

# AN EFFICIENT HARDWARE DESIGN OF AN OPTIMAL NONSTATIONARY FILTERING SYSTEM

*Srdjan Jovanovski, Veselin N. Ivanović*

*University of Montenegro, Dept. of Electrical Engineering, 81000 Podgorica, MONTENEGRO  
e-mails: [srdjaj@cg.ac.yu](mailto:srdjaj@cg.ac.yu), [very@cg.ac.yu](mailto:very@cg.ac.yu)*

## ABSTRACT

The development of a multi-cycle hardware design of a time-varying (TV) filtering system, suitable for real-time implementation on an integrated chip is outlined in this work. Based on results of time-frequency (TF) analysis and the instantaneous frequency (IF) estimation, the proposed design enables multiple detection of the local filter's region of support (FRS) in the observed time-instant, resulting in the efficient filtering of multicomponent FM signals. The proposed design optimizes critical design performances (such as hardware complexity, energy consumption and hardware cost), making it a suitable system for real-time implementation on a chip. The design has been verified by an FPGA (field-programmable gate array) circuit design.

**Index Terms**—Hardware design, Time-varying filtering, Instantaneous frequency estimation.

## 1. INTRODUCTION

Efficient nonstationary signals filtering requires a TV approach that may then benefit from the TF analysis results. Linear TF filters, their applications and online algorithms for their implementation have already been studied, [1]. Nonlinear filters, related to the Wigner distribution (WD), have also been studied, [2, 3], as well as the approaches for their implementation, [3, 4]. However, being quite complex, [3, 4], and unsuitable in the multicomponent signals case, [4], the considered approaches are unsuitable for real-time implementation. In this paper, a hardware design of the WD-related nonlinear TV filter suitable for real-time implementation has been proposed. The WD-related TV filtering definition, based on Weyl correspondence, [1–3], that overcomes distortion of the filtered FM signal is, [3]:

$$(Hx)(n) = \sum_{k=-N/2+1}^{N/2} L_H(n, k) STFT_x(n, k). \quad (1)$$

$L_H(n, k)$  is the Weyl symbol for the FRS,  $STFT_x(n, k) = DFT_m\{w(m)x(n+m)\}$  is the short-time FT (STFT) of the  $q$ -component noisy signal  $x(n) = \sum_{i=1}^q f_i(n) + \varepsilon(n)$ ,  $w(m)$  is

the real-valued lag window and  $N$  is the signal duration. Considering a single realization of the FM signals  $f_i(n)$ ,  $i=1, \dots, q$ , highly concentrated in TF plane and masked by a widely spread white noise, the FRS of the optimal TV filter corresponds to the union of the local IFs of signals  $f_i(n)$ , [2, 3]. Therefore, the filtering problem can be reduced to the local IF estimation in a noisy environment. In the TF analysis framework, this is performed by determining frequency points where TF distribution (TFD) of the noisy signal has local maxima, [5],

$$IF_i(n) = \arg[\max_{k \in Q_{k_i}} TFD_x(n, k)] \quad (2)$$

where  $Q_{k_i}$  is the basic frequency interval around  $f_i(n)$ , whose IF is  $IF_i(n)$ . Among the quadratic TF tools, the WD and the cross-terms-free WD, named the S-method (SM), [6], produce the best IF estimation characteristics in the highly nonstationary monocomponent and multicomponent signals, respectively, [5]. Thus, the TV filtering system (1), based on an already available SM real-time design, [7], and on the SM-based IF estimation, is proposed here. Moreover, the implementation based on the SM definition, [6, 7],

$$SM_x(n, k) = |STFT_x(n, k)|^2 + 2 \operatorname{Re} \left\{ \sum_{i=1}^L STFT_x(n, k+i) STFT_x^*(n, k-i) \right\} \quad (3)$$

requires the STFT samples, used also in the definition (1). In (3), the rectangular convolution window of width  $2L+1$  limits the summation in order to produce the cross-terms-free TFD. Note that (3) consists of real and imaginary computational lines, used for processing of STFT real and imaginary parts. Each of these lines takes form (3), obtained by replacing STFTs by their real and imaginary parts, respectively. In the case of real-valued signals, considered here,

$$(Hx)(n) = \sum_{k=-N/2+1}^{N/2} L_H(n, k) \operatorname{Re}\{STFT_x(n, k)\} \quad (4)$$

since  $SM_x(n, -k) = SM_x^*(n, k)$  is valid then, and, therefore,  $L_H(n, k)$  becomes a symmetric function in frequency, [4].

## 2. TV FILTER HARDWARE DESIGN

The architecture for the nonlinear TV filter real-time design

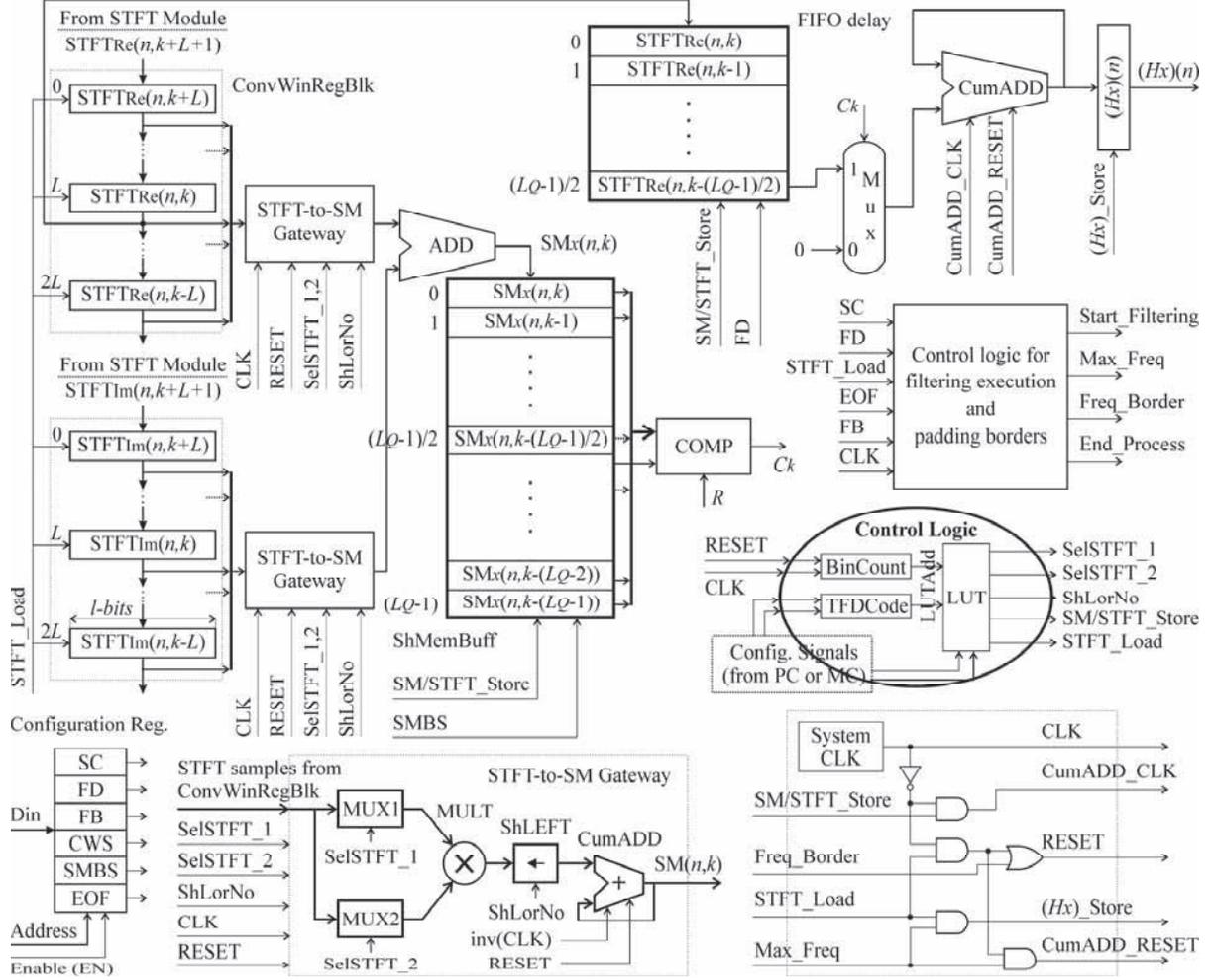


Figure 1: Proposed hardware design of the TV filtering system.

is given in Fig.1. The calculation is performed in  $L+3$  cycles per frequency point. ConvWinRegBlks and STFT-to-SM gateways, used in pairs, implement the SM real and imaginary computational lines in  $L+1$  cycles, as seen in [7]. The TV filter function is then implemented in the next two cycles. In  $(L+2)$ -nd cycle, the computed SM sample and the corresponding  $STFT_{Re}$  sample are stored in the shift memory buffer (ShMemBuff) and in the FIFO delay, respectively, by setting  $SM/STFT\_Store$  signal. In parallel, the COMP block, composing a set of comparators, generates the  $C_k$  signal that determines local IF. With the latency of half of the cycle,  $C_k=1$  enables participation of the FIFO delay output sample in the output signal generation. In  $(L+3)$ -rd cycle, the new STFT sample is imported by setting  $STFT\_Load$  signal, and the described process is repeated for the next frequency point. Simultaneously, and only when maximal frequency SM sample becomes the central ShMemBuff element, detected by the  $Max\_freq$  signal, the computed  $(Hx)(n)$  value is stored into the output register. With a latency of half of a cycle, the CumADD is reset and the execution, for the

next time-instant  $n$ , begins.

FIFO delay block is used to delay STFT samples. The FIFO delay output sample corresponds in frequency to the ShMemBuff central element. The ShMemBuff locations contain frequency-only-dependent SM samples from the basic frequency interval  $Q_k$ , eq.(2). The COMP block recognizes a local IF, determined by  $C_k=1$ , in the frequency point that corresponds to the maximal ShMemBuff element, but only if the maximal ShMemBuff element is: (i) the central ShMemBuff element, (ii) greater than the introduced spectral floor  $R$ , and if (iii) the ShMemBuff size ( $L_Q$ ) satisfies:

$$2 \times \max_{1 \leq i \leq q} \{A_i\} \leq L_Q < 2 \times \min_{\substack{1 \leq i, j \leq q \\ i \neq j}} |IF_i(n) - IF_j(n)| \quad (5)$$

where  $A_i$ ,  $i=1,2,\dots,q$  are the different widths of the non-overlapping SM auto-terms.

The above conditions (i) and (iii) have to be met in order to ensure:

- All frequency points from the observed auto-term, including the true IF, have the corresponding SM sample inside the ShMemBuff when the existence of the IF in

each of these points is investigated. This makes the IF estimation error to be noise-only-dependent inside the auto-terms' domains;

- For each auto-term and each time-instant  $n$ , only one value of  $L_H(n,k)$  can assume value 1. In this way, the influence of the frequency discretization on the IF estimation quality is reduced, as discussed in [3];
- Multiple detection of the local IFs in the observed time-instant  $n$  and, therefore, to enable the IF estimation in the case of multicomponent signals.

The condition (ii) has to be met in order to significantly suppress the noise influence outside the auto-terms' domains.

The process is managed by the Look-up-table (LUT). Its locations consist of the 3-bit control signals area (*ShLorNo*, *SM/STFT\_Store*, *STFT\_load* bits, respectively) and MUXs' addresses, Table 1. The binary counter generates its low addresses, while  $L$  from TFDCCode register sets the high ones. Operations at the maximal frequency are managed by *Start\_Filtering*, *Max\_Freq*, *Freq\_Border* and *End\_Process* signals. These signals are generated in the modules whose basic components are variable length up-down binary counters with asynchronous reset and binary magnitude comparators with binary references from the Configuration registers, Table 2. Binary counters' synchronization conditions are related to the CLK and *STFT\_Load* cycles. The *Freq\_Border* signal is generated to reset the gateways and, therefore, to pad the frequency border with  $2L$  0's.

Finally, the values of  $L_Q$  and  $R$  parameters have to be set. Wide frequency range (5), obtained in the case of highly concentrated, non-overlapping FM signals, suggests the robustness of the IF estimation with respect to the  $L_Q$ . Therefore, the ShMemBuff of several locations,  $L_Q \sim 2L+1$ , should usually be sufficient. The greater  $R$  values almost remove the influence of noise outside the auto-terms' domains, but they can produce significant edge cutting of the finite duration auto-terms (chirp signals auto-terms). Based on extensive experimental work, the  $R$  value best suited to most applications has been set at 10%–25% of the maximal SM value.

In different number of cycles by frequency point, the proposed design produces different IF estimation/filtering quality. The quality is improved with the incremental number of cycles, i.e. as  $L$  increases. However, higher  $L$  values can cause the cross-terms appearance and a significant increase in execution time. Therefore, our proposal is for TV filter designs with relatively small  $L$  ( $L=2$  or 3), since they already give the cross-terms-free WD estimation quality, [5], and have a small enough execution time (5 or 6 cycles by frequency point).

### 3. TESTING AND VERIFICATION

The proposed approach, Fig.1, is verified by an FPGA device real-time design. Real-valued test signal

Table 1: LUT's memory locations. The  $ADD_M$  denotes the address of the middle ConvWinRegBlk element. Symbol  $\ll$  denotes shift left logical operation and  $l = \text{Length}(\text{SelSTFT}_1)$ .

LUT Add	Ctrl signals area			SelSTFT_1	SelSTFT_2
0	0	0	0	$ADD_M \ll l$	$ADD_M$
1	1	0	0	$ADD_{M+1} \ll l$	$ADD_{M-1}$
.....	1	0	0	.....	.....
$L$	1	0	0	$ADD_{M+L} \ll l$	$ADD_{M-L}$
$L+1$	0	1	0	0	0
$L+2$	0	0	1	0	0

Table 2: Configuration registers' parameters, expressed by the number of needed *STFT\_Load* cycles.

Configuration register	Parameter's value
Start Convolution (SC)	$(2L+1)-1$
Filtering/FIFO Delay (FD)	$(L_Q-1)/2+1$
Frequency Border (FB)	$N-L-1$
Conv. Win. Size (CWS)	$2L+1$
ShMemBuff Size (SMBS)	$L_Q$
End of Filtering (EOF)	$N \times N-1$

$$f(t) = \sum_{i=1}^3 e^{-\alpha_i(t-\beta_i)^2} \cos(\gamma_i(t+\delta_i)^2) + \cos(3100t) \quad (6)$$

has been considered within the time-interval  $[0.1,1]$ , where  $\alpha_1=1$ ,  $\alpha_2=\alpha_3=100$ ,  $\beta_1=9/6$ ,  $\beta_2=3/8$ ,  $\beta_3=13/16$ ,  $\gamma_1=680$ ,  $\gamma_2=325$ ,  $\gamma_3=540$ ,  $\delta_1=1/5$ ,  $\delta_2=2$ ,  $\delta_3=-2/5$  and  $t = nT_w/N$ . It is masked by the high white noise such that  $SNR_m = 10\log(P_s/P_n) = -0.37$ [dB]. The Hanning lag window width of  $T_w=0.15$ , and  $L=3$ ,  $L_Q=11$ ,  $R=0.15 \times \max_{n,k}\{SM_x(n,k)\}$ ,  $N=256$  are applied. The results of the real-time FPGA implementation are presented in Fig.2.(f). Regardless of the negative influence of the frequency discretization, efficiency of the proposed TV filter design is evident, Figs.2.(d)–(f). The SNR improvement of 14.95[dB] has been achieved. It can be considered as very high, since the theoretical SNR improvement of up to approximately  $10\log(N/6)=16.3$ [dB] can be expected in a 6-component signal case, Fig.2.(a)–(c).

### 4. COMPARATIVE ANALYSIS AND CONCLUSIONS

Comparison of the architectures' resources used in the proposed and the existing designs, [1], and a comparison of their computational costs are given in Table 3. The proposed multi-cycle design significantly reduces the hardware complexity, and therefore facilitates substantial reduction of the used chip dimensions, energy consumption and hardware cost. In addition, the existing single-cycle online designs are highly dependent on the signal duration  $N$ . For larger  $N$  their hardware complexity significantly increases and, therefore, the corresponding real-time implementations on the chip are not always possible. On the other hand, the realization of the proposed design is always possible, since a small and const-

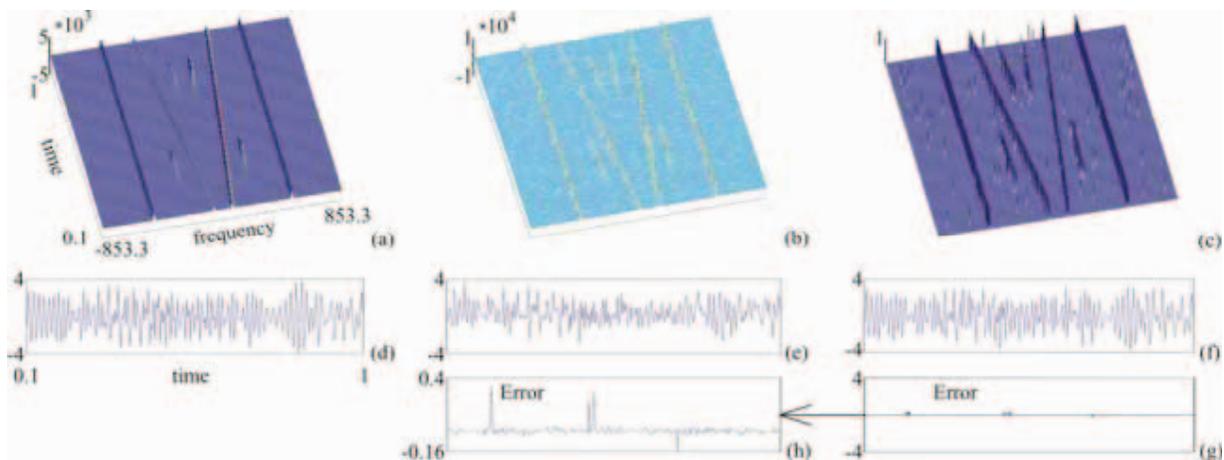


Figure 2: (a) SM of the signal  $f(t)$ ; (b) SM of the noisy signal; (c) Estimated FRS; (d) Signal  $f(t)$ ; (e) Noisy signal; (f) Output signal of the proposed hardware design, implemented in real FPGA devices, (g) Filtering error, (h) Enlarged filtering error.

Table 3. Complexity (hardware complexity and computational cost) of various online TV filters.  $\xi$  is the oversampling factor, used in the multiwindow Gabor filter, [1]. Shift left logical operation is not considered when the computational costs are given, because the time needed for its execution is much shorter than the time needed for other operations.

Filter type	Hardware complexity		# of operations per output sample
	# of used functional units	# of memory locations	
Zadeh	$N+N\log N$	$N^2/2+N$	$O(N+N\log N)$
Minimum-energy Weyl	$N+N\log N$	$N^2/2+N$	$O(N+N\log N)$
Approximate halfband Weyl	$N/2+N/2\log(N/2)$	$N^2/4+N$	$O(N/2+N/2\log(N/2))$
Multiwindow STFT	$2N^2(\log N+1+1/(2N))$	$N^2/2+3N/2$	$O(2N^2(\log N+1+1/(2N)))$
Multiwindow Gabor	$\xi N(2\log N+1)+\xi(N+1)$	$N^2/2+3N/2$	$O(\xi N(2\log N+1)+\xi(N+1))$
Proposed multi-cycle design	8	$5L+3L_0/2+27/2$	$O((2L+5)(N-2L))$

ant number of necessary functional units and memory locations are used in its real-time implementation.

The existing online designs, with the exception of the multiwindow STFT design, could have a better computational cost compared with the proposed design. However, the proposed design computational cost will be comparable with the existing designs, if the recommended TV filter design with relatively small  $L$  ( $L=2, 3$ ) is applied.

Finally, our design is capable of performing TV filtering of FM signals with arbitrary duration, since only computational cost of the proposed design depends on the signal duration  $N$ . On the other hand, existing designs complexities depend on  $N$ , resulting in ability of these systems to perform filtering of signals with predefined duration only.

## 5. REFERENCES

- [1] G. Matz, F. Hlawatsch: "Linear time-frequency filters: Online algorithms and applications," in *Applications in Time-Frequency Signal Processing* (A. Papandreou-Suppappola, ed.), CRC Press, 2002, pp.205–271.
- [2] G.F. Boudreaux-Bartels: "Time-varying signal processing using Wigner distribution synthesis techniques," in *The Wigner*

*Distribution – Theory and Applications in Signal Processing* (W. Mecklenbräuker and F. Hlawatsch, eds.), Elsevier, 1997, pp.269–317.

- [3] LJ. Stanković: "On the time-frequency analysis based filtering," *Ann. Telecomm.*, vol.55, no.5/6, May/June 2000, pp.216–225.
- [4] .Stanković, LJ. Stanković, V.N. Ivanović, R. Stojanović: "An architecture for the VLSI design of systems for time-frequency analysis and time-varying filtering," *Ann. Telecomm.*, vol.57, no.9–10, 2002, pp.974–995.
- [5] V.N. Ivanović, M. Daković, LJ. Stanković: "Performances of quadratic time-frequency distributions as instantaneous frequency estimators," *IEEE Trans. SP*, vol.51, no.1, Jan.2003, pp.77–89.
- [6] LJ. Stanković: "A method for time-frequency analysis," *IEEE Trans. SP*, vol.42, Jan.1994, pp.225–229.
- [7] V.N. Ivanović, R. Stojanović, LJ. Stanković: "Multiple clock cycle architecture for the VLSI design of a system for time-frequency analysis," *EURASIP J Appl.SP, Spec. Issue Design Methods for DSP Syst.*, vol.2006, pp.1–18.