14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

AN ARCHITECTURE FOR REAL-TIME DESIGN OF THE SYSTEM FOR MULTIDIMENSIONAL SIGNAL ANALYSIS

Veselin N. Ivanović, Radovan Stojanović, Srdjan Jovanovski, Ljubiša Stanković

Department of Electrical Engineering, University of Montenegro
Cetinjski put bb., 81000 Podgorica, MONTENEGRO
phone: + (381) 67 331 866, fax: + (381) 81 245 873, emails: {very,stox}@cg.ac.yu, srdjaj@cg.yu, l.stankovic@ieee.org
web: www.tfsa.cg.yu

ABSTRACT

Multiple clock cycle hardware implementation (MCI) of a flexible system for space/spatial-frequency signal analysis is proposed. Designed special purpose hardware can realize almost all commonly used two-dimensional space/spatial-frequency distributions (2-D S/SFDs) based on the 2-D Short-time Fourier transformation (2-D STFT) elements. The flexibility and the ability of sharing functional kernel, known as STFT-to-SM gateway, [1], *within S/SFDs execution, represent major advantages of this approach. These abilities enable one to optimize critical design performances of the multidimensional system, such as hardware complexity, energy consumption, and cost.*

1. INTRODUCTION

The STFT and the pseudo Wigner distribution (WD) (in their 1- and 2-D forms) represent conventional tools in time-frequency (TF), [3]-[4], and S/SF analysis, [5]-[9], because of their simplicity. But, they exhibit serious drawbacks: First – 1- and 2-D STFT have a low concentration around signals' instantaneous and local frequency, respectively, which may be inconvenient in many applications; Second – 1- and 2-D WD generate emphatic cross-terms in the case of multicomponent signals analysis, that seriously limit their applicability, despite the achieved high signals' concentration.

There are many attempts to overcome the mentioned problems. So, the 1- and 2-D S-method (2-D SM) for the TF and S/SF signal analysis, based on the corresponding STFT definition, are proposed in [4] and [7] and recently used in [10]-[15] and in [16]. Usage of the STFT, as an intermediate step in their definitions, makes them very attractive for implementation but, at the same time, quite numerically and time consuming, which significantly restricts their real-time applications. The hardware implementation, if possible, can overcome this nuisance. Having in mind the technology limitations in the hardware design, the 1-D systems for TF signal analysis are considered, usually in their single clock-cycle (parallel) implementation (SCI) forms, [17]. Such architectures for TF analysis and time-varying filtering, based on the SM, are presented in [13]-[15]. They are quite complex and require the duplication of basic calculation elements when they are employed more than once. Also, they realize only a single TFD – SM with a pre-defined convolution window width. In [1], [2] the MCI VLSI design, that overcomes drawbacks of parallel architectures from [13]-[15], [17], has been proposed.

There are many demands for development of 2-D S/SF systems. Such systems are more complex than the 1-D ones and often could not be realized. Additionally, the chip dimensions, power consumption and cost are significantly increased, while the processing speed is lowered. Therefore, in this paper we propose a way to extend the 1-D MCI architecture to the 2-D case. For this purpose a special architecture for S/SF analysis, based on the 2-D SM, has been developed. It consists of the MCI kernel and additional modules for convolution windowing, delays, frame managing, clock distribution, zero padding, and so on. The MCI allows a functional kernel to be used more than once per S/SFDs execution, as long as it is used on different clock cycles. The abilities to allow S/SFDs to take different number of clock cycles and to share a functional kernel within the execution of a single S/SFD are the major advantages of the proposed design. This will optimize the hardware requirements, giving us the possibility to realize S/SFDs by using standard devices.

The paper is organized as follows. After the introduction, the implemented S/SFDs are presented. MCI architecture for the 2-D system implementation is proposed in Section III.

2. REVIEW OF THE IMPLEMENTED S/SFDs

Two-dimensional SM, [7], is based on the unified definition of the 2-D STFT and the 2-D WD,

$$SM(n_1,n_2,k_1,k_2)=\sum_{i_1=-L_1}^{L_1}\sum_{i_2=-L_2}^{L_2}P(i_1,i_2)$$
$$\times STFT(n_1,n_2,k_1+i_1,k_2+i_2) \qquad (1)$$
$$\times STFT^*(n_1,n_2,k_1-i_1,k_2-i_2),$$

where the STFT of a 2-D signal $f(n_1,n_2)$ is:

$$STFT(n_1,n_2,k_1,k_2)$$
$$=\sum_{i_1=-N/2+1}^{N/2}\sum_{i_2=-N/2+1}^{N/2}f(n_1+i_1,n_2+i_2) \qquad (2)$$
$$\times w(i_1,i_2)e^{-j\frac{2\pi}{N}(i_1k_1+i_2k_2)}.$$

The assumed signal's duration is $N\times N$, $N=2^m$, and $P(i_1,i_2)$ is frequency domain (convolution) 2-D window, whose widths are $2L_1+1$ and $2L_2+1$ (i.e. $P(i_1,i_2)=0$, for $|i_1|>L_1$ and for $|i_2|>L_2$).

In the sequel, the 2-D SM (1) is implemented due to following reasons:

1. For the marginal cases of the window $P(i_1,i_2)$, when $L_1=L_2=0$, and $L_1=L_2=(N-1)/2$, the 2-D SPEC, $SPEC(n_1,n_2,k_1,k_2)= |STFT(n_1,n_2,k_1,k_2)|^2$, and the 2-D WD are obtained, respectively. Then, the arbitrary analysis of the 2-D SM unifies the corresponding analysis of the commonly used S/SFDs, 2-D SPEC and 2-D WD. Flexibility of the proposed hardware solution (in the rest of paper) is based on these facts.
2. (1) is very interesting for practical realization, since it allows us to implement 2-D SM by using the 2-D STFT.
3. By an appropriate selection of the convolution window size, the 2-D SM reduces (or, in same cases, eliminates) cross-terms, without degrading high concentration around local frequency of the 2-D WD, [7], [16]. Also, it produces better results than the commonly used S/SFDs, the 2-D SPEC and the 2-D WD, regarding some essential demands such as calculation complexity, cross-terms reduction, and noise influence suppression, [4], [7], [16], [18].

For a rectangular 2-D window $P(i_1,i_2)$ and $L_1=L_2=L$, the 2D-SM definition (1) becomes:

$$SM(n_1,n_2,k_1,k_2) = SM_R(n_1,n_2,k_1,k_2) + SM_I(n_1,n_2,k_1,k_2),$$

where $SM_R(n_1,n_2,k_1,k_2)$ and $SM_I(n_1,n_2,k_1,k_2)$ take form:

$$\begin{aligned} SM_R(n_1,n_2,k_1,k_2) &= STFT^2_{\mathrm{Re}}(n_1,n_2,k_1+i_1,k_2+i_2) \\ &+2\sum_{i_1=0}^{L}\sum_{i_2=1}^{L} STFT_{\mathrm{Re}}(n_1,n_2,k_1+i_1,k_2+i_2) \\ &\quad \times STFT_{\mathrm{Re}}(n_1,n_2,k_1-i_1,k_2-i_2) \qquad (3) \\ &+2\sum_{i_1=1}^{L}\sum_{i_2=0}^{L} STFT_{\mathrm{Re}}(n_1,n_2,k_1+i_1,k_2-i_2) \\ &\quad \times STFT_{\mathrm{Re}}(n_1,n_2,k_1-i_1,k_2+i_2), \end{aligned}$$

and $STFT(n_1,n_2,k_1,k_2)= STFT_{\mathrm{Re}}(n_1,n_2,k_1,k_2)+jSTFT_{\mathrm{Im}}(n_1,n_2,k_1,k_2)$. The $SM_R(n_1,n_2,k_1,k_2)$ and the $SM_I(n_1,n_2,k_1,k_2)$ are obtained by processing of real and imaginary parts of $STFT(n_1,n_2,k_1,k_2)$, respectively. The $SM_I(n_1,n_2,k_1,k_2)$ takes form (3) and it would be obtained by replacing the real part of the 2-D STFT with its imaginary part. Eq.(3) gives 2-D SM for the point (k_1,k_2) of the 2-D frequency plane. It involves only real multiplications in order to adapt itself for real-time implementation. Note that the summation in (3) involves $CN(L) =1+(L+1)L+L(L+1)=2L^2+2L+1$ terms (which will correspond to the number of clock cycles ($CN(L)$)), obtained by multiplying 2-D STFT elements that are symmetrically distributed around the (k_1,k_2) point in the 2-D frequency plane.

3. HARDWARE IMPLEMENTATION APPROACH

Based on (3), the hardware implementation of (k_1,k_2)-th ($k_1,k_2=0,1,...,N-1$) channel of the 2-D SM is done through its real computational line, since the imaginary one is identical. The design principle follows the developed form of (3), where each summation term is executed during the corresponding step (which takes one clock cycle). By breaking the S/SFDs execution into clock cycles, we are able to balance the amount of work done in each cycle, resulting in minimization of clock cycle time. During the first clock, when $L_1=L_2=0$, the 2-D SPEC is executed from the 2-D STFT element, $STFT(n_1,n_2,k_1,k_2)$, situated in the middle point of the convolution window. Residual summation terms, for increased indexes i_1 and/or i_2 are obtained in the next steps (second, third, ...). This improves the S/SFD concentration, aiming to achieve the one obtained by the 2-D WD.

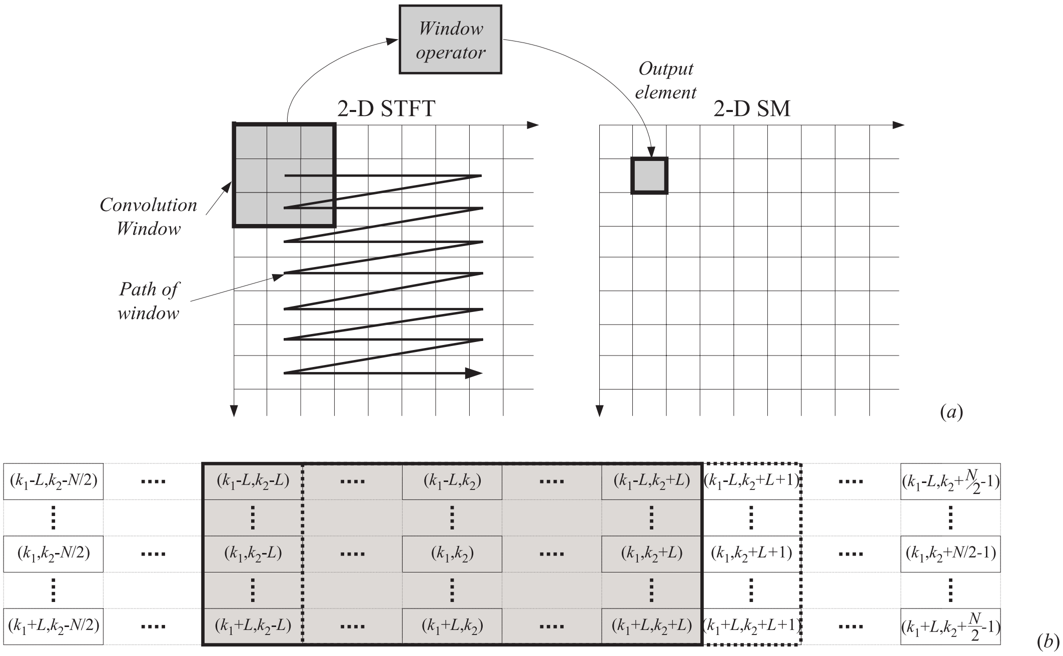
The 2-D SM with arbitrary L requires $CN(L)$ clock cycles (by one convolution window position) to be executed. The $CN(L)$ is a nonlinear function: 2-D SPEC requires one clock cycle, while 5 and 13 clock cycles are required for the 2-D SM with $L=1$ and $L=2$, respectively, etc. It is important to note that, for example, the 2-D SPEC execution step remains the first execution step for all S/SFDs with nonzero L values.

Generally seen, the hardware necessary for the 2-D SM real computational line consists of two main parts: the convolution window register file and the STFT-to-SM gateway. The convolution window register file presents the hardware implementation of the 2-D convolution window function. It determines order of the 2-D STFT input elements addresses for which the corresponding 2-D SM output will be computed according to the algorithm given by (3). The STFT-to-SM gateway is used to hardwarize this algorithm. It modifies the 2-D STFT elements obtained from the convolution window register file in order to produce improved concentration around local frequency based on the 2-D SM. STFT-to-SM gateway realizes 2-D SM calculation independently on the convolution window widths L, Table 1, [1], [2], allowing the implemented S/SFDs to take different numbers of clock cycles for their calculation. This is enabled by sharing STFT-to-SM functional units for different inputs in different steps (clock cycles) that are controlled by the set of control signals (see Fig.3. Details can be found in [1]). These abilities lead to minimize critical performances of the multidimensional systems: hardware complexity, energy consumption and cost. Namely, as it can be seen from Table 1, possible single clock cycle implementation of the 2-D SM (3) would involve significantly greater number of functional units necessary for the STFT-to-SM gateway implementation. It would significantly increase the necessary clock cycle time and make questionable the system (3) hardware implementation on chip when the single clock cycle implementation is prefered. Also, resource consumption of the hybrid implementation, [1], is beetwen the proposed one and single clock cycle implementations.

3.1 Convolution window register file

Each 2-D SM element at the system output is produced by sliding $(2L+1)\times(2L+1)$ convolution window $P(i_1,i_2)$ over 2-D STFT input elements, and by performing chosen operator, as is conceptually shown in Fig.1, (a)-(b). In our case, this operator is derived from (3) and implemented by the STFT-to-SM gateway, proposed and developed in [1], [2].

The implementation of the convolution window register file, given in Fig.2, follows the principles from Fig.1, (a)-(b). As seen in Fig.2, element $SM(n_1,n_2,k_1+L,k_2+L+1)$ represents an 2-D STFT input element, which will be loaded to the first element (register) of the convolution window register block.



$(k_1\text{-}L,k_2\text{-}N/2)$	····	$(k_1\text{-}L,k_2\text{-}L)$	····	$(k_1\text{-}L,k_2)$	····	$(k_1\text{-}L,k_2+L)$	$(k_1\text{-}L,k_2+L+1)$	····	$(k_1\text{-}L,k_2+\frac{N}{2}\text{-}1)$
⋮		⋮		⋮		⋮	⋮		⋮
$(k_1,k_2\text{-}N/2)$	····	$(k_1,k_2\text{-}L)$	····	(k_1,k_2)	····	(k_1,k_2+L)	(k_1,k_2+L+1)	····	$(k_1,k_2+N/2\text{-}1)$
⋮		⋮		⋮		⋮	⋮		⋮
$(k_1+L,k_2\text{-}N/2)$	····	$(k_1+L,k_2\text{-}L)$	····	(k_1+L,k_2)	····	(k_1+L,k_2+L)	(k_1+L,k_2+L+1)	····	$(k_1+L,k_2+\frac{N}{2}\text{-}1)$

Figure 1. Procedure of window sliding over 2-D STFT elements and of producing 2-D SM output: (a) Illustration of the convolution window sliding and its operation, (b) Position of the convolution window that corresponds to the (k_1,k_2)-th element of the 2-D SM (thick solid line box) and its next position (thick dashed line box). In the cell centre we denote the position of the stored 2-D STFT element in frequency-frequency plane.

STFT-to-SM gateway Implementation	**Adders**	**Multipliers**	**Shift Left Registers**	**Clock cycle time**
Proposed MCI	1	1	1	$T_m+T_a+T_s$
SCI of eq.(3) *when it would be possible*	$2L^2+2L$	$2L^2+2L+1$	$2L^2+2L$	$2T_m+(2L^2+2L+2)T_a+T_s$

Table 1: Total number of functional units in the STFT-to-SM gateway and corresponding clock cycle time in the cases of proposed MCI design and possible simple implementation of eq.(3). The 2-D SM with arbitrary L is considered. T_m is the multiplication time of a two-input adder, whereas T_s is the 1-bit shift time. The recursive form of the 2-D STFT module implementation is assumed when the clock cycle time in possible SCI case is calculated.

At the same time, each element of convolution window row k_1+L will be shifted by PIPO (parallel-in-parallel-out) registers to generate the 2-D STFT elements in time index (k_2+L-1, k_2+L-2, ..., k_2-L). The $2L$ FIFO delays are used to generate the 2-D STFT elements of the convolution window column k_2+L in time index (k_1+L-1, k_1+L-2, ..., k_1-L). It means that for each step of the convolution window register file, a new 2-D STFT element will be shifted in, while actual window will be sliced for one position left. Note that the convolution window register file is loaded with new 2-D STFT input element after at least $CN(L)$ clock cycles.

3.2 Architecture for the Real-Time Implementation

Conceptually, the hardware architecture we propose for implementation of the 2-D STFT to 2-D SM transformation is given in Fig.3 (as an example, L=1 is considered). It has several functional blocks and its operation principle can be described as follows. Hardware's 2-D STFT input is represented by STFT_IN signal which is controled by *STFT_IN_CLK*. Input values will be loaded to the first element of convolution window register block. PIPO shift register is used to generate data in time index (k_2, k_2-1) by shifting each element of the convolution window register block row k_1+1 (as well as rows k_1, k_1-1). FIFO delay blocks are used to generate data of the convolution window register block column k_2+1 in time index (k_1, k_1-1). Shifting of the STFT samples in the window area that is sliced for one position left, is controled by *SHIFT_IN_CLK*. It is necessary that the *STFT_IN_CLK* period be at least CN(1)=5 times greater then period of main system clock CLK. The task of "Control logic for windowed convolution and padding borders" is to manage convolution operation inside the frame. It generates control signals for STFT-to-SM gateway considering input parameters derived from frame size and window size. Table 2 describes these pa- rameters which are defined by "Configuration registers". As it is shown on Fig.3, control signals from "Control logic for windowed convolution and Padding borders" menage the operations of the STFT-to-SM gateway. At the same time, it considers input parameters and synhronization conditions related to the main clock signal CLK and the control signal

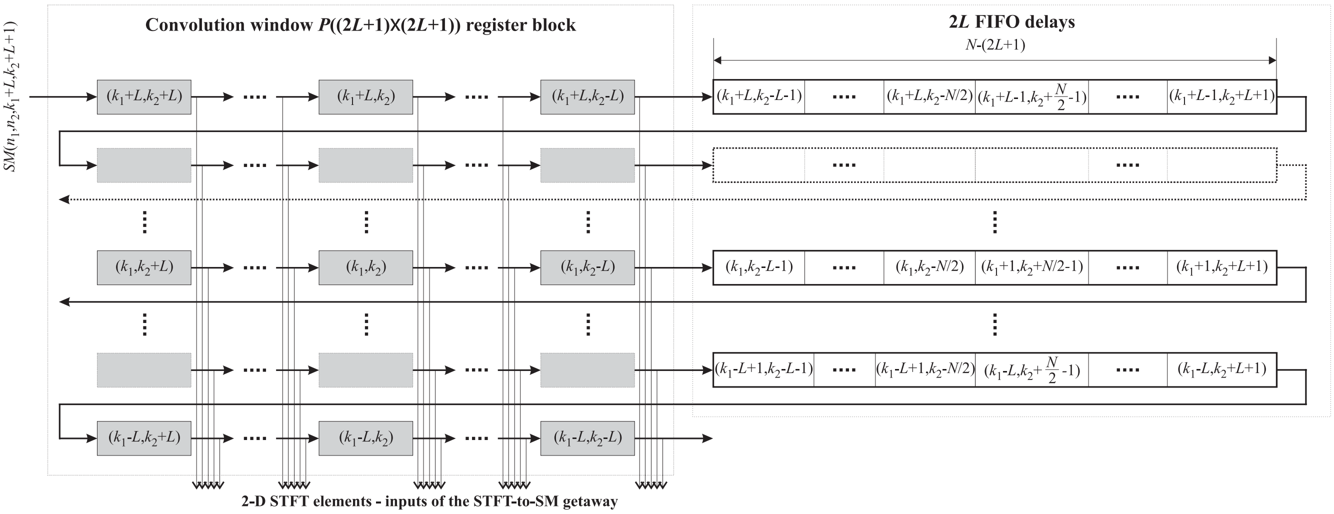


Figure 2. Convolution window register file implementation that realizes the convolution window function, illustrated in Fig.1.

STFT_CLK_IN and then, geneates the control signals *SM_START*, *SM_CLK_EN*, *LEFT_BORDER* and *DOWN_BORDER*. According to the algorithm (3), [1], [2], the *SM_CLK_EN* (SM_CLocK_ENable) signal forwards the series of SM_CLKs that run SM calculation for each window position. After *CN*(*L*)=5 *SM_CLK*s, the SM will be calculated and stored in the output register. Sliding window over 2-D signal generate *LEFT_BORDER* and *DOWN_BORDER* to allow to pad the borders of the frame with 0's.

For hardware realization, the CMOS SRAM FLEX 10K chips family have been chosen. After simulation and verification the Altera's EPF10K20RC240-3 chip is configured by using the synthesized code [19]. The rate of its 8-bit version silicon resources utilization are given in Table 3. Additionally, it has 189 (51 input and 114 output) I/O pins.

4. CONCLUSION

A multiple clock cycle implementation of an 2-D system for the S/SF analyze is proposed. The designed system is very flexible, since it allows the implemented S/SFDs to take different numbers of clock cycles and to share functional kernel, used to perform an S/SFD operation (STFT-to-SM gateway), within their execution. By designing MCI, the critical design parameters, such as hardware complexity, energy consumption and cost, are optimized.

REFERENCES

[1] V.N. Ivanović, R. Stojanović, and LJ. Stanković, "Multiple clock cycle architecture for the VLSI design of a system for time-frequency analysis," *EURASIP Journal on Applied Signal Processing, Special Issue on Design Methods for DSP Systems*, vol. 2006, pp. 1-18.

[2] V.N. Ivanović, and LJ. Stanković, "Multiple clock cycle real-time implementation of a system for time-frequency analysis," in *Proceedings of the 12th EUSIPCO*, Vienna, Austrija, Sept. 2004, pp.1633-1636.

[3] L.Cohen, *Time-frequency analysis*, Prentice Hall, 1995.

[4] LJ. Stanković, "A method for time-frequency analysis," *IEEE Trans. on SP*, vol. 42, no. 1, 1994, pp. 225-229.

[5] D.E. Dudgeon, and R.M. Mersereau, *Multidimensional digital signal processing*, Prentice Hall, 1984.

[6] L. Jacobsen, and H. Wechsler, "Joint spatial/spatial-frequency representation," *Signal Processing*, vol.14, 1988, pp.37-68.

[7] S. Stanković, LJ. Stanković, and Z. Uskoković, "On the local frequency, group shift and cross-terms in the multidimensional time-frequency distributions; A method for multidimensional time-frequency analysis," *IEEE Trans. on SP*, vol.43, no.7, July 1995, pp.1719-1725.

[8] Y.M. Zhu, F. Peyrin, et R. Goute, "Transformation de Wigner-Ville: description d' un nouvel outil de traitment du signal et des images," *Ann. Telec.*, vol.42, no.3-4, 1987, pp.105-117.

[9] G. Cristobal, C. Gonzalo, and J. Bescos, "Image filtering and analysis through the Wigner distribution function," in *Advances in Electronics and Electron Phisics, ed. P.W. Haekes, AcademicPress*, Boston, MA, 1991.

[10] P. Goncalves, and R.G. Baraniuk, "Pseudo affine Wigner distributions: Definition and kernel formulation," *IEEE Trans. on SP*, vol. 46, no. 6, 1998, pp. 1505-1517.

[11] C. Richard, "Time-frequency-based detection using discrete-time discrete-frequency Wigner distribution," *IEEE Trans. on SP*, vol. 50, no. 9, 2002, pp. 2170-2176.

[12] L.L. Scharf, and B. Friedlander, "Toeplitz and Hankel kernels for estimating time-varying spectra of discrete-time random processes," *IEEE Trans. on SP*, vol. 49, no. 1, 2001, pp. 179-189.

[13] S. Stanković, and LJ. Stanković, "An architecture for the realization of a system for time-frequency analysis," *IEEE Trans. on CAS-II*, vol. 44, no. 7, 1997, pp. 600-604.

[14] D. Petranović, S. Stanković, and LJ. Stanković, "Special purpose hardware for time-frequency analysis," *Electronics Letters*, vol. 33, no. 6, 1997, pp. 464-466.

[15] S. Stanković, LJ. Stanković, V.N. Ivanović, and R.Stojanović, "An architecture for the VLSI design of systems for time-frequency analysis and time-varying filtering," *Ann. Telec.*, vol. 57, no. 9-10, 2002, pp. 974-995.

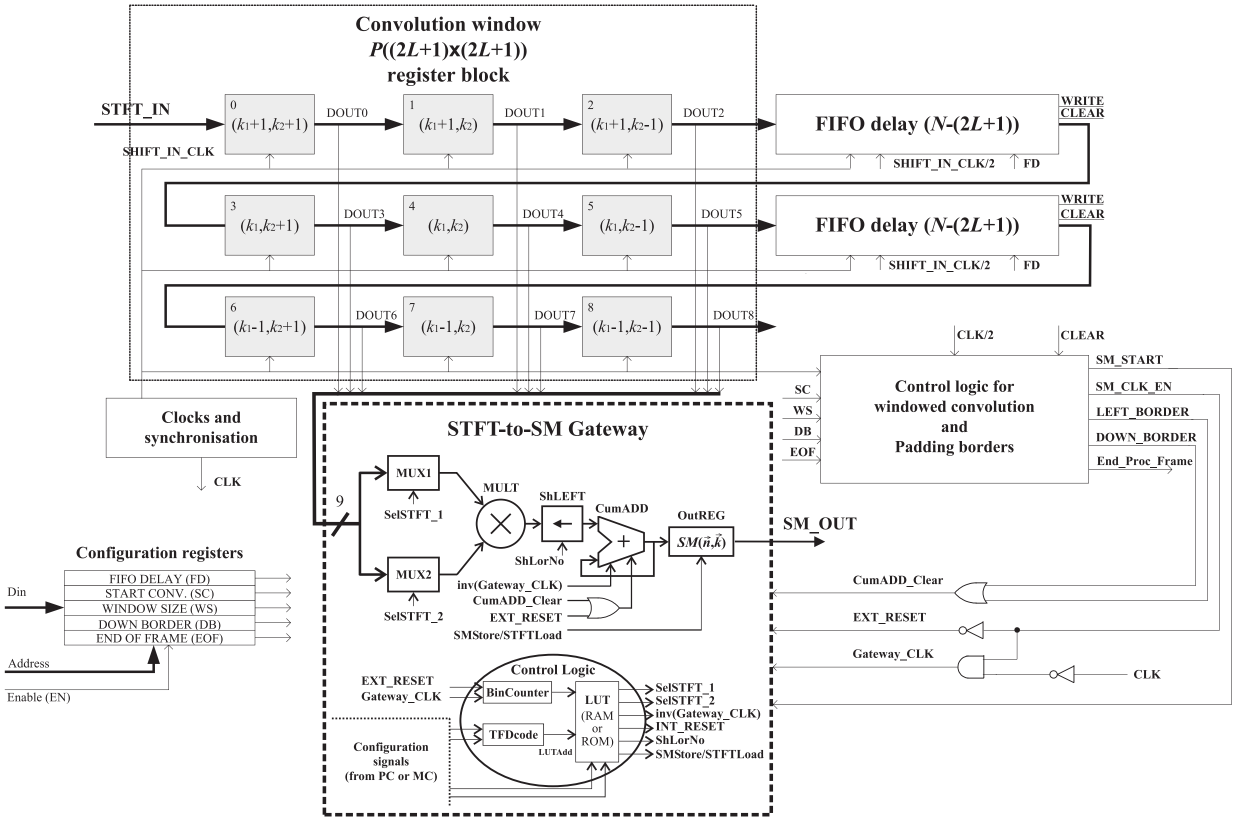


Figure 3. Proposed MCI design of the 2-D SM with L=1. In the centre of registers we denote position of the stored 2-D STFT element in frequency-frequency plane, whereas the number in the left upper register's corner represents the address position of the corresponding 2-D STFT element at the STFT-to-SM gateway input multiplexors.

Confuration register	Parameter specified and its description	Perameter’s value
FIFI Delay (FD)	Delay for generating data in row’s time index	$N-(2L+1)$
Start Convolution (SC)	Start of window operation	$2LN+(2L+1)-1$
Window Size (WS)	Size of the convolution window	$2L+1$
Down Border (DB)	Border position	$(N-2L)\times N$
End of Frame (EOF)	End of frame position	$N\times N-1$

Table 2: Parameters specified in “Configuration registers”, their description and values. Parameters are expressed by the number of the convolution window sliding steps.

Device	LCs	LCs utilized	Memory bits	Memory utilized	Embedded cells	Embedded cells utilized	EABs	EABs utilized	Flip-flops required
EPF10K20RC240-3	921	79%	1216	9%	28	58%	4	66%	217
EPF10K30BC356-3	965	55%	4288	34%	28	58%	4	66%	227

Table 3: Utilized silicon resource for 8-bit 64×64 and 8-bit 256×256 (second row) 2-D STFT-to-2-D SM implementation.

[16] LJ. Stanković, S. Stanković, and I. Djurović, “Space/spatial-frequency analysis based filtering,” *IEEE Trans on SP*, vol.48, no.8, Aug.2000, pp.2343-2352.

[17] K.J.R. Liu, “Novel parallel architectures for Short-time Fourier transform,” *IEEE Trans. on CAS-II*, vol. 40, no. 12, 1993, pp. 786-789.

[18] LJ. Stanković, V.N. Ivanović, and Z. Petrović, “Unified approach to the noise analysis in the Wigner distribution and spectrogram,” *Ann. Telec.*, vol. 51, no. 11-12, 1996, pp. 585-594.

[19] A. Iborra, C. Fernändez, B. Älvarez, J.M. Fernändez-Merono, “FPGA solution of low cost applications of real-time AVI systems,” *Dedicated Sys.Mag.*, vol.Q2, 2001, pp.79-84.