Illia Verbovskyi, Valerii Zhabin

IMPROVING THE EFFICIENCY OF FUNCTIONS COMPUTATION IN ON-LINE MODE ON FPGA

The paper considers the possibility of reducing resources and increasing the speed of computing functions in on-line mode on the basis of recursive-digital low-pass filter.

Key words: redundant system, FPGA, dependent operations overlapping, on-line mode, recursive-digital low-pass filter.

Fig.: 6. Bibl.: 5.

Urgency of the research. Most of the algorithms designed to accelerate computations, do not involve optimizing the input and output process. Which requires more resources and time, particularly when it is implemented on FPGA. Therefore, there is a need to analyze and increase the number of operations that use this method. It can improve method characteristics.

Target setting. In the operation of computer systems in real time, when the duration of data processing is limited by external factors, we can parallelize calculations using a certain set of individual computing modules. But when it comes to the chain of operations, this method is impossible, because the operations are dependent. In this case, the result of one operation is used as an operand for the next. However, partial overlapping dependent operations is possible by on-line mode. This approach is now being used and is giving excellent results [5]. In addition, there is lack of a large number of algorithms for calculating functions, that implement on-line mode, for usage in FPGA.

Actual scientific researches and issues analysis. Over time, the number of algorithms that use on-line mode is increasing, but many of them still do not use the possibility of bitwise input, which does not fully solve the issue of reducing resources. In particular, in [1] the high radix algorithms that shortens the critical path of the multiplier is studied, and in [3] the application of algorithm for logarithm, exponential, and powering computation is investigated. Each of them does not implement full bitwise processing of operands, which requires the consumption of a large number of pins when working with multi-bit operands.

Uninvestigated parts of general matters defining. This article covers the analysis and usage of bitwise input when calculating functions in on-line mode. The study focuses on increasing speed and reducing equipment spending.

The research objective. Analyze the possibilities of improving the efficiency of calculations based on a recursive-digital low-pass filter in on-line mode.

The statement of basic materials. One of the approaches to solving the problem of reducing the number of connections between system components is the use of quasiparallel computing blocks (CB) that exchange data with each other using a serial code. Although the numbers are represented by a serial code at the inputs and outputs of such CB, their internal organization is closer to parallel devices. In this connection, they were called quasiparallel [4]. Based on the method and formulas considered in [4], we can obtain formulas for generalizing addition, subtraction and multiplication:

$$N_i = 2R_{i-1} + F_i, \tag{1}$$

$$R_i = N_i - z_i, \tag{2}$$

$$z_i = \begin{cases} -1, \ if \ \ N_i < -2^{-1}; \\ 0, \ \ if \ \ -2^{-1} \le N_i < 2^{-1}; \\ 1, \ \ if \ \ 2^{-1} \le N_i. \end{cases} \tag{3}$$

where N_i, R_i – internal variables, z_i – result digit, F_i – function increment, calculated from partial operands X_i and Y_i, according to the following formulas (addition/subtraction and multiplication, respectively):

$$F_i = 2^{-p}(x_i \pm y_i), \tag{4}$$

$$F_i = 2^{-p}(x_i Y_i + y_i X_{i-1}), \tag{5}$$

where p – on-line delay.

General model structure. The structure of the device for calculation is built by special modules using formulas (1-5). For example, to calculate a recursive-digital low-pass filter according to the following formula:

$$y_i = b \cdot (x_i - y_{i-2}) + c \cdot (x_{i-1} - y_{i-1}) + x_{i-2} + x_{i-1}, \tag{6}$$

it is necessary to construct the Synchronous Dataflow Graph (SDFG) [2] then to turn all formulas from SDFG into a tree of operations (Fig. 1).

Then, in the tree, the operation nodes, must be replaced with special computing blocks. These blocks perform one of the considered operations in formulas (4-5). The structure of the modular system that performs the calculations is shown in Fig. 2.

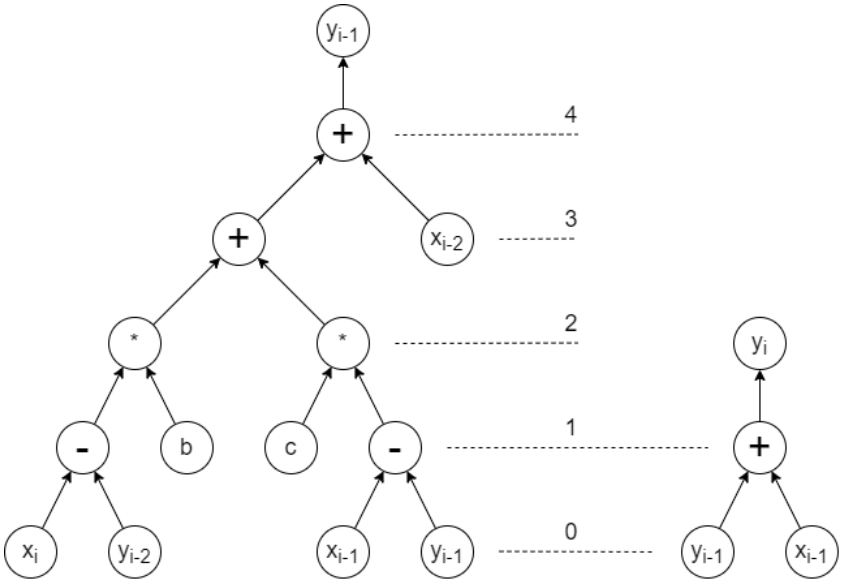


Fig. 1. Tree of operations for SDFG

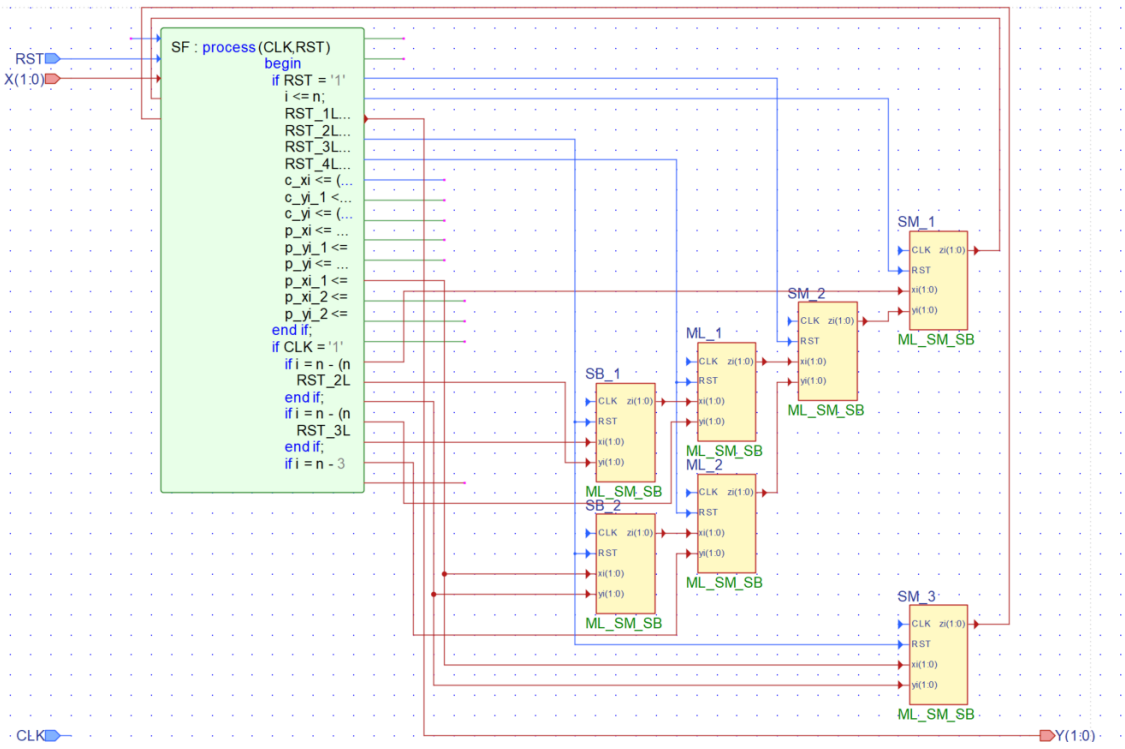


Fig. 2. Structure of the modular system

Experiments and analysis. As a result of the filter, the amplitude-frequency characteristic (AFC) is similar to the correct low-pass filter: there are decays at the appropriate frequencies, then phase change and repetition in the opposite direction, while the input and output values of the filter also remain systematic and similar to the sinusoidal and cosine signals, respectively (Fig.3).

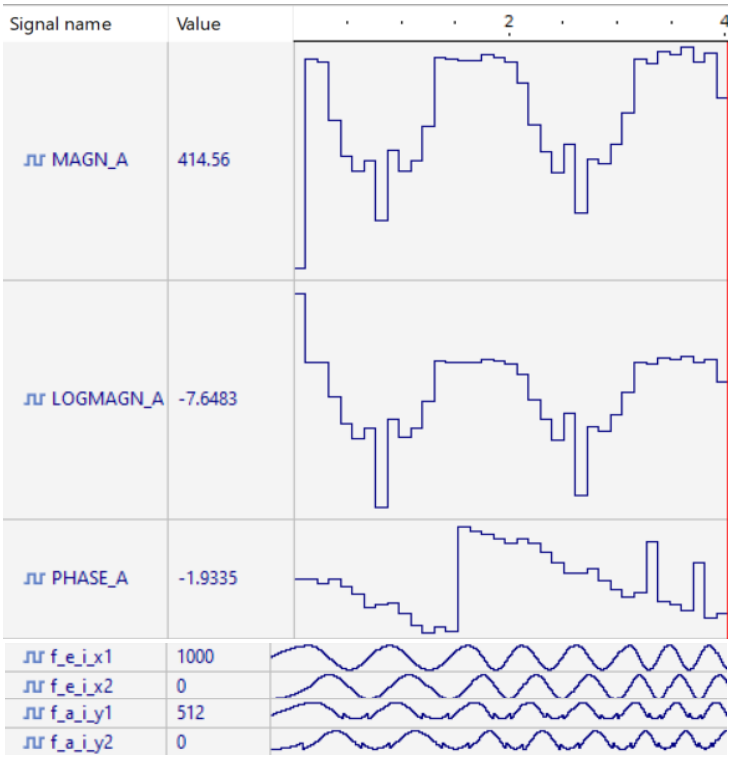
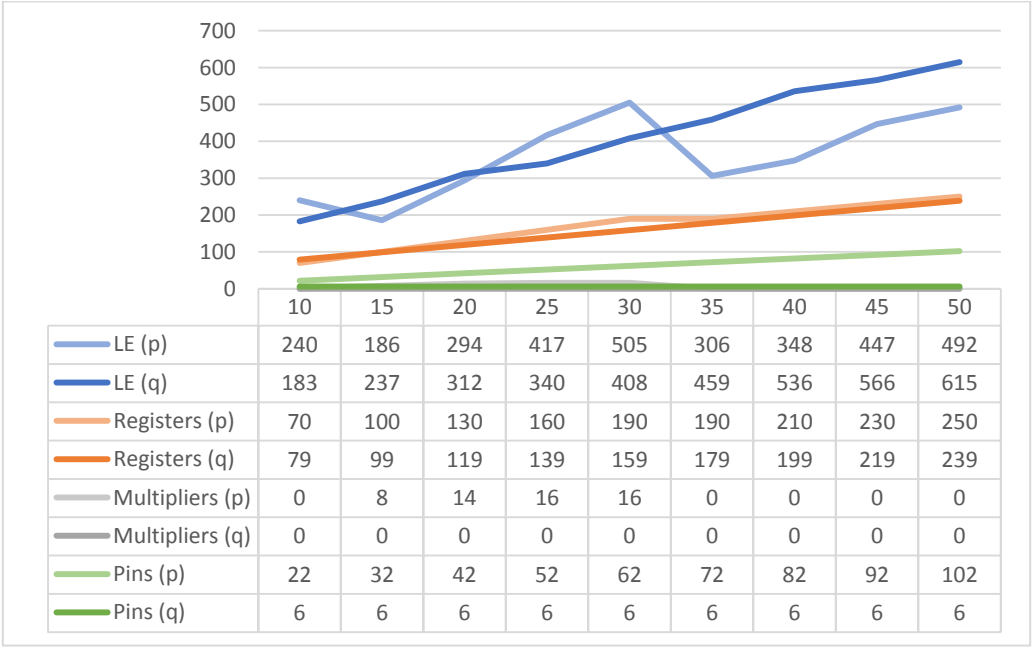


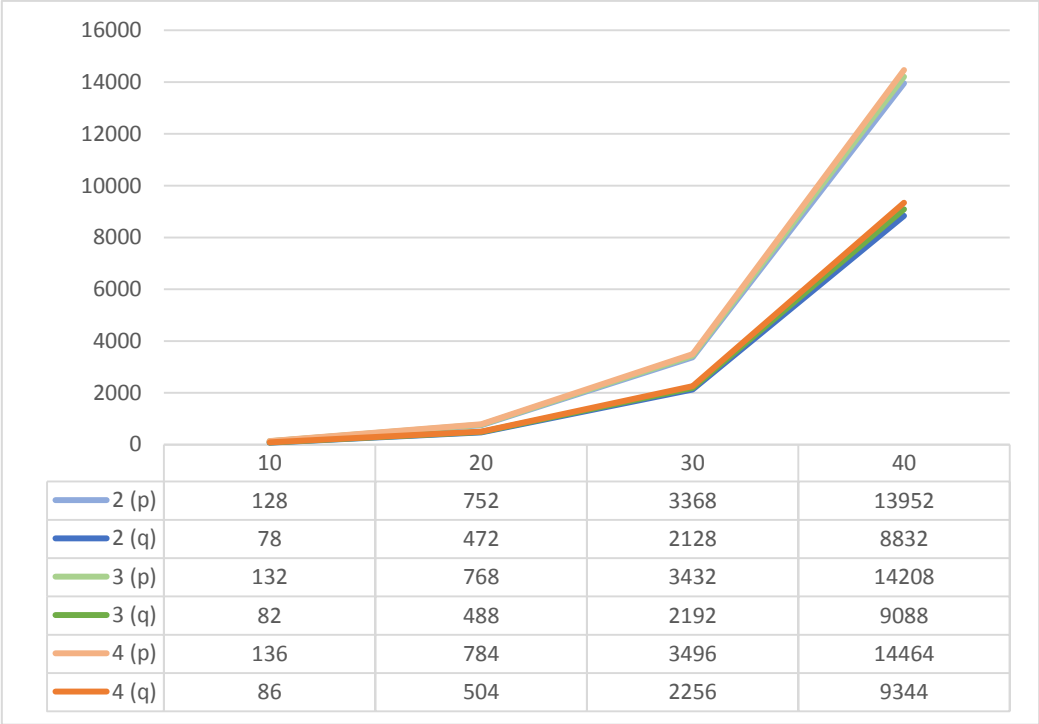
Fig. 3. AFC with I/O values of the developed filter

After testing and synthesizing the system on FPGA, were obtained the results of comparing the parallel and quasiparallel system in terms of resources used (Fig.4) and reducing the number of iterations (Fig.5).



	10	15	20	25	30	35	40	45	50
LE (p)	240	186	294	417	505	306	348	447	492
LE (q)	183	237	312	340	408	459	536	566	615
Registers (p)	70	100	130	160	190	190	210	230	250
Registers (q)	79	99	119	139	159	179	199	219	239
Multipliers (p)	0	8	14	16	16	0	0	0	0
Multipliers (q)	0	0	0	0	0	0	0	0	0
Pins (p)	22	32	42	52	62	72	82	92	102
Pins (q)	6	6	6	6	6	6	6	6	6

Fig. 4. Comparing the parallel (p) and quasiparallel (q) system in terms of resources used



	10	20	30	40
2 (p)	128	752	3368	13952
2 (q)	78	472	2128	8832
3 (p)	132	768	3432	14208
3 (q)	82	488	2192	9088
4 (p)	136	784	3496	14464
4 (q)	86	504	2256	9344

Fig. 5. Comparing the parallel (p) and quasiparallel (q) system in terms of number of iterations with on-line delay (p = 2, 3, 4)

Thanks to the on-line mode, dependent operations can overlap in time, which allows to reduce the time of calculations. A comparison of the calculation of function (6) in conventional and on-line mode of operation is shown in Fig. 6.

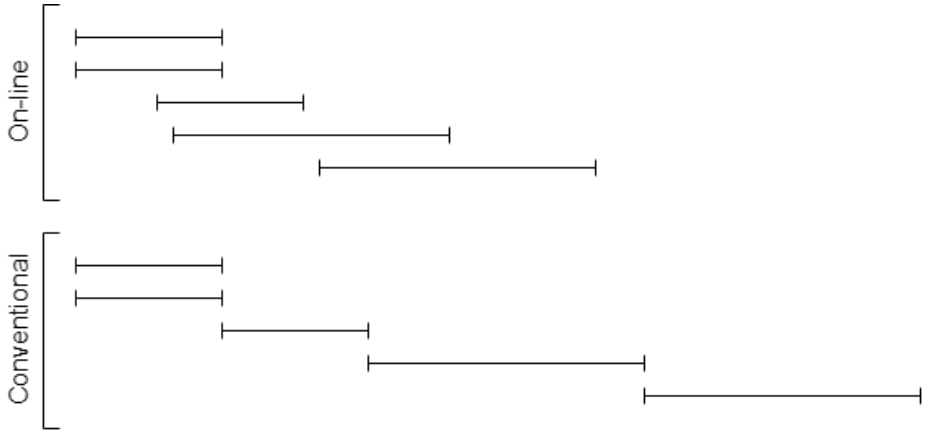


Fig. 6. Reduction of calculations time

Conclusions. The article shows the possibility of reducing resources and accelerating the calculation of a function such as a recursive-digital filter. A device has been developed which, for this example, allows to achieve an acceleration of up to 40% , as shown in Fig. 5, and reduce the number of pins to a constant value.

At the same time, this device has room for improvement. If after certain operations you do not increase the size of the number to increase accuracy, and leave it as in the input operands, the percentage of acceleration will increase. In addition, support for floating-point operations should be added, which will expand the range of numbers. All these changes will definitely improve the result.

References

1. Amin A. A. M., Shinwari M. High-Radix Multiplier-Dividers: Theory, Design, and Hardware. IEEE Transactions on Computers. 2010. Vol. 59, no. 8. P. 1009–1022.

2. Khan S. A. Digital Design of Signal Processing Systems. Chichester : John Wiley & Sons Ltd, 2011. 586 p.

3. Piñeiro J., Ercegovac M. D., Bruguera J. D. Algorithm and Architecture for Logarithm, Exponential, and Powering Computation. IEEE Transactions on Computers. 2004. Vol. 53, no. 9. P. 1085–1096.

4. Zhabin V. I., Korneĭchuk V. I., Tarasenko V. P. Computation of rational functions for many arguments. Automation and Remote Control. 1978. Vol. 38, no. 12. P. 1864–1871.

5. Zhabin V., Zhabina V., Verba O. Asynchronous On-Line Float-Point Computations in Systems with Direct Connections between Computation Units. IEEE 2nd International Conference on System Analysis & Intelligent Computing : Proceedings, Kyiv, 5–9 October 2020. Kyiv, 2020. P. 1–5.

AUTHORS

Verbovskyi Illia – student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: illyaverb@gmail.com

Zhabin Valerii – Doctor of Technical Sciences, Professor, Professor of Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: v.zhabin@kpi.ua