

AN FPGA DESIGN OF THE SYSTEM FOR SPACE/SPATIAL-FREQUENCY SIGNAL ANALYSIS

VESELIN N. IVANOVIĆ, RADOVAN STOJANOVIĆ, SRDJAN JOVANOVSKI

Department of Electrical Engineering

University of Montenegro

Cetinjski put bb., 81000 Podgorica

MONTENEGRO

phone: + (381) 67 331 866, fax: + (381) 81 245 873,

web: www.tfsa.cg.yu

Abstract: - FPGA implementation of the system for space/spatial-frequency (S/SF) signal analysis is developed. Multiple clock cycle hardware implementation (MCI) of this system is proposed in [1]. The developed system is based on the two-di-mensional S-method (2-D SM) and its relationship with the 2-D Short-Time Fourier Transformation (STFT). Designed system optimizes critical design performances of the multidimensional system (hardware complexity, energy consumption, and cost) by sharing functional kernel, known as the STFT-to-SM gateway, [2], [3], within the S/SFDs execution.

Key-Words: - space/spatial-frequency signal analysis, Multiple clock cycle hardware implementation.

1 Introduction

Conventional tools used in time (space/spatial)–frequency signal analysis, the spectrogram (SPEC) and the pseudo Wigner distribution (WD), exhibit serious problems: low SPEC concentration around analyzed signals' instantaneously (local) frequency and the emphatic interference effects in the case of multicomponent signal analysis by using WD. These problems seriously limit applicability of these conventional tools. Consequently, almost all methods, proposed in the past two or three decades, are defined to retain high resolution of the WD and, at the same time, to alleviate interference effects when the multicomponent signals are analyzed. The SM, [4], [5], represents a very successful and very popular, [6]–[13], attempt in overcoming the above noted problems. In the case of multidimensional (m -dimensional) signals analysis, the SM can be written in a following vector notation, [5], and [12]:

$$SM(\vec{n}, \vec{k}) = \sum_{\vec{i}} P(\vec{i}) STFT(\vec{n}, \vec{k} + \vec{i}) STFT^*(\vec{n}, \vec{k} - \vec{i}), \quad (1)$$

where $P(\vec{i})$ is rectangular frequency domain (convolution) window, with $2L+1$ width in each directions, whereas $STFT(\vec{n}, \vec{k})$ is the m -dimensional STFT of the analyzed m -dimensional signal $f(\vec{n})$, and $\vec{n} = (n_1, n_2, \dots, n_m) \in \mathbb{R}^m$

Usage of the STFT, as an intermediate step in the SM definition, makes SM very attractive for implementation but, at the same time, quite numerically and time consuming. This significantly restricts its real-time applications. The hardware

implementation, if possible, can overcome this nuisance. Additionally, the SM includes the STFT and the WD as its marginal cases, obtained for minimal and maximal convolution $P(\vec{i})$ window width, respectively. However, it produces better results than these conventional methods regarding some essential demands, such as calculation complexity, cross-terms reduction, and noise influence suppression, [4], [5], [12], [13].

For a long period of time, having in mind the technology limitations in the hardware design, only the 1-D systems for TF signal analysis are considered, usually in their single clock-cycle (parallel) implementation (SCI) forms, [9]–[11], [14]. They are quite complex and require duplication of the basic calculation elements when they are employed more than once. In [1]–[3] the MCI hardware design, that overcomes drawbacks of parallel architectures from [9]–[11], [14], has been proposed.

Recently, the demands for development of the multidimensional systems are increased. Such systems are more complex than the 1-D ones and often could not be realized: the chip dimensions, power consumption and cost are significantly increased, while the processing speed is lowered. In [1] we propose a way to extend the 1-D MCI architecture to the 2-D case. The MCI architecture, proposed in [1], 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

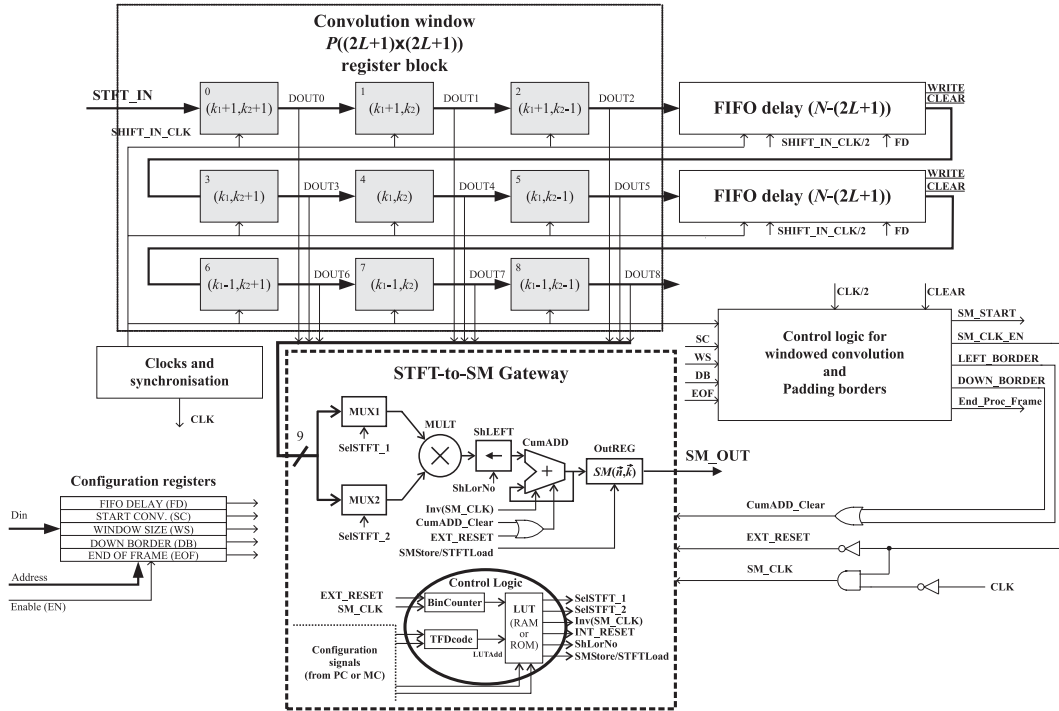


Fig. 1. Proposed MCI hardware design of the 2-D SM with $L=1$. In the centre of registers we denote position of the stored 2-D STFT element in frequency-frequency plane, whereas the number in the left upper register's corner represents the address position of the corresponding 2-D STFT element at the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

cycles and to share a functional kernel within the execution of a single S/SFD are highlighted as the major advantages of that design. Mentioned advantages optimize the hardware requirements. Using these possibilities here we realize S/SFDs by standard devices by developing the FPGA implementation of this system.

The paper is organized as follows. After the introduction, the overview of the implemented architecture is presented. FPGA implementation of the 2-D system is developed in Section III. In Section IV the system implementation is tested and verified.

2 OVERVIEW OF THE IMPLEMENTED ARCHITECTURE

The system for S/SF signal analysis is based on the SM (1). Since the STFT is the complex transformation, the $SM(\bar{n}, \bar{k})$ is calculated by independent processing of the $STFT(\bar{n}, \bar{k})$ real and imaginary parts, [1]–[3], [9]–[11]. Then, (1) involves only real multiplications and it is adapted for real-time hardware implementation. These parts of $SM(\bar{n}, \bar{k})$ take the same form. In the 2-D domain case, this form is:

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

Eq.(2) gives the 2-D SM for the point (k_1, k_2) of a 2-D frequency plane. It involves $CN(L)=1+(L+1)L+L(L+1)=2L^2+2L+1$ summation terms (which will correspond to the number of clock cycles ($CN(L)$) in MCI), obtained by multiplying 2-D STFT elements that are symmetrically distributed around the (k_1, k_2) point in the 2-D frequency plane.

The 2-D SM hardware implementation, shown in Fig.1, is done through its real computational line, since the imaginary one is identical. The design principle follows the developed form of (2), where each summation term is executed during the corresponding step (which takes one clock cycle). During the first clock, when $L=0$, the 2-D SPEC is executed from the 2-D STFT element, $STFT(\bar{n}, \bar{k})$, 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

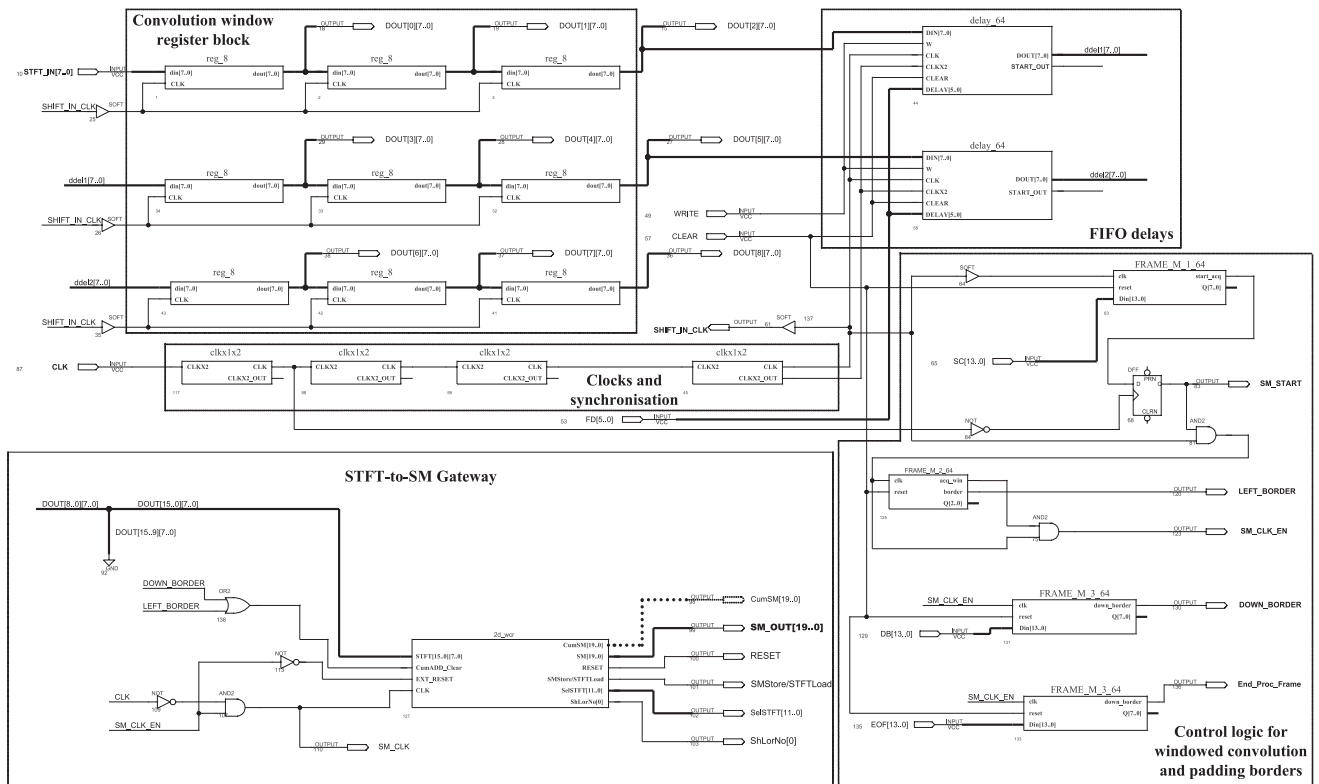


Fig. 2. Detailed schematic of developed FPGA implementation.

the 2-D WD. Note that the 2-D SM with arbitrary L requires $CN(L)$ clock cycles (by each point (k_1, k_2) of the 2-D frequency plane) to be executed. 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 the clock cycle time.

The presented hardware consists of two main parts: the convolution window register file and the STFT-to-SM gateway. The convolution window register file represents the hardware implementation of the 2-D convolution window function. It determines the order of the 2-D STFT input elements addresses for which the corresponding 2-D SM output will be computed according to the algorithm (2). The STFT-to-SM gateway is used for hardware realization of 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 , allowing the implemented S/SFDs to take different numbers of clock cycles for their calculation. This is made possible 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.1. Details can be found in [2]). These abilities lead to minimization of the critical performances of the multidimensional systems: hardware complexity, energy consumption and cost.

3 FPGA IMPLEMENTATION APPROACH

The module consists of nine registers, noted as “Convolution window register block”, simulates window’s sliding over the input 2-D STFT elements. Signal *STFT_IN* represents a 2-D STFT input data. *SHIFT_IN_CLK* clock signal enables registers’ loading in appropriate period of time. Sliding of the window over the input signal for one position left is done by loading one STFT element *STFT_IN* per clock cycle *STFT_IN_CLK*. Then, each element of the convolution window register block row k_1+1 (as well as rows k_1, k_1-1) is shifted by PIPO (parallel-in-parallel-out) shift registers to generate data in time index (k_2, k_2-1) . 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) . Note that the period of *STFT_IN_CLK* must be at least $CN(1)=5$ times greater than the period of system clock *CLK*, in order to enable the corresponding SM calculation in 5 CLKs.

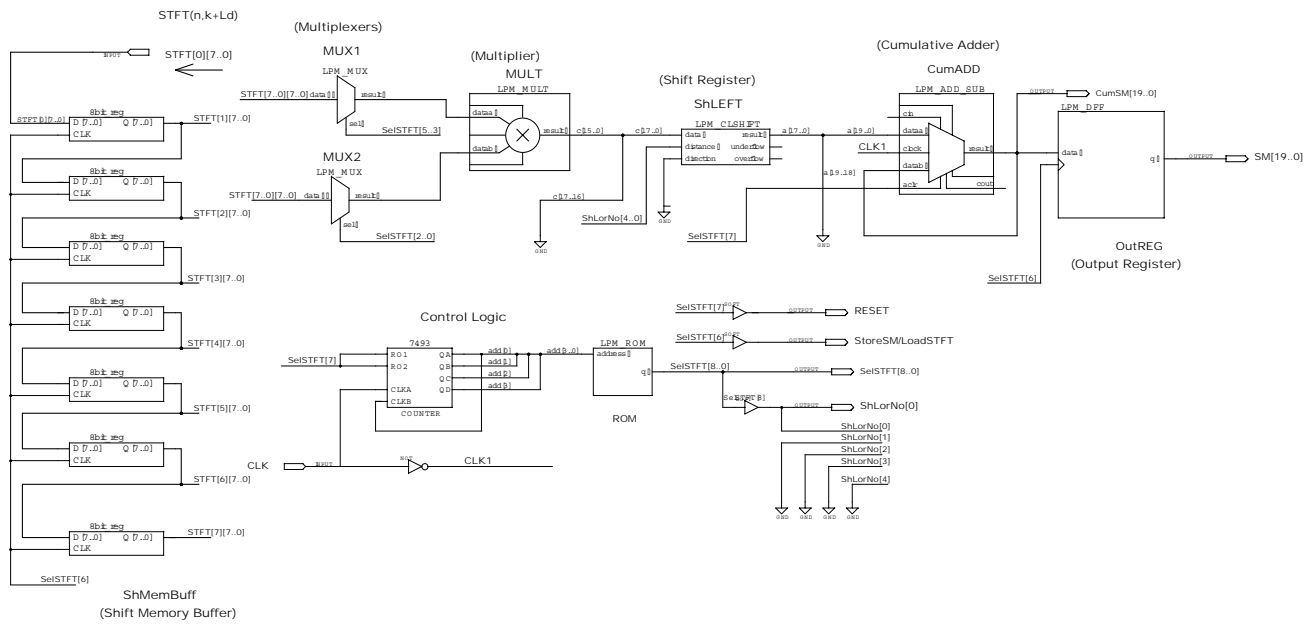


Fig. 3: The schematic diagram of the 8-bit STFT to SM gateway implemented in FPGA.

Convolution operations inside the frame are managed by data arrangement part, which is called "Control logic for windowed convolution and padding borders". The task of this block is to generate signals SM_START , SM_CLK_EN , $LEFT_BORDER$, $DOWN_BORDER$, End_Proc_Frame considering input parameters derived from frame size N and window size L . These parameters are stored in the "Configuration registers" module, Fig.1, and their values are as follows: $FD=N-(2L+1)$, $SC=2LN+(2L+1)-1$, $WS=2L+1$, $DB=(N-2L)\times N$, $EOF=N\times N-1$, [1]. Considering the input parameters as well as synchronization conditions related to the main clock signals CLK and $SHIFT_IN_CLK$, the signals SM_CLK_EN , $LEFT_BORDER$ and $DOWN_BORDER$ manage the operation of the STFT-to-SM gateway by generating its control signals $CumADD_Clear$, EXT_RESET and SM_CLK clock signal. The signal SM_START implicitly participates in the STFT-to-SM gateway operation managing through participating in generation of the other mentioned signals (SM_CLK_EN , $LEFT_BORDER$, $DOWN_BORDER$, End_Proc_Frame), whereas End_Proc_Frame signal indicates the end of whole calculation process. Note that the $LEFT_BORDER$ and/or $DOWN_BORDER$ signals cause generation of the $CumADD_Clear$ signal which resets cumulative adder integrated in STFT-to-SM gateway. For each window position, the SM_CLK_EN signal forwards the series of SM_CLK s that run SM calculation according to the algorithm given by eq.(2). After $CN(1) = 5$ SM_CLK s, the SM will be calculated and stored in

the output register. Additionally, the SM_CLK_EN signal resets the gateway when it takes zero value. When the window slides over the 2-D signal, the signals $LEFT_BORDER$ and $DOWN_BORDER$ are generated, to allow padding the borders of the frame with 0's. In the FPGA implementation of the system, shown in Fig.2, new library components were designed for "Control logic for windowed convolution and padding borders" module. It consists of different $FRAME_M_X$ modules for generating SM_START , $LEFT_BORDER$, SM_CLK_EN , $DOWN_BORDER$ and End_Proc_Frame signals, respectively. The basic components of these modules are variable length up-down counters with asynchronous reset and binary magnitude comparators. Each counter controls setting of the corresponding output signal from the "Control logic for windowed convolution and padding borders" module by counting up to the appropriate parameter's value from the "Configuration registers". The SM_START signal is generated to indicate that the system is ready for execution, considering input parameter SC (Start Convolution). Note that in the FPGA implementation, the SM_CLK_EN signal is used as clock signal for frames $FRAME_M_3_64$ that generate $DOWN_BORDER$ and End_Proc_Frame signals.

The FPGA implementation scheme of the complete system is given in Figs.2 and 3. The system units are implemented using the mixed approach allowed by design hierarchy, standard digital components from Altera's libraries, AHDL based mega functions and developed VHDL based modules. The adopted or developed VHDL or

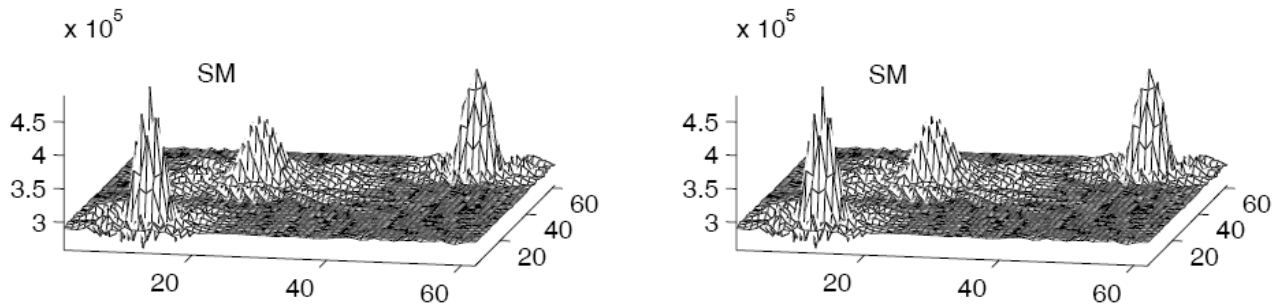


Fig. 4. The S/SF representation of the analyzed 2-D signal $f(x,y)+f_s(x,y)$ obtained by using the proposed hardware design (left-hand side), implemented in real FPGA devices (Altera's 10K series), and by numerical implementation (right-hand side).

Design	Signal duration	Device	LCs	LCs utilized	Memory bits	Memory utilized	Embedded cells	Embedded cells utilized	EABs	EABs utilized	Flip-flops required
MCI	64×64	EPF10K20RC240-3	921	79%	1216	9%	28	58%	4	66%	217
SCI	64×64	EPF10K50RC240-3	2438	84%	1024	5%	16	20%	2	20%	190
MCI	256×256	EPF10K30BC356-3	965	55%	4288	34%	28	58%	4	66%	227
SCI	256×256	EPF10K50RC240-3	2478	86%	4096	20%	16	20%	2	20%	200

Table I. Utilized silicon resource for 8-bit 64×64 and 8-bit 256×256 2-D STFT to 2-D SM implementation.

AHDL components have been parameterized in terms of input data size, horizontal and vertical depths of the FIFO delays, as well as of the window and image dimensions. Cascades of general latch registers (Altera's 8dff) build convolution window register block. The FIFO delay is composed from Altera's Cycle-shared FIFO Parameterized Megafunction (CSFIFO) with added threshold – read request feedback.

4 Testing and Verification

In order to verify the chip operation, before its programming, the compilation and simulation have been performed by processing usually complex 2-D test signal:

$$f(x, y) = \cos[20\pi(x - 0.75)^2 + 22\pi(y - 0.75)^2] + 0.5e^{j[-100\cos(\pi x/2) + 100\cos(\pi y/2)]} \quad (3)$$

in the range $|x| < 0.75$, $|y| < 0.75$, combined with the signal

$$f_s(x, y) = \cos\{1000\pi[(x + 0.5)^2 + (y - 0.5)^2]\} \quad (4)$$

whose, comparatively small, domain is $|x + y| < 0.1$, $|y - x - 1| < 0.1$. We have applied the Hanning window in the 2-D STFT definition, whose widths along the x and y axes are $W_x = W_y = 1$, respectively, and $N = 64$. The computed 2-D STFT elements (their real and imaginary parts), normalized at the range $[0, 255]$ and rounded to the 8-bit integers, are imported to the designed system

input. Results of the real-time implementation are presented in Fig.4, left-hand side. In order to verify the obtained results, numerical analysis, based on the same 2-D STFT elements, is performed and the results are presented in Fig.4, right-hand side. Accuracy of the results obtained by using the designed system can easily be checked. Note that the results from Fig.8 are computed at the point $(x, y) = (-0.25, -0.25)$.

After simulation and verification the Altera's EPF10K20RC240-3 chip is configured by using the synthesized code, [28]. It has 189 (51 input and 114 output) I/O pins. The rates of its 8-bit version silicon resources utilization are given in Table I, first row. Additionally, Table I gives the comparison of two approaches (MCI and SCI) for different signal duration $N \times N$ ($N = 64$ and $N = 256$ are considered) and 3×3 convolution window. It can be easily noted that the occupation of the silicon resources, described by the total number of logic cells (LCs), is significantly less in the MCI case. The targeting devices are selected according to the optimal resource occupation. Consequently, used MCI devices have smaller capacity than the corresponding SCI ones. Naturally, the same device could be used for both approaches (MCI and SCI) provided that its maximal capacity is determined by the SCI requirements. Also, for both approaches (MCI and SCI), the LCs slightly varies with signal's duration. On the other hand, the usage of memory

bits significantly increases with signal's duration. This is a consequence of the fact that delay functions are implemented by using FIFO memories. Precisely, for the $N \times N$ analyzed signal, the total number of used memory bits would be expressed as $2 \times N \times 8 + MBS$, where MBS represents the number of memory bits used for the implementation of the Look-Up-Table from the STFT-to-SM gateway control logic, Fig.1. Note that for SCI approach we have $MBS=0$.

4 Conclusion

FPGA implementation of the flexible system for S/SF signal analysis is presented. The system is based on the MCI of the 2-D SM. 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, within their execution. This property enables optimization of the critical design parameters.

References:

- [1] V.N. Ivanović, R. Stojanović, S. Jovanovski, and LJ. Stanković: "An architecture for real-time design of the system for multidimensional signal analysis", in *Proc. of the 14th EUSIPCO, Florence, Italy*, Sept.2006.
- [2] V.N. Ivanović, R. Stojanović, and LJ. Stanković, "Multiple clock cycle architecture for the VLSI design of a system for time-frequency analysis," *EURASIP Jou. on App. Sig. Pro., Special Issue on Design Methods for DSP Systems*, vol.2006, pp.1-18.
- [3] V.N. Ivanović, and LJ. Stanković, "Multiple clock cycle real-time implementation of a system for time-frequency analysis," in *Proc. of the 12th EUSIPCO, Vienna, Austrija*, Sept.2004, pp.1633-1636.
- [4] LJ. Stanković, "A method for time-frequency analysis," *IEEE Trans. on SP*, vol.42, no.1, 1994, pp.225-229.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] D. Petranović, S. Stanković, and LJ. Stanković, "Special purpose hardware for time-frequency analysis," *Electronics Letters*, vol.33, no.6, 1997, pp.464-466.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] 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.
- [15] L.Cohen, *Time-frequency analysis*, Prentice Hall, 1995.
- [16] D.E. Dudgeon, and R.M. Mersereau, *Multidimensional digital signal processing*, Prentice Hall, 1984.
- A. Iborra, C. Fernández, B. Álvarez, J.M. Fernández-Merono, "FPGA solution of low cost applications of real-time AVI systems," *Dedicated Sys.Mag.*, vol.Q2, 2001, pp.79-84.