

A. Ambede, S. Shreejith, A. P. Vinod and S. A. Fahmy are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798. E-mail: {abhishek7, shreejit1, asvinod, sfahmy}@ntu.edu.sg.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2015.2469072, IEEE Transactions on Circuits and Systems II: Express Briefs

FIR filters with sharp transition BWs. In FRM, interpolation, i.e., replacement of each delay in the filter structure by D delays, is performed using different values of D to obtain multi-band frequency responses. The resultant and their complementary frequency responses are algebraically operated upon, and the desired subbands are extracted using suitable low order masking filters. Based on FRM, a reconfigurable FIR filter was proposed in [14] for SDR handsets. In [15], a coefficient decimation method (CDM) was proposed to obtain low complexity and reconfigurable FIR filters. CDM involves the selective usage of a single prototype filter's coefficients to obtain variable frequency responses. A modified coefficient decimation method (MCDM) which acts on the filter coefficients in a different way than the CDM was proposed in [16]. In [17], the combination of CDM and MCDM was proposed and the combined method was termed as improved coefficient decimation method (ICDM), consisting of two sets of coefficient decimation operations – ICDM-I and ICDM-II.

In this paper, we propose a comprehensive ICDM based VDF for SDR handsets. With the help of hardware implementation architecture and a detailed design procedure, we demonstrate how the ICDM can be employed in the proposed VDF to obtain variable frequency responses. We present FPGA implementation results for non-pipelined and pipelined implementation architectures of the proposed VDF along with their comparative analysis.

The rest of the paper is organized as follows: Section II includes a brief literature review of the ICDM. In Section III, the comprehensive ICDM based VDF is presented with its design procedure, non-pipelined and pipelined hardware implementation architectures, suitable design examples and FPGA implementation results for multiple VDF designs. Section IV presents our conclusions.

II. ICDM: Brief Literature Review

CDM consists of two operations - CDM-I and CDM-II [15]. In CDM-I operation, coefficients of a lowpass prototype (original) filter are decimated by a factor M, i.e., every M^{th} coefficient is retained and the others are replaced by zeros, to obtain a multi-band frequency response. The resultant frequency response has its subbands located at center frequencies given by even multiples of π/M, i.e., $2\pi k/M$ where k is an integer ranging from 0 to (M-1). In CDM-II operation, every M^{th} coefficient is retained and all others are discarded to obtain a lowpass frequency response with its BW M times that of the prototype filter. Similar to CDM, MCDM consists of two operations - MCDM-I and MCDM-II [16]. In MCDM-I operation, if the prototype filter is decimated by a factor M, every M^{th} coefficient is retained and the sign of every alternate retained coefficient is reversed. All other coefficients are replaced by zeros. This results in a multi-band frequency response with center frequency locations of the subbands given by odd multiples of π/M, i.e., $(2k+1)\pi/M$. In MCDM-II operation, every M^{th} coefficient is retained and all others are discarded. The sign of every alternate retained coefficient is reversed to obtain a highpass frequency response with its BW M times that of the prototype filter. The combination of CDM and MCDM is termed as ICDM, which consists of four operations – CDM-I, CDM-II, MCDM-I and MCDM-II. These are classified as ICDM-I (includes CDM-I and MCDM-I) and ICDM-II (includes CDM-II and MCDM-II) [17]. It can be noted that center frequency resolution of π/M can be achieved for the subbands in the output frequency responses obtained after performing ICDM-I operations. On the other hand, by performing ICDM-II operations, subband BWs that are integer multiples of that of the prototype filter can be obtained using appropriate values of M.

III. Comprehensive ICDM based VDF

A. Hardware Implementation Architecture

Fig. 1 shows the hardware implementation architecture for realizing the comprehensive ICDM based VDF. It is a transposed direct form FIR filter structure which is appropriately modified to enable ICDM operations on the filter coefficients. In Fig. 1, the multipliers $h_0, h_1, \ldots, h_{(N-1)}, h_N$ denote the prototype filter coefficients. Only half of the filter coefficients need to be implemented due to their symmetry property, which will reduce the number of coefficient multiplications by 50%. The multiplexers labeled as '*mux I*' are used to perform ICDM-I operations, i.e., replacement of filter coefficients by zeros according to decimation factor M_1. The multiplexers labeled as '*mux II*' are used to perform ICDM-II operations, i.e., elimination of filter coefficients according to decimation factor M_2. The adder/ subtractor (*add/ sub*) blocks are used to perform sign reversal of appropriate coefficients in the MCDM operations. The default operation of *add/ sub* blocks is addition. When sign of a filter coefficient is to be reversed in an MCDM operation, the corresponding *add/ sub* block is set to perform subtraction by providing the select (*sel*) signal appropriately. The usage of *add/ sub* blocks to perform sign reversal instead of multiplying the filter coefficients by -1 is more efficient in terms of hardware resource utilization. The '*decimation selector*' block is used to provide select signals of multiplexers *mux I* and *mux II* as well as those of the *add/sub* blocks. When $M_1=M_2$, the select signals of *mux I* and *mux II* corresponding to every filter coefficient are identical. The multiplier at the output of the implementation architecture is used to scale the filter output by M_{out}, to regain the original magnitude level of the passband after ICDM operations.

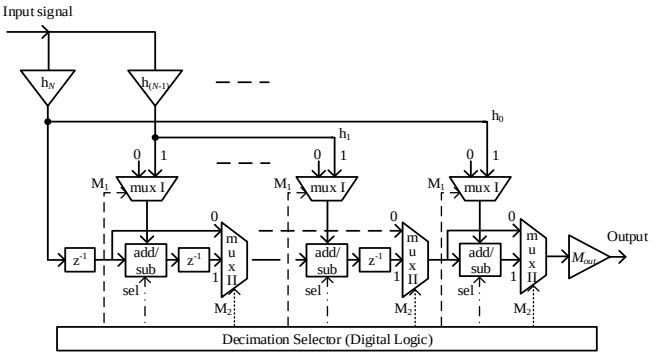

Fig. 1. Comprehensive ICDM based VDF: Hardware implementation architecture.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2015.2469072, IEEE Transactions on Circuits and Systems II: Express Briefs

B. Design Procedure

A detailed design procedure to obtain different frequency responses using the proposed VDF hardware implementation architecture (shown in Fig. 1) is presented in this section.

Step-1: Prototype filter design-

Compute the order of the required prototype filter using the formula [17]

$$N = \left[-\frac{4\log_{10}(10\delta_p\delta_s)}{3(f_s - f_p)} - 1 \right] + \frac{4\log_{10} M_{\max}}{3(f_s - f_p)} \qquad (1)$$

where N = order of the prototype filter,

f_p, f_s = passband and stopband cutoff frequencies of the prototype filter (normalized for the range 0 to 1, where 1 corresponds to half of the sampling frequency),

δ_p, δ_s = worst case (most stringent) passband and stopband ripple specifications for the output frequency responses,

$M_{\max}$ = maximum possible decimation factor (without aliasing), which can be calculated as $M_{\max} = \lfloor (1/f_s) \rfloor$.

Obtain the filter coefficients using a suitable filter design tool. Implement the filter coefficients using the proposed hardware implementation architecture shown in Fig. 1. The following steps describe how different types of frequency responses can be obtained by performing appropriate ICDM operations using the implemented filter structure.

Step-2: Selection of coefficient decimation factor values-

Let M_1 be the decimation factor required to obtain a desired type of frequency response using an appropriate ICDM-I operation (CDM-I or MCDM-I). The CDM-II operation can be simultaneously performed to modify BWs of subbands in the default ICDM-I output response. Let M_2 be the decimation factor required in the CDM-II operation.

Step-3: Obtaining desired frequency responses-

To obtain a frequency response similar to that of an ICDM-I operation by M_1 and having subband BWs M_2 times that of the default ICDM-I output response, perform the following operations simultaneously:-

3(a). CDM-II using M_2 to achieve the desired BW.

3(b). Appropriate ICDM-I operation (CDM-I or MCDM-I) using M_1'=M_1xM_2 to obtain desired frequency response.

3(c). Scaling the output of the filter using $M_{out} = M_1$xM_2.

Notes: Variable lowpass and highpass frequency responses having BWs M_2 times that of the prototype filter can be obtained by setting M_1=1 and performing CDM-II and MCDM-II operations respectively, using M_2. M_1 and M_2 are to be appropriately chosen based on the desired frequency response. The range of values for both M_1 and M_2 is {1, 2, 3, …, $M_{\max}$}. To prevent aliasing in the resultant frequency response, they should be chosen such that (M_1 x M_2) ≤ $M_{\max}$.

C. Illustrative Example

In this section, we illustrate the flexibility of the proposed VDF to provide variable frequency responses. Using MATLAB filter design tool, a lowpass prototype filter is obtained for the specifications f_p = 0.07, f_s = 0.1, δ_p = 0.05 dB, δ_s = -40 dB, $M_{\max}$ = 10. The order of the prototype filter is computed to be 180, using (1). Variable lowpass and highpass frequency responses can be obtained by setting M_1=1 and performing appropriate ICDM-II operations using M_2 = {1, 2, 3,…, 10}. Fig. 2(a) shows different lowpass and highpass frequency responses obtained by performing CDM-II and MCDM-II operations respectively. As described in *Step-3* of the design procedure, appropriate ICDM-I and CDM-II operations simultaneously, to obtain different bandpass, bandstop and multi-band frequency responses. Fig. 2(b) shows a few bandpass frequency responses obtained using the designed VDF. From figures 2(a) and 2(b), it can be noted that variable frequency responses can be obtained using the proposed VDF by performing appropriate ICDM operations on-the-fly, by simply selecting suitable values of M_1 and M_2 to control the multiplexers and *add/ sub* blocks. To increase the number of distinct possible frequency responses, the implementation architecture shown in Fig. 1 can be modified to include parallel branches to perform different ICDM operations on the same set of prototype filter coefficients. The output frequency responses from the parallel branches can be added/ subtracted from each other to retain the desired subbands and discard the unwanted ones. Complementary and interpolated frequency response operations can also be incorporated in the modified architecture by adding appropriate delay and multiplexer blocks. The different output frequency responses can be algebraically operated upon or suitably frequency response masked to obtain various desired subband configurations.

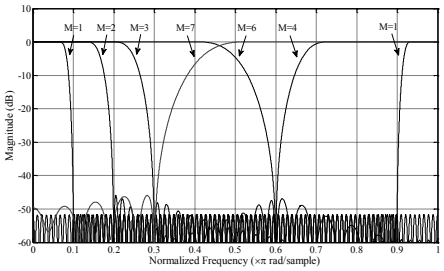

Fig. 2(a). Lowpass and highpass frequency responses obtained by performing CDM-II and MCDM-II operations respectively.

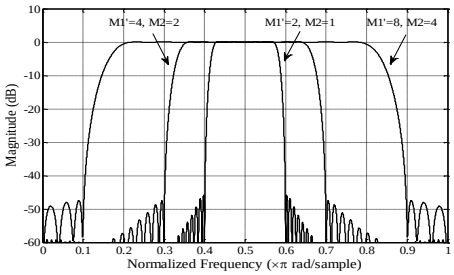

Fig. 2(b). Bandpass frequency responses obtained by performing MCDM-I using M_1=2 and different CDM-II operations.

D. Channelization Design Example

The ability of the comprehensive ICDM based VDF to extract channels corresponding to different wireless communication standards is illustrated in this section. Consider a SDR channelization scenario wherein channels corresponding to three standards - WCDMA, WiMAX and LTE are to be extracted in different time intervals.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2015.2469072, IEEE Transactions on Circuits and Systems II: Express Briefs

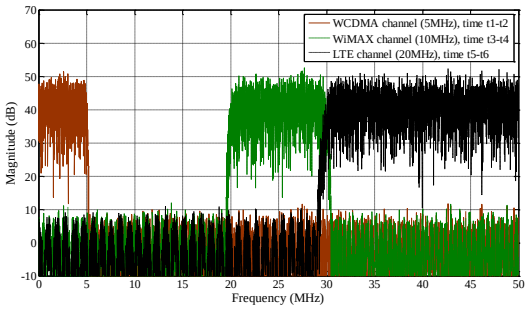

Fig. 3(a). Input signal containing WCDMA, WiMAX and LTE channels at different time intervals.

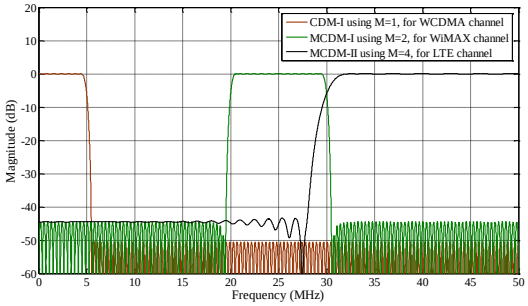

Fig. 3(b). Frequency responses used for extracting the three desired channels shown in Fig. 3(a).

Fig. 3(a) shows one channel each of WCDMA, WiMAX and LTE with BWs 5MHz, 10MHz and 20MHz respectively, present in the input signal at different time intervals, i.e., WCDMA signal during the time interval t_1-t_2, WiMAX during t_3-t_4 and LTE during t_5-t_6. The sampling frequency is 100MHz. Let the desired passband and stopband ripple specifications be 0.05dB and - 40dB respectively. The design procedure given in Section III-B is used to obtain a comprehensive ICDM based VDF which can extract these three desired channels from the input signal during their corresponding time intervals. The prototype filter is designed for the specifications f_p = 0.09, f_s = 0.11, $M_{\max}$ = 9, with order computed to be 266 using (1). Fig. 3(b) shows the frequency responses obtained after appropriate ICDM operations, which are used to extract the desired channels from the input signal.

E. Implementation Results

In this section, we present hardware implementation results for different VDF designs of the comprehensive ICDM based VDF. Following the design procedure given in Section III-B, three comprehensive ICDM based VDFs: VDF-I, VDF-II and VDF-III were designed with prototype filters of orders 60, 120 and 180 respectively. The three VDFs were implemented on a Xilinx Virtex-6 (xc6vlx240t) field-programmable gate array (FPGA) based on the hardware implementation architecture shown in Fig. 1. DSP blocks on FPGAs enable high throughput filter design, however, they are optimized for traditional transpose form filter structures. High level descriptions can map such structures efficiently. But with complex structures, their performance benefits are more difficult to take advantage of, hence custom implementations are necessary [18]. We disabled DSP block inference for all

TABLE I
FPGA IMPLEMENTATION RESULTS: COMPREHENSIVE ICDM BASED VDF DESIGNS

	VDF-I	VDF-II	VDF-III
No. of occupied slices	4280	7993	12276
Maximum Frequency (MHz)	124.83	100.02	85.25
Dynamic Power (mW) @ 50MHz	350.57	580.54	1017.40
Energy for 100 cycles (mJ)	1.136	2.578	5.738

the VDF implementations since the fixed coefficients and the proposed complex structure do not benefit from their use. Table I shows the resource utilization and performance (maximum operating frequency) of the three VDFs and the corresponding power estimates generated using post-place and route simulation data. As expected, it is observed that increasing order of the prototype filter results in degraded performance since the critical path length increases with depth of the combinatorial computational path. This dependence of operating frequency on the prototype filter order can be a major bottleneck if high operating frequencies are desired along with stringent frequency response specifications, which necessitate the use of higher order prototype filters.

To eliminate the operating frequency bottleneck, we present a modified hardware implementation architecture for the comprehensive ICDM based VDF, as shown in Fig. 4. The modified architecture is extensively pipelined to break the combinatorial path, which is then balanced in the input data-path by adding delay-chains. The pipelined architecture, though increases the latency from input to output by N clock cycles (for N-order prototype filter), isolates the critical path and prevents the accumulation effect that is present in the original architecture.

To compare the pipelined architecture (Fig. 4) with the straightforward non-pipelined architecture (Fig. 1), the three VDF designs were re-implemented using the pipelined architecture, and the same design options for the same FPGA device. Let the pipelined VDF implementations be denoted as VDF-I-P, VDF-II-P and VDF-III-P with prototype filters of orders 60, 120 and 180 respectively. Table II shows the corresponding implementation results. Comparing the results of the different VDF designs shown in tables I and II, we can observe that the pipelined implementations achieve average

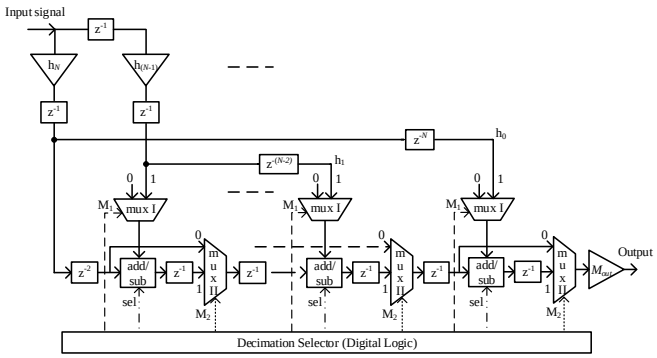

Fig. 4. Comprehensive ICDM based VDF: Pipelined hardware implementation architecture.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2015.2469072, IEEE Transactions on Circuits and Systems II: Express Briefs

TABLE II
FPGA IMPLEMENTATION RESULTS: PIPELINED COMPREHENSIVE ICDM BASED VDF DESIGNS

	VDF-I-P	VDF-II-P	VDF-III-P
No. of occupied slices	2985	5789	9188
Maximum Frequency (MHz)	157.85	158.08	157.75
Dynamic Power (mW) @ 50MHz	183.23	277.92	474.87
Energy for 100 cycles (mJ)	0.821	1.912	4.407

reduction of 27.66% in the number of occupied slices, despite the increased number of registers used. The pipeline stages added in the designs allow the synthesis tool to optimize the multiplexers and related logic in the delay-chain, thus reducing the overall resource usage. The reduction in number of occupied slices directly translates to an average reduction of 51.06% in dynamic power consumption for the different VDF designs. Tables I and II also show the energy consumed by the different VDF designs for computing filter output for 100 input samples. It can be observed that the pipelined implementations achieve average energy reduction of 25.59% when compared with the non-pipelined implementations. To ensure a fair comparison, all the power and energy estimates were generated based on signal activity rates determined by simulating the designs at an operating frequency of 50MHz. When the operating frequencies are compared, it can be observed that the pipelined implementations achieve significantly higher operating frequencies than their non-pipelined counterparts. An average maximum operating frequency of 157.89 MHz was obtained for the pipelined implementations, which remained consistent across the different VDF designs irrespective of the prototype filter orders. The proposed pipelined implementation architecture thus enables efficient realization of the comprehensive ICDM based VDF designs with high operating frequencies that are independent of the prototype filter order.

IV. CONCLUSION

In this paper, we proposed a variable digital filter (VDF) based on the low complexity improved coefficient decimation method (ICDM). The proposed comprehensive ICDM based VDF provides variable lowpass, highpass, bandpass, bandstop, and multi-band frequency responses on-the-fly, using a single set of prototype filter coefficients. A design procedure and hardware implementation architecture for the comprehensive ICDM based VDF were provided. Low complexity FPGA implementations were achieved for multiple VDF designs by pipelining the proposed hardware implementation architecture, which also enabled significantly higher operating frequencies that are independent of the prototype filter order. Due to the high frequency response flexibility of ICDM and the low complexity implementations with high operating frequencies that are possible, the comprehensive ICDM based VDF is highly suitable for channelization in resource constrained software defined radio (SDR) handsets. These characteristics also make the proposed VDF a promising candidate for use in SDR transceivers for the next-generation 5G wireless communications. The proposed VDF has a limitation of requirement of high order prototype filter when sharp transition BWs are desired, which may lead to high implementation complexity. To address this limitation, future work will involve combining the proposed VDF with the low complexity FRM [13] technique or using multi-stage filter realization techniques which are used to reduce overall filter orders and their implementation complexities.

REFERENCES

[1] J. Mitola, "The software radio architecture," *IEEE Commun. Mag.*, vol. 33, no. 5, pp. 26-38, May 1995.

[2] E. Buracchini, "The software radio concept," *IEEE Commun. Mag.*, vol. 38, no. 9, pp. 138-143, Sept. 2000.

[3] T. Hentschel, M. Henker and G. Fettweis, "The digital front-end of software radio terminals," *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 40-46, Aug. 1999.

[4] T. Janevski, "5G mobile phone concept," in *Proc. 6th IEEE CCNC*, Las Vegas, USA, Jan. 2009, pp. 1-2.

[5] J. G. Andrews, S. Buzzi, Choi Wan, S. V. Hanly, A. Lozano, A.C.K. Soong and J. C. Zhang, "What will 5G be?," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065-1082, Jun. 2014.

[6] K. Amiri, Y. Sun, P. Murphy, C. Hunter, J. R. Cavallaro, and A. Sabharwal, "WARP, a unified wireless network testbed for education and research," in *Proc. Intl. Conf. MSE*, San Diego, USA, Jun. 2007, pp. 53–54.

[7] P. D. Sutton, J. Lotze, H. Lahlou, S. A. Fahmy, K. E. Nolan, B. Ozgul, T. W. Rondeau, J. Noguera, and L. E. Doyle, "Iris: An architecture for cognitive radio networking testbeds," *IEEE Comm. Mag.*, vol. 48, no. 9, pp. 114–122, Sept. 2010.

[8] V. Bhatnagar, G.S. Ouedraogo, M. Gautier, A. Carer and O. Sentieys, "An FPGA software defined radio platform with a high-level synthesis design flow," in *Proc. 77th IEEE VTC Spring*, Dresden, Germany, Jun. 2013, pp. 1-5.

[9] G.S. Ouedraogo, M. Gautier and O. Sentieys, "Frame-based modeling for automatic synthesis of FPGA-Software Defined Radio," in *Proc. 9th Intl. Conf. CROWNCOM*, Oulu, Finland, Jun. 2014, pp. 341-346.

[10] S. Shreejith, B. Banarjee, K. Vipin, and S. A. Fahmy, "Dynamic Cognitive Radios on the Xilinx Zynq Hybrid FPGA", presented in *10th Intl. Conf. CROWNCOM*, Doha, Qatar, Apr. 2015.

[11] T. Solla and O. Vainio, "Comparison of programmable FIR filter architectures for low power," in *Proc. 28th ESSCIRC*, Florence, Italy, Sept. 2002, pp. 759-762.

[12] K-H Chen and T-D Chiueh, "A low-power digit-based reconfigurable FIR filter," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 8, pp. 617-621, Aug. 2006.

[13] Y. C. Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," *IEEE Trans. Circuits Syst.*, vol. 33, no. 4, pp. 357-364, Apr. 1986.

[14] R. Mahesh and A. P. Vinod, "Reconfigurable frequency response masking filters for software radio channelization," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 3, pp. 274-278, Mar. 2008.

[15] R. Mahesh and A. P. Vinod, "Low complexity flexible filter banks for uniform and non-uniform channelisation in software radios using coefficient decimation," *IET Circuits, Devices & Sys.*, vol. 5, no. 3, pp. 232-242, May 2011.

[16] A. Ambede, K. G. Smitha and A. P. Vinod, "A modified coefficient decimation method to realize low complexity FIR filters with enhanced frequency response flexibility and passband resolution," in *Proc. 35th Intl. Conf. TSP*, Prague, Czech Republic, Jul. 2012, pp. 658-661.

[17] A. Ambede, K. G. Smitha and A. P. Vinod, "A low complexity uniform and non-uniform digital filter bank based on an improved coefficient decimation method for multi-standard communication channelizers," *Springer Circuits Syst. Signal Process. (CSSP)*, vol. 32, no. 6, pp. 2543-2557, Dec. 2013.

[18] R. Bajaj and S. A. Fahmy, "Evaluating the efficiency of DSP Block synthesis inference from flow graphs," in *Proc. 22nd Intl. Conf. FPL*, Oslo, Norway, Aug. 2012, pp. 727-730.